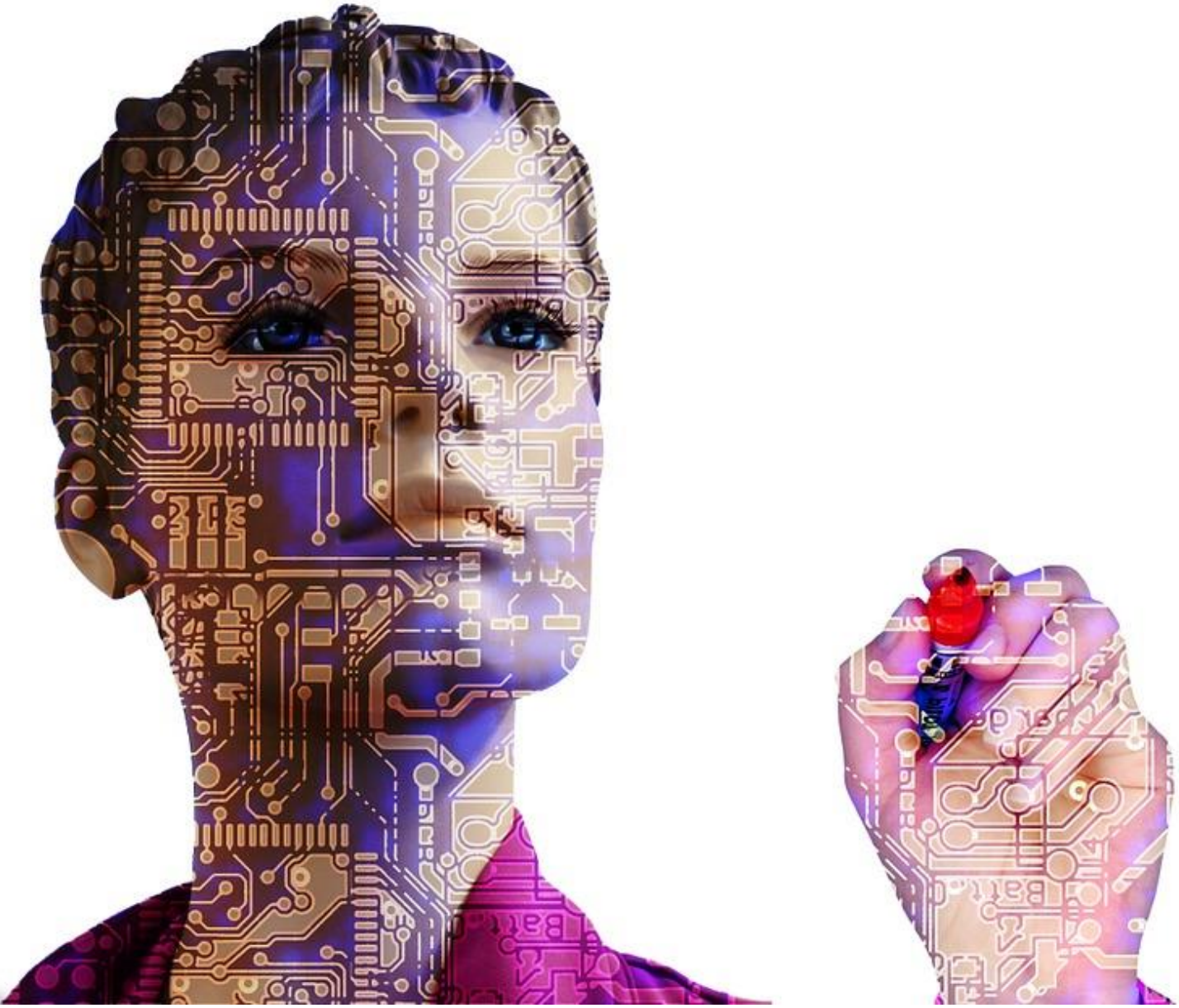
GCSE Computing
Revision Guide

OCR Specification



Contents

Systems Architecture 4
CPU 4
Systems Architecture 5
Von Neumann Architecture 5
Systems Architecture 6
Embedded Systems 6
Memory 7
RAM and ROM 7
Storage 8
Secondary Storage 8
Storage 9
Secondary Storage 9
Wired and Wireless Networks 10
Types of Networks 10
Wired and Wireless Networks 11
Client Server and Peer to Peer Networks 11
Wired and Wireless Networks 12
The Internet 12
Wired and Wireless Networks 13
The Internet 13
Network Topologies Protocols and Layers 14
Wi-Fi 14
Network Topologies Protocols and Layers 15
Star and Mesh Topologies 15
Packet Switching 15
Network Topologies Protocols and Layers 17
Protocols and Layers 17
Systems Security 18
Forms of Attack and Threats 18
Systems Security 19
Identifying and Preventing Vulnerabilities 19
Systems Software 20
Operating System 20

Systems Software 22
Utility Systems Software 22
Ethical, Legal, Cultural and Environmental Concerns 23
Open Source vs Proprietary Software 23
Ethical, Legal, Cultural and Environmental Concerns 24
Issues 24
Algorithms 26
Searching 26
Linear vs Binary Search 26
Algorithms 27
Sorting 27
Bubble vs Insertion vs Merge Sort 29
Algorithms and Programming Techniques 30
Pseudocode 30
Algorithms and Programming Techniques 32
Pseudocode 32
Producing Robust Programs 34
Defensive Design 34
Producing Robust Programs 35
Maintainability and Errors 35
Producing Robust Programs 36
Testing 36
Computational Logic 37
Logic Gates 37
Truth Tables 37
Applying Computing Related Mathematics: 38
Translators and Facilities of Languages 39
Types of Languages 39
Translators and Facilities of Languages 40
Translators 40
Translators and Facilities of Languages 41
The IDE 41
Representation of Data 42
Units 42

Representation of Data 43
Numbers - Binary 43
Representation of Data 44
Numbers - Adding Binary 44
Representation of Data 45
Numbers - Hexadecimal 45
Representation of Data 46
Shifts - Check Digits 46
Representation of Data 47
Images 47
Representation of Data 48
Characters 48
Representation of Data 49
Sound 49
Representation of Data 50
Compression 50

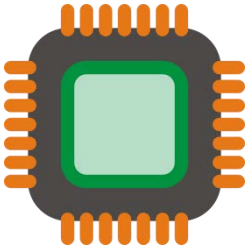
Topic 1.1

Systems Architecture

CPU

- Purpose of the CPU
- How common characteristics of CPUs affect their performance

Specification — Revised



ⓘ The purpose of the CPU is to carry out the processing of data on the computer system

ⓘ It performs the fetch-decode-execute cycle.

ⓘ The CPU fetches, decodes and executes instructions

The Performance of the CPU

Clock speed	Cache Size	Number of Cores
This determines the rate at which instructions are carried out each second The clock speed is measured in Hertz (Hz) A 3.6 Ghz processor carried out 3.6 billion calculations a second	Cache memory is a buffer that sits between the CPU and main memory. The CPU will check here first for instructions that have been fetched before The larger the cache the more space there is for instructions the CPU needs The cache has similar access speeds to the CPU and is therefore quicker to fetch instructions from	A core is an independent processor in the CPU A dual core has 2, quad core 4, hex core 6 processors working simultaneously The higher the number of cores the better performance of the computer as it can multitask

- Clock speed – number of fetch-decode-execute cycles a second
- Cache Size – high speed memory used by the CPU
- No of Cores – number of independent processors in the CPU working together

ⓘ The CPU contains **registers** which are temporary memory stores within the Cpu which have a specific purpose.

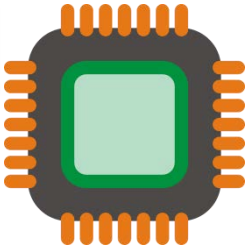
Systems Architecture

Von Neumann Architecture

- Von Neumann Architecture and the purpose of different registers

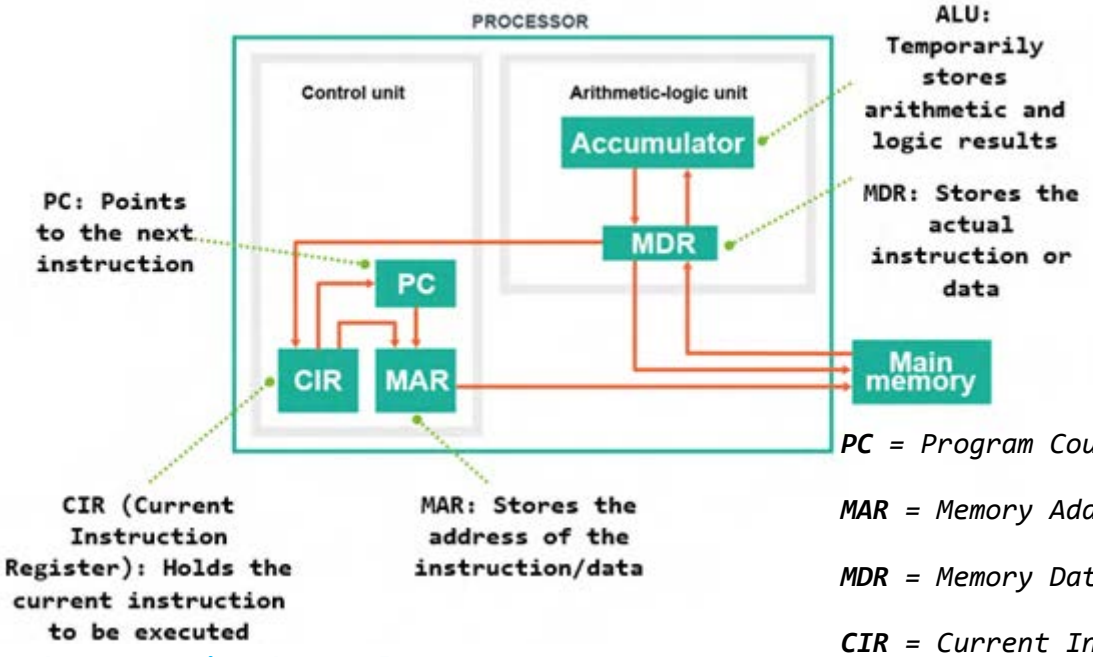
Specification

Revised



ⓘ The Von Neumann Architecture describes a system where data and programs/instructions are stored in the same main memory location

ⓘ The fetch-decode-execute cycle is the process of fetching these instructions from memory and executing them



PC *= Program Counter*

MAR *= Memory Address Register*

MDR *= Memory Data Register*

CIR *= Current Instruction Register*

The steps in the cycle:

Fetch	Decode	Execute
ⓘ Address from the PC is copied to the MAR ⓘ Instruction from MAR is fetched and copied to MDR ⓘ The instruction in the MDR is copied to the CIR ⓘ Increment the PC	ⓘ The instruction in CIR is decoded by the control unit ⓘ Data may be loaded into the MDR	ⓘ The instruction is performed ⓘ The ALU may be used for any logic or calculations ⓘ The result is stored in the accumulator

Systems Architecture
Embedded Systems

- Purpose of embedded systems
- Examples of embedded systems

Specification

Revised

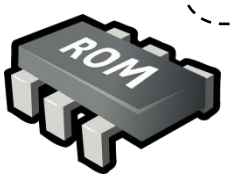
- An embedded system is one which has a processor built in to another device
- A computer System that is made up of both Hardware and Software often known as Firmware
- Usually for very specialised tasks
- Doesn't usually contain an Operating System

Examples:
- Dishwasher
- Microwave
- Fridge
- Smart phone
- TV

Memory
RAM and ROM

- The difference between RAM and ROM
- The purpose of RAM in a computer system
- The purpose of ROM in a computer system
- The need for Virtual Memory
- Flash Memory

Specification — Revised



RAM and ROM

Random Access Memory (RAM)	Read Only Memory (ROM)
Purpose = Stores data and programs currently being used by the computer	**Purpose** = Stores instructions needed to start up the computer – contains the boot program
ⓘ Can be changed by the computer at any time	ⓘ Programmed during the computers manufacture and cannot normally be changed
ⓘ Volatile memory (data is lost when the power is turned off)	ⓘ Non-Volatile (data is not lost when the power is turned off)
ⓘ Larger memory (Starting at 4GB in most computers)	ⓘ Small (Only MB needed for the boot program)
ⓘ The more RAM the more programs that can be run at the same time. Allows for more multitasking.	ⓘ ROM is needed as it is always there to start the computer

Flash and Virtual Memory

Flash	Virtual
ⓘ Flash memory is type of non-volatile (ROM) memory that can be changed and does not need a power supply to keep its contents ⓘ There are no moving parts which make it fast and reliable ⓘ Examples of flash memory in use: ⓘ Memory cards in digital cameras. ⓘ Mini/Micro SD cards in Smartphones. ⓘ USB memory sticks. ⓘ Solid state drives	ⓘ Virtual memory is part of the hard drive used as an extension to RAM. If there is not enough RAM to hold all the data and run the programs needed then it will make use of some of the hard drive. ⓘ Access speeds from the hard drive are slower than from RAM. More RAM reduces the need for virtual memory which improves performance.

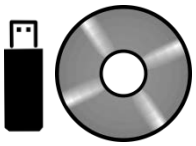
GCSE COMPUTING

Storage

Secondary Storage

- The need for secondary storage
- Data capacity and calculation of data capacity requirements
- Common types of storage: Optical, Magnetic, Solid State

Specification — Revised



ⓘ Secondary storage is needed to store our files and programs when they are not in use. These commonly fall into three categories: Optical, Magnetic and Solid State.

ⓘ Secondary storage is long term, non-volatile storage

Optical	Magnetic	Solid State/Flash
ⓘ Lasers write data to the surface of a disk. ⓘ Optical media includes: CD, DVD, Blu-Ray ⓘ Excellent for distributing software ⓘ Good capacity ⓘ Low cost ⓘ Light and portable ⓘ Can get damaged over time ⓘ Slow access speed	ⓘ The magnetic tape is moved along a read-write head inside a disk drive ⓘ Examples: hard disk and tapes ⓘ Used for backups ⓘ High capacity ⓘ Cheap ⓘ Reliable ⓘ Slow to read due to moving parts	ⓘ No moving parts make solid state memory have a very fast access speeds. Most are a type of flash memory ⓘ Examples: USB drives, memory cards, solid state hard drives ⓘ Large capacity but less than magnetic tape ⓘ More expensive ⓘ Portable ⓘ Reliable and not affected by being moved around

Calculating Storage Requirements

ⓘ From the section on data representation we know how many bits are in a byte and how many bytes in a kilobyte etc. We also looked at how to calculate the size of an image.

ⓘ Using the knowledge we can calculate how much storage will be needed in different scenarios. For example:

A text file that contains 10000 characters. Give your answer in KB

We know that each character is 1 byte in ASCII. So 10000 x 1 = 10000. There are 1024 bytes in a kilobyte so 10000/1024 = 9.77kb

Storage

Secondary Storage

- Suitable storage devices and storage media for a given application, and the advantages and disadvantages of these, using characteristics:
  - Capacity
  - speed
  - portability
  - durability
  - reliability
  - cost

Specification — Revised

ⓘ When choosing an appropriate device for a given scenario you need to consider the following characteristics:

- **Capacity**: How much space there is to store files.
- **Access Speed**: How quickly the computer can read and write data to or from a storage device or write data to it.
- **Portability**: Can you easily unplug it and carry it away?
- **Durability**: How easily is it damaged? Will it survive being dropped?
- **Reliability**: Can the data always be accessed and be correct?
- **Cost**: How expensive is the storage device

Quick Comparison:

Access Speed

Slowest — *Optical Disk* — *Magnetic Tape* — *HDD* — *USB/Flash* — *SDD* — Fastest

Cost

Cheap — *Magnetic Tape* — *Optical Disk* — *HDD* — *USB/Flash* — *SDD* — Expensive

Capacity

Small — *Optical Disk* — *USB/Flash* — *SDD* — *HDD* — *Magnetic Tape* — Large

Portability

Difficult — *SDD* — *HDD* — *Magnetic Tape* — *Optical Disk* — *USB/Flash* — Easy

Durability/Reliability

Low — *Optical Disk* — *Magnetic Tape* — *USB/Flash* — *SDD* — *HDD* — High

GCSE COMPUTING

Wired and Wireless Networks

Types of Networks

- Types of Networks: LAN and WAN
- Hardware needed to connect stand-alone computers into a Local Area Network

Specification — Revised

ⓘ A network is 2 or more computers connected to each other.

Advantages of a network:

1. It allows communications between workers.
2. It allows data to be shared.
3. It allows peripherals to be shared.
4. It allows computers to be upgraded more easily.
5. It allows distributed processing

WAN	LAN
ⓘ Wide Area Network ⓘ Computers are geographically remote/long distance away ⓘ Communication medium is not owned by the company	ⓘ Local Area Network ⓘ Covers small geographic area located on one site ⓘ All hardware for a LAN is owned by the organisation that uses it

Hardware Needed

Network Interface Card (NIC)	Wireless Access Points
ⓘ Any device connected to a network needs to have a NIC (normally built into the motherboard) ⓘ Produces electrical signals for receiving and sending messages on the network ⓘ Has a unique MAC address	ⓘ Allows wireless devices to connect to a network ⓘ A type of switch for wireless devices
Router/Switch	**Transmission Media**
ⓘ A switch connects devices together on a LAN. ⓘ They send data along a network using the MAC address for the destination. ⓘ A switch sends data to the intended destination only ⓘ A router is used to send data between networks and connect devices to the internet	ⓘ Physical wires that connect devices together in a LAN. ⓘ **Twisted Pair** (Cat5e/Cat6): Most common. Cheap and easy to install. Fast and reliable transmission. ⓘ **Coaxial**: Bulkier than twisted pair. Made of a single copper wire which is insulated to minimise interference ⓘ **Fibre Optic**: Can transmit data as light. High performance and do not suffer from interference. Can transmit over large distances but is expensive

Wired and Wireless Networks

Client Server and Peer to Peer Networks

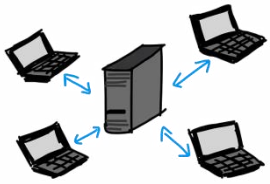
- The different roles of computers in a client server and a peer to peer network
- Factors that affect the performance of networks

Specification

Revised

ⓘ There are two main ways that networks can be organised. Once involves a server and the other does not

Server = A server is essentially a more powerful computer that manages a network and provides services to the clients. Some of the functions of the server are: to share files, provide security, provide access to programs and to backup files.

Client Server	Peer to Peer Network
 ⓘ Has a designated server. The nodes are clients and make requests to access files or programs from the server. ⓘ The server is a powerful machine typically with more memory that can serve the needs of the clients. ⓘ Handles the security (logins), backup and installation of software ⓘ Expensive to set up and you need to have a specialist to maintain the network ⓘ If the server goes down the clients cant access the network	 ⓘ All computers are equal ⓘ No server ⓘ As there is no server backups and updated need to be done individually ⓘ Copying of files between devices creates duplicates ⓘ Tend to be slow ⓘ Security is up to each user ⓘ Suitable when a small network such as a home network

Performance of Networks

- ⓘ Bandwidth = the amount of data that can be transferred in a given time. The higher the bandwidth the better the performance
- ⓘ Too many users or heavy use can slow down the network
- ⓘ Wired connections are generally faster than wireless connections and more secure
- ⓘ Hardware used to setup network can have an effect
- ⓘ Network topology used

Wired and Wireless Networks

The Internet

- The internet as a worldwide collection of computer networks:
  - DNS (Domain Name Server)
  - Hosting
  - The Cloud
- The concept of virtual networks

Specification

Revised

The Internet = A collection of worldwide networks. It is the largest WAN connecting networks all around the world.

World Wide Web = is not the same as the Internet. The WWW is a collection of websites

Hosting = A host is another computer that stores a particular resource. For example websites are hosted on other computers/servers which we access by typing in the address.

The Cloud	
Uses the Internet to store files and applications that we can access remotely. An example is Google Drive where we can access out files from anywhere with an Internet connection.	
Advantages	**Disadvantages**
ⓘ Can offer increased storage when needed ⓘ Easy to share files ⓘ Can access work and files from anywhere as long as you are connected ⓘ Provides security and backup automatically for you ⓘ Don't need to pay IT staff to manage the hardware and there is no need to purchase expensive hardware	ⓘ Need a connection to the internet ⓘ Need to trust someone else to backup and secure your data ⓘ Can be vulnerable to hackers ⓘ Subscription fees may be expensive ⓘ There are issues with who owns the data stored on the cloud

Wired and Wireless Networks

The Internet

- The internet as a worldwide collection of computer networks:
  - DNS (Domain Name Server)
  - Hosting
  - The Cloud
- The concept of virtual networks

Specification — Revised

Domain Name System – DNS

Websites all have a unique IP address which is used to access them. These are hard to remember so they have a domain name which is used in the uniform resource locator (URL) such as www.google.co.uk.

The DNS translates these URLs or domain names into the IP address so you don't need to remember them.

Benefits of using a DNS

- Constantly updated by other DNS servers
- When you request an address(URL), the DNS server looks up the URL and returns the IP address, or
- Searches for the address from other DNS servers
- People do not need to remember IP addresses
- As long as you are connected to a DNS server you can have access to all the addresses

Virtual Network

Virtual Network = A network that is software based. It uses the same existing physical network but creates other individual networks without having to rewire them.

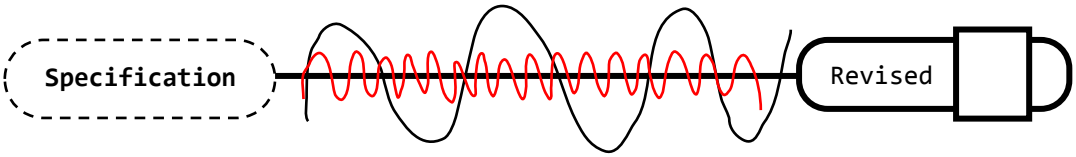
- Each virtual network has its own security and firewall
- Several virtual networks can exist on the same physical network
- Virtual networks can also be created by using the services of the cloud
- An example could be in a school where a virtual network is set up which allows all the students to be connected and this is separate to the virtual network for all the admin staff

GCSE COMPUTING

Network Topologies Protocols and Layers

Wi-Fi

- Frequency and Channels
- Encryption



To connect to a wireless network the following hardware devices can be used: Wireless Access Point, Router, Modem, Hub.

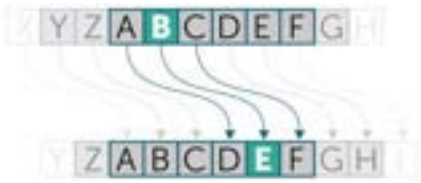
Frequency	Channels
ⓘ Frequency is the rate at which the signal changes ⓘ The number of times it repeats per unit of time (GHz) ⓘ Wifi uses two radio frequency bands. 2.4 GHz and 5 GHz.	ⓘ A channel is the range of frequencies that will transmit data ⓘ Channels overlap ⓘ 2 devices using the same channel may suffer from interference ⓘ Changing the channel can reduce interference.

Encryption

Data that is transmitted over a network can be intercepted. Encryption is used to prevent the data being understood if it is intercepted.

Encryption = putting the data into a code that cannot be understood unless you have the key to decrypt it.

The most common encryption security for Wi-Fi is WEP and WPA

Caeser Cipher	Symmetric Encryption	Public Key (Asymmetric Encryption)
Letters are shifted by a given number 	Same key used to encrypt and decrypt a message	Two keys! A public key known to everyone for encrypting and a private, secret key for decrypting.

Network Topologies Protocols and Layers

Star and Mesh Topologies

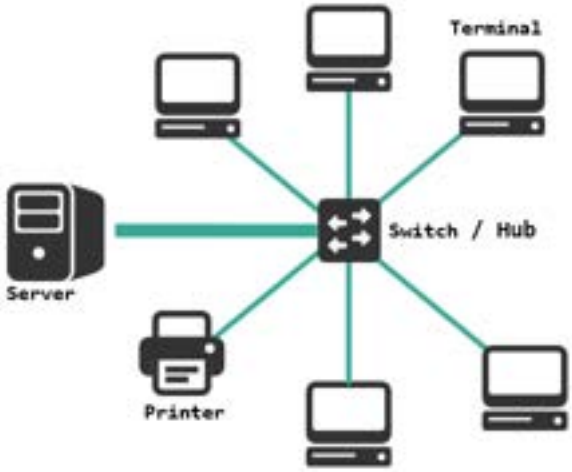
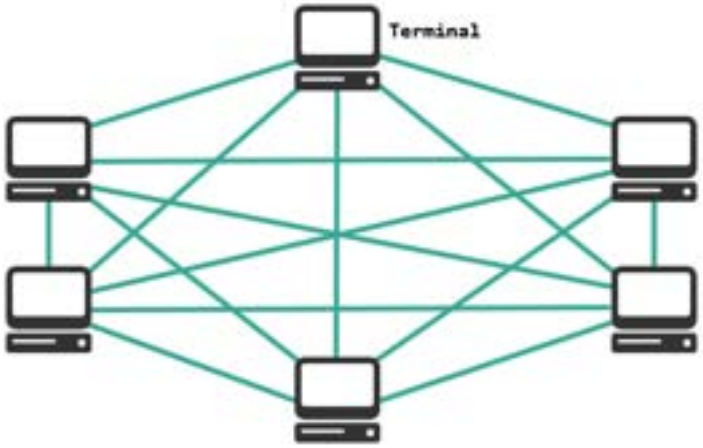
Packet Switching

- Star and Mesh Network Topologies
- Packet Switching
- Ethernet

Specification

Revised

Topology = the layout of the network

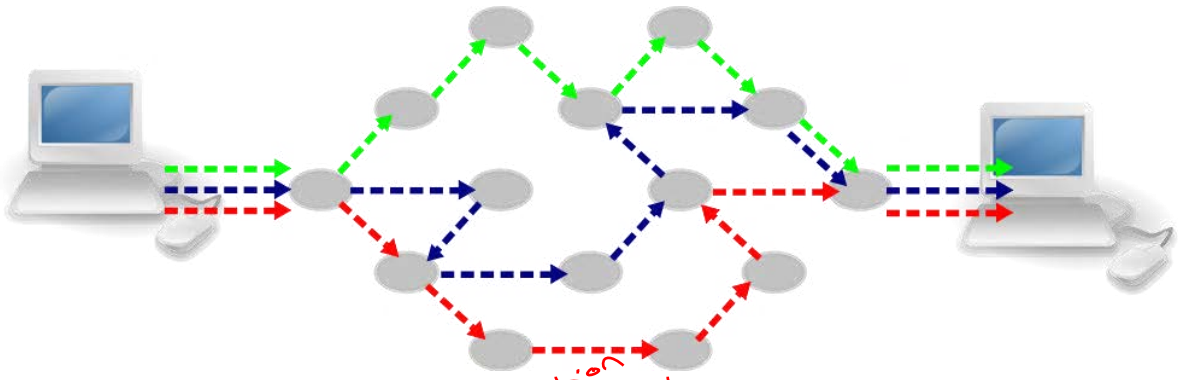
Star Network	Mesh Network Network
ⓘ All computers are connected to a central switch or server that controls the network ⓘ Data is sent to the server which then sends it to another device.	ⓘ A decentralised network that allows devices to be connected directly or indirectly to each other ⓘ Data is sent along the fastest route from one device to another ⓘ You can have a full or partial mesh
	
ⓘ Fast data transfer as there are fewer collisions ⓘ If one cable fails the other terminals are not affected ⓘ If the central device (switch or server) goes down the whole network goes down	ⓘ Data can be trasmitted to different devices simultaneously ⓘ An involve redundant connections ⓘ Network maintenance and administration can be difficult

Packet Switching

= Splitting data to be sent over a network into equal sized packets and then they are sent tacking different routes

- The computer splits the file into packets
- Each packet is of a fixed size
- The packets are given a header including the destination address and the packet number
- Packets find their own way across the network to the destination
- Server waits until all packets have arrived
- Server reorders packets to create the file
- Any missing / non-arriving packets are re-requested
- error checking is performed on receipt of packets

Contents of a Packet		
Header	**Payload**	**Footer**
Destination Address Packet number	The data that is being sent	Error checking such as a check sum



Ethernet

Ethernet is a protocol

Transport connection protocol
internet protocol

- It is within the TCP/IP stack
- It governs the connection of devices over the internet
- Governs the transmission of data between devices over the internet
- Uses cables to transmit data between devices in a LAN Local area network

Protocol = *Rules/Agreed ways of doing something such as connecting or communicating*

Network Topologies Protocols and Layers

Protocols and Layers

- Protocols that include TCP/IP, HTTP, HTTPS, FTP, POP, IMA, SMTP
- The concept of Layers

Specification — Revised

Protocol = *Rules/Agreed ways of doing something such as connecting or communicating*

TCP/IP = *Protocol for how data is sent between networks*

TCP= Transmission Control Protocol	IP = Internet Protocol
ⓘ Sets the rules for how devices connect to a network ⓘ Splits the data into packets ⓘ Reassembles the packets at the other end ⓘ Checks data is correctly sent and delivered	ⓘ Is responsible for the packet switching WAN

Other Protocols

HTTP = Hyper Text Transfer Protocol	Used by websites and to communicate with web servers
HTTPS = HTTP Secure	A secure version of HTTP
FTP = File Transfer Protocol	Uses to send or retrieve files to or from a server LAN
POP3 = Post office Protocol 3	Retrieve emails from a server. Held until downloaded and then deleted
IMAP = Internet Message Access Protocol	Retrieve emails from a server. Stored on server until deleted. Can view from several devices.
SMTP =Simple Mail Transfer Protocol	Used to send emails between servers

Layers = *Group of protocols that share similar functions*

ATND

Layer Name	Protocols in Layer	Examples
4. Application Apps	Selects the correct protocol depending the application. E.g. sending an email or viewing a website	*HTTP, FTP, SMTP*
3. Transport	Controls the data flow and splitting data into packets	*TCP*
2. Network	Making connections and controlling the packet switching directing the data packets	*IP*
1. Data Link	Physical hardware that connects 2 hosts such as the NIC and cabling	*Ethernet*

Data can be passes between adjacent layers. E.g. layer 2 can pass data to layer 1 and 3, but layer 1 can only pass data to layer 2

- ⓘ Self-contained which allows them to function without affecting the other layers
- ⓘ They can be changed without affecting the other layers
- ⓘ *It allows network communication to be broken down into manageable pieces*
- ⓘ *Having set rules(protocols) ensures that companies make compatible hardware and software*

GCSE COMPUTING

Systems Security

Forms of Attack and Threats

- Malware
- Phishing
- People as the 'weak point' in secure systems (social engineering)
- Brute force attacks
- Denial of service attacks
- Data interception and theft
- The concept of SQL injection
- Poor Network Policy

Specification — Revised

Virus	ⓘ Software that replicates itself ⓘ Deletes data // fills hard drive space // slows computer
Malware	ⓘ Malicious software that can take different forms such as Viruses(program that replicates itself and causes damage), Trojan horses(software that tricks the user into installing it), spyware (gathers information about the user)
Phishing	ⓘ Using fraudulent emails to try and obtain your personal information such as passwords and credit card numbers. Often present to be from reputable companies
Brute Force	ⓘ A method of trying to gain access to data by tying all possible combinations to discover a user's password.
Denial of Service	ⓘ Designed to shut down a network or webserver by flooding it with traffic that it cannot handle.
Social Engineering	ⓘ Social engineering is the art of manipulating people so they give up confidential information
SQL Injection	ⓘ SQL is the language used to control databases. ⓘ Malicious code entered into a website form to modify he SQL statement that is executed resulting in either: unauthorised access to data; modification of data; deletion of data; insertion of data ⓘ Can be prevented by validating user input

Systems Security

Identifying and Preventing Vulnerabilities

- Penetration testing
- Network forensics
- Network Policies
- Anti-Malware software
- Firewalls
- User access levels
- Passwords
- Encryption

Specification — Revised

Network Policy	ⓘ Agreed procedures for people in an organization to follow. This sets out what you can and can't do. For example not being allowed to use external storage devices or using email for personal use.
Penetration Testing	ⓘ Testing by simulating a possible attack in order to identify any vulnerabilities
Network Forensics	ⓘ Monitoring and analysis of computer network traffic to check for intrusion
Anti-Malware Software	ⓘ Software that is designed to detect and remove malware from infecting the computer.
Firewalls	ⓘ Prevents unauthorised access to a network by checking incoming traffic.
User Access Levels	ⓘ Limiting the access to information depending on the type of user. For example an administrator would have access to all files whereas an employee at a lower level would only see he files they need.
Strong Password Features	ⓘ A strong password is one that is difficult to guess. A strong password will be at least 6 characters and use upper and lower cases, numbers and symbols.
Encryption	ⓘ Encoding data so that it cannot be understood if intercepted without the key.

GCSE COMPUTING

Systems Software
Operating System

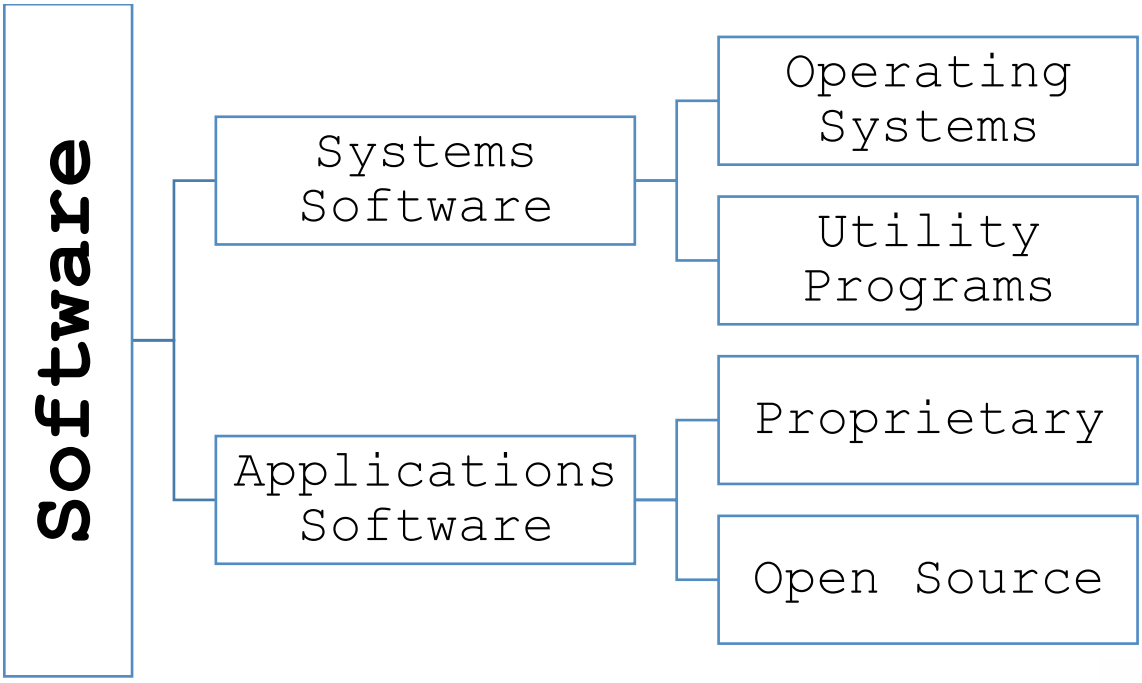
- The purpose and functionality of systems software
- Operating systems:
  - User Interface
  - Memory Management/Multitasking
  - Peripheral Management and Drivers
  - User Management
  - File Management

Specification — Revised

Systems Software = Software designed to maintain and control the hardware, it also provides and interface between software and hardware. The two main types of systems software are operating systems and utility programs

Operating system = controls the hardware and software/provides and interface for the user

Utility programs = help to maintain the computer



Function	Description
User Interface	ⓘ Allows the user to communicate with the hardware. **Different Types:** Command line, Graphical User Interface (GUI), Menu Drive, Voice For a GUI remember WIMP for the key features: **W =** Windows **I =** Icons **M =** Menus **P** = Pointers
Memory Management/Multitasking	ⓘ Allocates memory to applications – can run more than one piece of software at a time ⓘ Removes data no longer needed – frees up space for other programs ⓘ Moves data between **RAM and Virtual Memory** – allows more programs to run ⓘ **Multitasking:** The OS can run multiple applications at the same time ⓘ They are taking it in turns to get processor time to execute instructions ⓘ The OS must manage how the processes share the processor
Peripheral Management and Drivers	ⓘ A **device driver** is a program that controls a peripheral device such as a printer. It allows the operating system to communicate with the device ⓘ A **peripheral** is an external device not directly connected to the CPU such as a monitor or keyboard ⓘ The operating system deals with taking input from the devices and sending output to them.
User Management	ⓘ Provides user **names and passwords** for security ⓘ **Levels of access**. Gives different users access to specific data or resources ⓘ Operating systems can be single user or multi user. A multiuser OS allows several users to use the computer at the same time (e.g. for a mainframe or supercomputer).
File Management	ⓘ Stores data in a hierarchical structure which allows data to be stored in a organised way ⓘ Manages where data is stored and the user does not need to know how ⓘ Allows files and data to be retrieved and edited.

GCSE COMPUTING

Systems Software

Utility Systems Software

- Utility Systems software:
  - Encryption Software
  - Defragmentation
  - Data Compression
  - The role and methods of backup:
  - Full
  - Incremental

Specification — Revised

Encryption	ⓘ Puts data in a code (encrypts) that cannot be understood unless the user has a key to decrypt it. ⓘ Prevents unauthorised people accessing the data if it is intercepted.
Defragmentation	ⓘ Data when stored on a hard disk can get fragmented (split up) which makes it slower to access the data. Data is saved in the available spaces and not always grouped together. ⓘ This can happen as files are moved, deleted or changed size. ⓘ Defragmentation reorganises the data so that it is stored together
Compression	ⓘ Reduce the size of files before sending or saving ⓘ So they can be transmitted faster or take up less space ⓘ Allows media on websites to load faster
Backup	ⓘ A backup is making a copy of the original file ⓘ A full backup copies every files on the system. This can can take time and a lot of space ⓘ An incremetal backup only copies files that have changed since the last backup was made. This is faster to create and uses less space. If a full restore is needed this is slower than resroring from a full backup

GCSE COMPUTING

Ethical, Legal, Cultural and Environmental Concerns

Open Source vs Proprietary Software

- Open Source vs Proprietary Software

Specification

Revised

Proprietary

Proprietary software cannot be copied/altered (without permission of the copyright owner). It is distributed only as a compiled program/source code not available

Advantages	Disadvantages
Support is available if there is a problem Updates are regularly available Will have tested extensively	Can be expensive as you have to purchase licenses It may be inflexible to the user's needs Can tie you in to uses one particular platform

Open Source

Open source software is distributed with its source code and can be modified

Advantages	Disadvantages
License free and its source code is available to others to use and improve It can be altered as the source code is available Can be reliable as many people modify and improve it	May not be anyone to contact if something goes wrong Updates may not be available or be irregular

Why you would choose to make your software Open Source:

- Other people can improve and maintain your code
- You can receive an income from advertising
- Could reach a wider audience

Why you would choose to make your software Open Proprietary:

- Receive an income from selling
- Copyright your code so you can prevent people from modifying and using

Ethical, Legal, Cultural and Environmental Concerns

Issues

- Ethical Issues
- Leal Issues
- Cultural Issues
- Environmental Issues
- Privacy Issues

Specification

Revised

Legal = What is considered right and wrong in the eyes of the law

The Data Protection Act 1998	ⓘ 8 principles relating to the gathering and storage of personal data. ⓘ Data must be kept secure, up to date, cannot sell without permission ⓘ Data subject have the right to see data held about them
Computer Misuse Act 1990	ⓘ To prevent hacking and unauthorised access. ⓘ Covers 3 offences: 1. Gaining unauthorised access 2. Gaining unauthorised access with the intent to commit a rime 3. Gaining unauthorised access and modifying material.
Copyright Design and Patents Act 1988	ⓘ Protects anything that has been created such as images, software, books ⓘ You cannot make copies without permission
Creative Commons Licensing	ⓘ Allows you to legally share media and software with others by giving them permission to use
Freedom of Information Act 2000	ⓘ Allows the public to access information held by organisations

Stakeholder = Individuals or Groups of people who have an interest in an organisation

There will be some overlap between these categories when discussing certain issues. For example Computer Surveillance may be an ethical, cultural and privacy issue.

Ethical

= An **ethical** act is something that is fair and morally right. Sometimes issues arise that are not covered by any laws.

Examples of ethical issues around computer systems are:

- The Digital Divide = Will everyone be able to afford it?
- Will there be fair charging?
- Are some countries being exploited as a source of cheap labour for call centres and for programming?
- Does the system design promote accessibility for all?
- Is the computer system being misused?
- Cyberbullying and Trolling are problems
- Censorship of the Internet

Cultural

= Issues related to how groups of people with certain beliefs or practices may be affected. Also concerned with changes in society.

- Cyberbullying and Trolling are problems
- Health issues linked to the use of computers. E.g. eye strain, repetitive strain injury, back pain.
- Can improve access to information which can lead to better lifestyles.
- Can lead to stress as we are always 'switched on' such as always checking emails for work
- Some cultures or groups of people may not have access to certain technology.
- Changing the way we work, shop and access information and services.

Environmental

Computer systems can be good for the environment but can also harm it with waste and energy use.

- Computer systems can reduce the number of resources used such as paper
- Creating e-waste. Waste is an issue as computers can contain toxic materials
- Computers use energy. This can be reduced by using modern screens for example.
- Computer systems can lead to more efficient manufacture.
- Computer systems can lead to less fuel being used for transport

Privacy Issues

- Many website required personal information. Is this information private and secure?
- Social media is an issue as lot of personal information and images are published
- Some feel their privacy is being invaded. Images and information being shared online.
- The more data that is store online and on mobile devices increases the risk of it being stolen.

Algorithms
Searching

- Standard searching algorithms: Linear and Binary

Specification — Revised

- **Linear Search** - Items are examined in order from the start of the list until the item is found.
- **Binary Search** - A type of divide and conquer algorithm where the list is repeatedly split/divided to make searching easier and faster as there are less items to look at

Linear Search	Binary Search
Steps: 1. Look at the first item 2. If this is the item you are looking for then stop as you have found it 3. If not then look at the next item in the list. Repeat steps 2 and 3 until you find your item or get to the end of the list	**Steps:** 1. Sort the list in order 2. Find the middle item in the ordered list 3. If this is the item then stop. 4. If not compare the item to the middle one. If it comes before (is less) than the middle item then get rid of the right side of the list. If it comes after (more than) the middle item get rid of the left side of the list You will have half the list left. Repeat steps 3 and 4 until you get your item
Example: *Given the following list of numbers and you want to find the number 7:* 2 | 0 | 1 | 7 | 4 | 3 | 5 | 6 *You will start at 2 and then move to the next items: 0, 1 until you get to 7*	***Example:*** *Given the following list of numbers and you want to find the number 96:* 6 | 23 | 45 | 55 | 67 | 90 | 92 | 96 | 99 *You start in the middle with the number 67. The number 96 is greater than 67 so you get rid of the left side of the list and you are left with:* 67 | 90 | 92 | 96 | 99 *You go to the middle item again which is 92. (6 is greater than 92 so you get rid of the left side of the list. You then are left with 96 and 99. You can either start with 96 or 99 and you will find your number with one more step.*

Linear vs Binary Search

- Linear search can be performed when the list is unsorted. With a binary search the list has to be sorted first
- A binary search is faster with large lists.

Algorithms
Sorting

- Standard sorting algorithms: Bubble, Merge and Insertion

Specification — Revised

ⓘ **Bubble Sort** - Compares pairs of items and parses through the list until they are in the correct order

ⓘ **Merge Sort** - Is an example of a divide and conquer algorithm. It splits the list into smaller lists and merges pairs of sub-lists together

ⓘ **Insertion Sort** - Takes each item in turn and places it in the correct place starting with the first item in the list.

Bubble Sort

Steps:

1. Look at the first 2 items in the list
2. If they are in the wrong order swap them
3. Move to the next pair (the 2nd and 3rd items) and repeat step 2
4. Repeat step 3 until you get to the end of the list
   Repeat steps 1-4 until there are no swaps left

ⓘ Not very efficient on large lists
ⓘ For a list of n items you need to do n – 1 passes to sort.

Example:

10	6	5	17	4	9	13	3
6	10	5	17	4	9	13	3
6	5	10	17	4	9	13	3
6	5	10	17	4	9	13	3
6	5	10	4	17	9	13	3
6	5	10	4	9	17	13	3
6	5	10	4	9	13	17	3
6	5	10	4	9	13	3	17

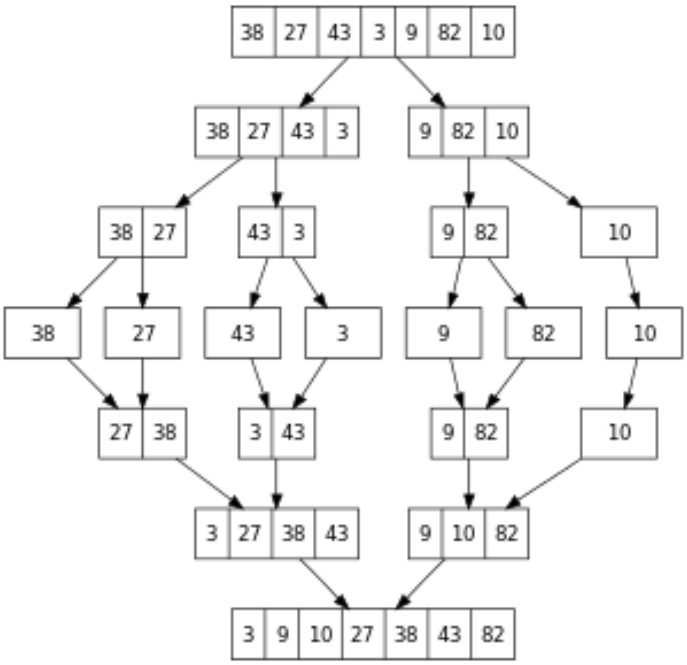
1. *Starting on the left the pairs of items are compared and swapped if needed (in blue).*
2. *We then move to the next pair and repeat until we get to the end of the list.* ***This is first pass****. The largest item will be at the end of the list after the first pass*
3. *Red indicated the item is sorted.*
4. *Steps 1 and 2 are repeated until all items are in order*

Merge Sort

Steps:

1. Split the list in half (the split list are called sub-lists).
2. Keep repeating step 1 on each sub-list until you get lists with 1 item only
3. Merge the pairs of sub-lists together sorting the items in the correct order.
4. Repeat step 3 until you merge all of the sub lists together

Example:



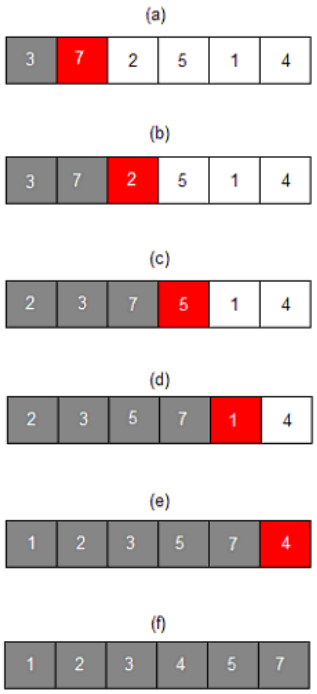
1. The list is split in half and this is repeated until you have sub-lists with only 1 item.
2. The pairs of sub-lists are then merged together in order.
3. This is repeated until they are all merged in order

Insertion Sort

Steps:

1. Look at the second item in the list
2. Compare it to all items before and insert the item in the correct place
   Repeat step 2 until you get the end by moving to the next number and placing it into the correct place.

Example:



1. Each item is taken in turn starting from the left.
2. The red item is taken and inserted into the correct position in the left side of the list.
3. You have 2 lists: The unsorted list to the right and the sorted list to the left of the item.
4. You repeat the steps until it is sorted

Bubble vs Insertion vs Merge Sort

- ⓘ Insertion sort uses less memory than a merge sort.
- ⓘ Bubble sort is the slowest out of the 3
- ⓘ Merge sort is the fastest on large lists out of the 3
- ⓘ Bubble sort is a simple algorithm that can be implemented and is efficient to check if a list is already in order.

Topic 2.2

Algorithms and Programming Techniques

Pseudocode

- How to produce Algorithms using Pseudocode and Flowcharts
- Interpret, correct or complete algorithms
- the use of variables, constants, operators, inputs, outputs and assignments
- the use of the three basic programming constructs used to control the flow of a program: sequence selection iteration (count and condition controlled loops)
- the use of data types: integer real o Boolean character and string
- Casting
- The common arithmetic and Boolean operators

Specification — Revised

ⓘ **Variable** = a named memory location which stores a value. The value stored in this space can be used and changed while the program is running. There is usually no value set when declared.

ⓘ **Constant** = a named memory location which stores a value that will not change in the program. Its value is set when declared.

Sequence	Selection	Iteration
Following a set of instructions in order	The path the program takes based on a condition being met. For example, using the IF-ELSE statements.	Repeating a set of instructions. Iteration is another name for a loop

Data Types

- **Integer** — Whole number
- **Real** — A number with a decimal fraction
- **Boolean** — True or False
- **Character** — A single alphanumeric character
- **String** — A series of alphanumeric characters

Mathematical Operators		Comparison Operators	
+	Add	==	Equal to
-	Subtract	>	Greater than
*	Multiple	<	Less than
/	Divide	<> or !=	Is not equal to
MOD	Modulus. Returns the remainder	>=	Greater or equal to
DIV	Returns the integer value from division	<=	Less than or equal to
^	Exponentiation. To the power of		

PROGRAMMING TECHNIQUE	PSEUDOCODE EXAMPLE
INPUT	name = input ("What is your name")
OUTPUT/PRINT	output (name) *or joining a string and variable together* output ("your name is:" + name)

SELECTION
 	
IF..ELSE
	if entry == "a" then
	  print("You selected A")
	elseif entry=="b" then
	  print("You selected B")
	
	else
	  print("Unrecognised selection")
	endif

ITERATION FOR LOOP	for i = 1 to 100: output (i)

ITERATION
 	
WHILE LOOP
	count = 0
	while count < 100:
	  output(count)
	  count = count + 1

CASTING (CHANGING ONE DATA TYPE TO ANOTHER)	str(cost) int(score) e.g. str(3) returns "3" int("3") returns 3

Algorithms and Programming Techniques

Pseudocode

- the use of basic string manipulation
- the use of basic file handling operations:
  - open
  - read
  - write
  - close
- the use of records to store data
- the use of SQL to search for data
- the use of arrays (or equivalent) when solving problems, including both one and two dimensional arrays
- how to use sub programs (functions and procedures) to produce structured code

Specification — Revised

STRING HANDLING	
LENGTH OF A STRING	`stringname.length`
GETTING A SUB STRING (PART OF THE STRING)	`stringname.substring(start_position, number_of_characters`
CHANGE STRING TO UPPER CASE	`stringname.upper`
CHANGE STRING TO LOWER CASE	`stringname.lower`

FILE HANDLING	
OPENING A FILE	`myFile = open("computing.txt")`

READING FROM A FILE	`myFile = openRead("computing.txt")`
	
	`while NOT myFile.endOfFile()`
	`  print(myFile.readLine())`
	`endwhile`
	`myFile.close()`

WRITING TO A FILE	`myFile = openWrite("sample.txt")`
	`myFile.writeLine("Hello World")`
	`myFile.close()`

ARRAYS/LISTS	
CREATING A LIST	names = ["Bob", "Alex", "Chris"]
GETTING AN ITEM FROM A LIST	names[1] *this would output Alex*
LOOPING THROUGH A LIST	for i in name: output names[i]
LENGTH OF A LIST	Output (len(names)
USING A LOOP TO COUNT THE VALUE OF ITEMS IN A LIST	scores = [5, 12, 5, 7, 8] total = 0 for i in scores: total = total + i

FUNCTIONS	
FUNCTION RETURNS A VALUE	function triple(number) return number*3 endfunction *e.g. to call funtion* *y=triple(7)*
PROCEDURE DOES NOT RETURN A VALUE	procedure greeting(name) print("hello"+name) endprocedure e.g. to call procedure greeting("Hamish")

SQL	
SELECT	
SELECT * FROM Customers	This selects everything (*) from the Customers database.
SELECT *ContactName,Address* FROM *Customers* WHERE *ContactName = Mr Smith;*	This selects the ContactName and Address columns from the Customers table and then specifically looks for a Mr Creosote in the ContactName field.
SELECT *ContactName,Address* FROM *Customer* WHERE ContactName LIKE Smi*;	This selects the ContactName and Address columns from the Customers table and then looks for a something LIKE Smi* in the ContactName field. This is a more open search and will return any value that is like the pattern provided. You can also use these operators:

=	Equal
<>	Not equal (!= sometimes)
>	Greater than
<	Less than
>=	Greater than or equal
<=	Less than or equal
BETWEEN	Between an inclusive range
LIKE	Searcher for a pattern
IN	Specify multiple values

SELECT * FROM Customers WHERE Country = 'England' AND (City = 'Camelot' OR City = 'Palermo');	You can also use Boolean operators (AND, OR) to refine a search and these can be combined using brackets.

Producing Robust Programs

Defensive Design

- Defensive Design Considerations:
  - Input sanitisation/validation
  - Planning for contingencies
  - Anticipating misuses
  - Authentication

Specification — Revised

- Producing a Robust program means that it is able to deal with errors and unexpected circumstances so that it can function correctly.
- We can try and achieve this through defensive design.

Input Validation	Input Sanitisation
Checking if data is sensible and reasonable (does not check if it is correct) Checks it meets given criteria e.g. text must be 8 characters long	Cleans the data of any unwanted characters before the data is entered. It may remove spaces or symbols from the text before it is input Ensures data entered contains only the permitted characters

Types of Validation Checks

Check digit	the last one or two digits in a code are used to check the other digits are correct	Bar code readers in supermarkets use check digits
Format check	checks the data is in the right format	For example a date of birth
Length check	checks the data isn't too short or too long	A password which needs to be 8 letters long
Lookup table	looks up acceptable values in a table	For example gender
Presence check	checks that data has been entered into a field	E.g. you must enter your name when filling out a form
Range check	checks that a value falls within the specified range	E.g. Number of days in month is 1-31

- **Contingency Plan** = We can anticipate how data could be misused or errors that may occur and we can plan for these.
- **Authentication** = check that the user is who they say they are. E.g. authenticate with a user name and password or using a 2 step authentication when a code is generated (using a key device or a message sent) which must be entered to confirm identity. You could also limit the number of login attempts.

Producing Robust Programs

Maintainability and Errors

- Maintainability: Comments and Indentation
- Purpose of Testing
- Types of testing: iterative and final
- How to identify syntax and logic errors
- Selecting and using suitable test data

Specification — Revised

ⓘ Once a program has taken into consideration defensive design it will need to maintain. Easy ways to maintain a program are:

Maintainability = making a program easier to understand and make changes to

Indentation	ⓘ Groups blocks of code together making is easy to read and pick out features ⓘ Blocks of code can be found easily in the future to make changes
Comments	ⓘ Explain the purpose of parts of the code and how they work ⓘ Makes it easier for someone else to understand your code and make changes
Sensible Variable and Sub routine name	ⓘ Makes the program easier to understand when the names used are meaningful and can be followed in the program

Syntax Error	Logic Error
Errors in the rule of the language. Examples: ⓘ Not declaring variables before use ⓘ Not indenting a line of code, missing a colon or bracket. ⓘ Variables not spelt the same as where they are declared	The program may still run but not work as intended. Examples: ⓘ Incorrect output ⓘ Infinite loops ⓘ Incorrect expressions ⓘ Conditions not being met

Producing Robust Programs

Testing

- Purpose of Testing
- Types of testing: iterative and final
- Selecting and using suitable test data

Specification — Revised

ⓘ **Testing** = checking that the program works and in the way it was intended to

Iterative Testing	Final Testing
ⓘ Testing the code as you create it ⓘ Fixing and making changes as a result ⓘ Going through this cycle until it meets the requirements	ⓘ Carried out at the end once it has been written ⓘ Checks that they requirements have been met ⓘ Fix any errors

Test Data = Data that is entered when performing the test
Valid/Normal = Test data that represents typical data that should be accepted
Valid Extreme = Test data that is valid but at the end of the range
Invalid = Test data that is outside of the suitable range and should be rejected
Invalid Extreme = Test data that is just outside the valid range
Erroneous: Test data that should be rejected because it is the wrong type

Test Plan = an outline of what is going to be tested. It should cover at least the following:

- ⓘ Purpose of the test
- ⓘ Test data
- ⓘ Expected Result

For example to test the input values in a range where only whole numbers between 1-10 are accepted the following types of test could be carried out.

Test No	What is Being Tested	Test Data	Expected
1	Normal values inside the range are accepted	5 Valid	Value is accepted
2	Values outside the range are rejected	12 Invalid	Value is rejected
3	The last value in the range is accepted	10 Valid Extreme	Value is accepted
4	The values just outside the range are rejected	11 Invalid Extreme	Value is rejected
5	Data of the wrong type is rejected	A Erroneous	Value is rejected

Computational Logic

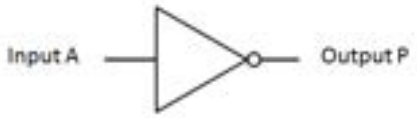
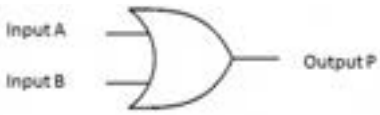
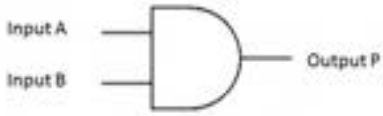
Logic Gates

- Why data is represented in a computer system in binary form
- Simple logic diagrams using the operations AND, OR and NOT
- Truth Table
- Combining Boolean operators using AND, OR and NOT to two levels
- Applying logical operators in appropriate truth tables to solve problems

Specification — Revised

- A binary number system means that only two digits can be used. These two digits are 0 and 1.
- Using only 0 and 1 makes it easier to design the electronic circuits that the computers will use.
- Memory and circuits in a computer are made by wiring millions of transistors together. They can make simple logic calculations such as are both inputs a 1. These simple circuits are called logic gates.

There are 3 main types of gates:

NOT	OR	AND
		
The NOT gate is a very simple gate – if 0 is input the it outputs 1 and if 1 is input then it outputs 0.	The OR gate tells us if one or both of the two inputs are 1 by outputting 1, otherwise output 0	The AND gate tells us if both inputs are 1, by outputting 1, otherwise it outputs 0.

Truth Tables

- We express this relationship between inputs and output as a truth table.
- We use A,B, C…for the inputs and P, Q, R … as outputs.
- Below are the truth tables for the NOT, AND and OR gates

NOT

A	P
0	1
1	0

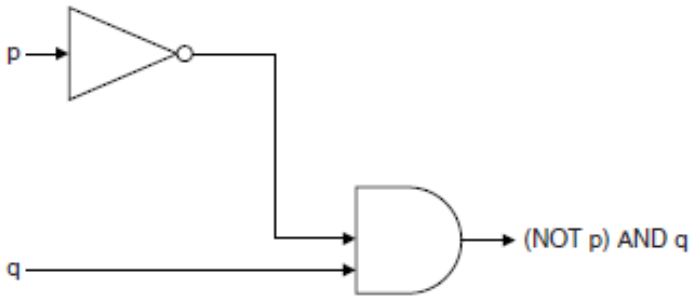
OR

A	B	P
0	0	0
0	1	1
1	0	1
1	1	1

AND

A	B	P
0	0	0
0	1	0
1	0	0
1	1	1

Gates can be combined together to form more complex gates. For example:



Here. We have joined together a NOT gate and an AND gate. The completed truth table would be:

p	q	(NOT p) AND q
0	0	0
1	0	0
0	1	1
1	1	0

You need to first work out the input p which is NOT p. This is in brackets. You do the brackets first

You then combine the result of this with q and show the result of going through the AND gate

Applying Computing Related Mathematics:

Operator	Purpose
+	Addition
-	Subtraction
/	Division
*	Multiplication
Exponentiation(^)	To the power of. E.g. 10^2
MOD	Gives the remainder after division. E.g. 5 MOD 2 = 1
DIV	Gives the integer result of the division. E.g. 5 DIV 2 = 2

Translators and Facilities of Languages

Types of Languages

- Characteristics and purpose of different levels of programming language, including low-level languages

Specification — Revised

- Machine code is code that is in binary. It is the code the CPU uses when it decodes an instruction and it is specific to a processor.
- Assembly code uses mnemonics to represent the binary instructions.
- High level code is understandable to a human being and look more like general English

Machine Code	Assembly Code	High Level Code
Low Level Code Is what the computer understands (Binary)	Low Level Uses mnemonics (abbreviations) to represent instructions such as STA for store and LDA to load Commonly used for embedded computers	Uses commands that are close to structured English *Examples include: Python, Java, C++, JavaScript, C#*
Processor Specific	Processor Specific	Portable to different devices
Can run straight away and does not need to be translated	Needs to be translated to machine code	Needs to be translated to machine code

	Low Level	High Level
Advantages	Uses less memory Can run faster Can manipulate hardware components directly	Easy to learn and understand Easier to find errors/debug More portable to different machines
Disadvantages	Harder to learn Hard to find errors/debug Specific to one machine	Will run slower Takes more memory and slower to run

Translators and Facilities of Languages

Translators

- The purpose of translators
- Characteristics of an assembler, compiler and an Interpreter

Specification — Revised

ⓘ Computers do not understand anything other than machine code. So programs that are written in Assembly or High Level Languages need to be translated into machine code

ⓘ A **translator** is used to translated high level or assembly code into machine code

High level code Machine Code → is translated by a **Compiler** or an **Interpreter**

Assembly Code to Machine Code → is translated by an **Assembler**

Translator	Advantages	Disadvantages
Assembler Convert assembly language into machine code.		
Compiler Converts the ***whole code*** *into* ***object code*** *(an executable version of the program)* before running it. The object code runs independently from the source code and compiler. **Used in distribution.**	ⓘ Customers cannot see the source code when you distribute the program ⓘ Code runs quickly when compiled.	ⓘ Slow to compile ⓘ Can be more difficult to debug as errors are generated at once for all the code
Interpreter Converts the ***code one instruction at a time,*** and executes the instruction before moving to the next. It does not create object code and the source code is run each time. **Mainly used in development.**	ⓘ Easy to debug code ⓘ Code can be developed and tested in stages	ⓘ Interpreter is needed on the target machine ⓘ Slower run time because the program is translated every time it is run.

Object Code = *A translated version of the program (source code) that can be run without the source code*

ⓘ *Web pages are an example of where code is interpreted each time the page loads. The Browser will have an interpreter and run the script that is embedded into the page such as JavaScript.*

Translators and Facilities of Languages

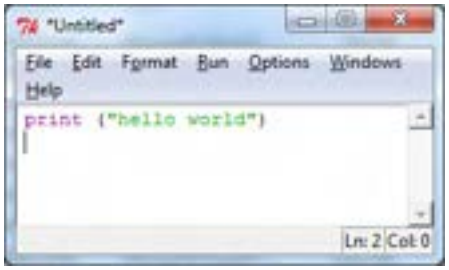
The IDE

- common tools and facilities available in an integrated development environment (IDE):
  - Editors
  - Error diagnostics
  - Run-time environment
  - Translators.

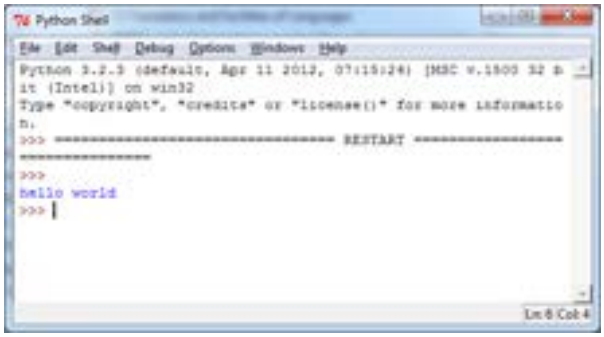
Specification — Revised

ⓘ The Integrated Development Environment (IDE) has tools and facilities that try to and make it easier to write and debug programs

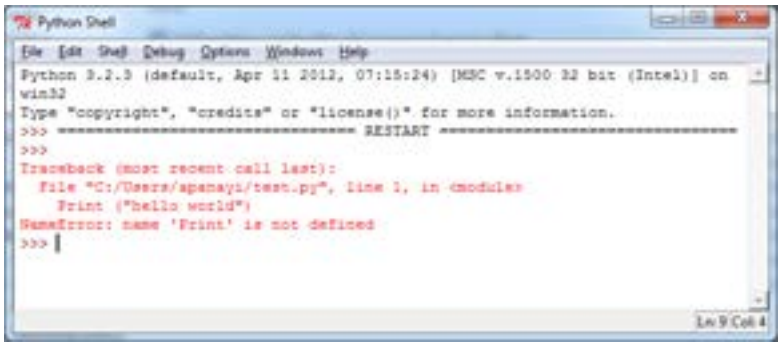
1. An **editor** which allows you to enter and edit the code. This can have features to make it easier to write your code by using colour coding and automatically indenting your code



2. **Run time environment**. Allows the code to be run as it is being developed



3. **Translator.** May have a compiler or an interpreter or both to translate the high level code to machine code.
4. **Error Diagnostics**. These identify errors in code and potential problems



Representation of Data

Units

- Bit, nibble, byte, kilobyte, megabyte, gigabyte, terabyte, petabyte
- How data needs to be converted into a binary format to be processed by a computer.

Specification — Revised

Binary is a number system that uses just two symbols, **0** and **1**.

Computers do not understand anything other than machine code. So data and instructions need to be converted and stored in **binary**

- A single zero or one is a **Bit**
  - **1 or 0**
- 4 zeros and/or ones is a **nibble**
  - **1010**
- 8 zeros and/or ones is a **Byte**
  - **10101010**

Common description of bits/bytes	No of bits/bytes
a 1 or a 0 (b)	1 bit
1 nibble	4 bits
1 byte (B)	8 bits
1 Kilobyte (KB)	1024 bytes (1024 x 8 bits)
1 Megabyte (MB)	1024 Kilobytes
1 Gigabyte (GB)	1024 Megabytes
1 Terabyte (TB)	1024 Gigabytes
1 Petabyte	1024 Terabytes

ⓘ *You need to be able to calculate how many bits/bytes in a particular measurement. E.g. How many megabytes in 2 Gigabytes? (2 x 1024 = 2048)*

Representation of Data

Numbers - Binary

- How to convert positive denary whole numbers (0-255) into 8 bit binary numbers and vice versa

Specification — Revised

- The **binary** number system uses multiple of two instead of ten (our base 10 number system) for column headings.
- The **denary** number system is our method of counting
- When converting **8 - bit binary** to denary and vice-versa draw a table like the one below with the headings.

128	64	32	16	8	4	2	1

Converting Binary to Denary

We place the binary number in the table and then add up the column values where there is a 1. E.g. convert the number 01010101 to denary:

128	64	32	16	8	4	2	1
0	1	0	1	0	1	0	1

= 64 + 16 + 4 + 1

Answer = 85

Converting Denary to Binary

Start on the left side of the table (largest number) and move along checking if the number is divisible by the column heading. If it is you place a 1 else you place a 0. E.g. convert 56 to binary

- Is the number divisible by 128? The answer is no so we put a 0.
- Is the number divisible by 64? The answer is no so we put a 0.
- Is the number divisible by 32? The answer is yes so we put a 1 and note the remainder which is 24.
- Is 24 (the remainder) divisible by 16? The answer is yes so we put a 1 and note the remainder which is 8.
- Is 8 (the remainder) divisible by 8? The answer is yes so we put a 1 and note the remainder which is 0.
- Is 0 divisible by 4? The answer is no so we put a 0 and note the remainder which is 0.
- We do the same thing for 2 and 1 and place a 0 in the column

128	64	32	16	8	4	2	1
0	0	1	1	1	1	0	0

So therefore 56 in binary is: 00111000

Representation of Data

Numbers-Adding Binary

- How to add two 8 bit binary integers and explain overflow errors which may occur

Specification — Revised

ⓘ Adding binary numbers uses the same method as base 10 (denary numbers). We add the values and the value is larger than that column we carry a number to the next column.

ⓘ There are **four rules** for adding binary:

1	**0 + 0 = 0** so we write 0
2	**0 + 1 = 1** so we write 1
3	**1 + 1 = 10** so we write 0 and carry 1
4	**1 + 1 + 1 = 11** so we write 1 and carry 1

- Here is an example:

Binary				
	0	1	0	0
	0	1	0	1
+	**1**	**0**	**0**	**1**
Carried values	1			

You can always check your answers by converting them to denary and checking them

- As we are using 8 bits there is a limit on the largest number that can be stored. **Overflow errors** can occur.
- Overflow occurs when we are adding 1 + 1 in the last column. The Carried 1 has nowhere to go and is lost. This causes the wrong answer

Binary								
	1	1	0	0	1	0	1	0
+	1	0	1	1	0	0	1	0
1	**1**	**1**	**1**	**1**	**1**	**1**	**0**	**0**
1						1		

We need a 9th bit otherwise the last bit is lost.

When adding binary numbers do not convert them into denary first, but you can check them at the end

Representation of Data

Numbers - Hexadecimal

- How to convert positive denary whole numbers (0-255) into 2 digit hexadecimal numbers and vice versa
- How to convert from binary to hexadecimal equivalents and vice versa

Specification

Revised

- Large binary numbers are difficult for programmers to remember and they want something that is easy to convert from binary.
- Each hex digit represents four binary digits exactly.
- This makes it a quicker way for programmers to write numbers.
- Hexadecimal numbers are based on base 16.
- They have 16 different digits: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F
- We use the letters A to F to represent the decimal numbers 10-15

Hexadecimal	Binary	Decimal
0	0000	0
1	0001	1
2	0010	2
3	0011	3
4	0100	4
5	0101	5
6	0110	6
7	0111	7
8	1000	8
9	1001	9
A	1010	10
B	1011	11
C	1100	12
D	1101	13
E	1110	14
F	1111	15

Converting Binary to Hexadecimal

- An 8 bit binary number is split into 2 nibbles.
- Each nibble can store numbers from 0 -15.
  - Example:

10011011

8	4	2	1
1	0	0	1

8	4	2	1
1	0	1	1

Converting Hexadecimal to Denary

- All we do is multiply the numbers by their place values and add them together. For example, take the hexadecimal number 4F.

Place Value	16	1
Hex Digits	3	F
Denary	= 3 x 16	= 15 x 1
	= 48	= 15

- 48 + 15 = 63
- So, 3F is 63 in denary.

Converting Denary to Hexadecimal

1. Covert your denary number into binary in the normal way
2. Split your 8 bit binary number into 2 nibbles
3. Convert each nibble into a hex digit

Or

1. Divide the number by 16 and write down the whole number this is the left most hex digit
2. The remainder is the next HEX digit and this is the digit to the right.

*e.g. 172 divided by 16 is 10 remainder 12. 10 is **A** and 12 is **C**. So the HEX digits for 172 are **AC***

Representation of Data

Shifts-Check Digits

- Binary Shifts
- Check Digits

Specification — Revised

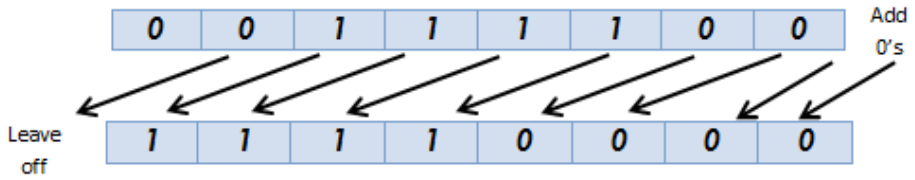
A binary Shift moves the Binary number a given number of places to the left or right

ⓘ Moving the binary number to the right will make it smaller (right shift). Moving it to the left will make it larger (left shift)

Example

Left Shift

Left Shift of 1 is equivalent to multiplying by 2. A shift of 2 is the same as multiplying by 4

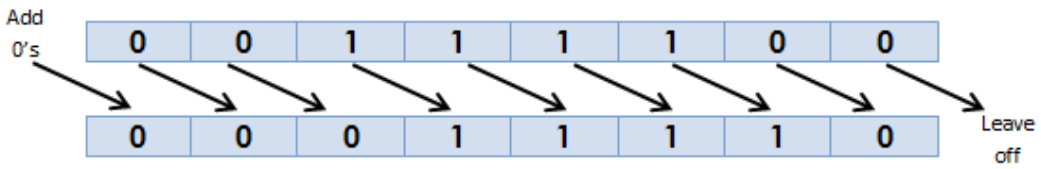


This is a left shift of 2. The first binary number is 60 and has shifted 2 places to become 240 (the equivalents of multiplying by4)

Bits are moved to the left the given number of places. Any gaps are filled with 0's

Right Shift

Right Shift of 1 is equivalent to dividing by 2. A shift of 2 is the same is dividing by 4



This is a right shift of 1. The first binary number is 60 and has shifted 1 place to become 30 (the equivalents of dividing by 2)

Bits are moved to the right the given number of places. Any bits that fall of are ignored and any gaps are filled with 0's. There may be a loss of accuracy when dividing as bits may be removed.

ⓘ A check digit is an additional digit at the end of a string of other numbers designed to check for mistakes in input or transmission. Calculated using modulo 10.

ⓘ The first 12 digits of the barcode are the unique item number, the 13th is the check digit

ISBN	5	0	1	4	0	1	6	1	5	0	8	2	1
Weight	1	3	1	3	1	3	1	3	1	3	1	3	
Multiplication	5	0	1	12	0	3	6	3	5	0	8	6	
Addition	Add all the numbers												49
Remainder	Find the remainder when divided by 10												9
Subtraction	Subtract the result from 10												1

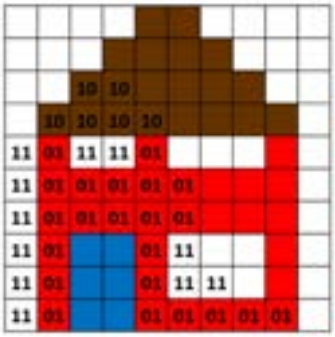
This is an example of how the check digit is calculated using the modulo-10 system. Each digit is multiplied by a weight. All numbers are added together and divided by 10. The remainder is subtracted from 10 to give the check digit

Representation of Data

Images

- How an image is represented as a series of pixels represented in binary
- Metadata included in the file
- The effect of colour depth and resolution on the size of an image file.

Specification — Revised



Bitmap Images are represented as a series of Pixels which each pixel stored as a binary code.

- More bits per pixel = more colour combinations
  - 1 bit = 2 Colours
  - **2 bits = 4 Colours**
  - 3 bits = 8 Colours
  - 4 bits = 16 Colours

Pixel	A **pixel (short for picture element) is one specific colour** Each pixel is stored as a binary value
Colour Depth	The number of bits used for each pixel determines how many colours we can use. This is known as the **colour depth.** The more bits per pixel the greater the colour depth and the more bits we need to store the image 4 bits can represent 16 colours, 8 bits can represent 256 colours,
Resolution	The resolution is the concentration of pixels. Usually measure in dots per inch (DPI). The higher the more pixels.
Metadata	Extra information stored about the image. Metadata includes data such as: *[The resolution, Width and height, Colour depth, Exposure, ISO, Aperture, File format]*

The effect of colour depth and resolution on the size of an image file	
A larger **colour depth** and **resolution** = higher quality image + larger file size	A smaller **colour depth** and **resolution** = lower quality image + smaller file size

Calculating the size of an image file
Size = resolution x colour depth (in bits) *To get the number of bytes divide the answer by 8*

Representation of Data

Characters

- The use of binary codes to represent characters
- The term 'character-set'
- The relationship between the number of bits per character in a character set and the number of characters which can be represented (for example ASCII, extended ASCII and Unicode).

Specification — Revised

Every time a character is typed on a keyboard a code number is transmitted to the computer. Characters are stored in binary by assigning a **unique binary code to represent each character**

Character Set	is the group of characters that can be coded
ASCII	Uses 7 bits so can represent 128 (2^7) characters
Extended ASCII	Uses 8 Bits so can represent 256 (2^8) characters
Unicode	Uses 16 bits so can represent 65,536 (2^{16}) characters

Problem with ASCII and Use of Unicode
With all the languages and characters/symbols in the world ASCII is not sufficient enough to represent all of these (only 128 or 256 characters). Unicode uses 16 bits and allow a much larger character set (65,536 or 2^{16}).

ASCII and Pure Binary
ⓘ Numbers as Characters have a different ASCII code compared to their actual (pure) binary representation. For example the character **1** has the ASCII code 0110001 which is 48 in Denary. The **pure binary** for 1 is 000001.
ⓘ You cannot do arithmetic on ASCII characters representing numbers – they must first be converted to pure binary numbers.

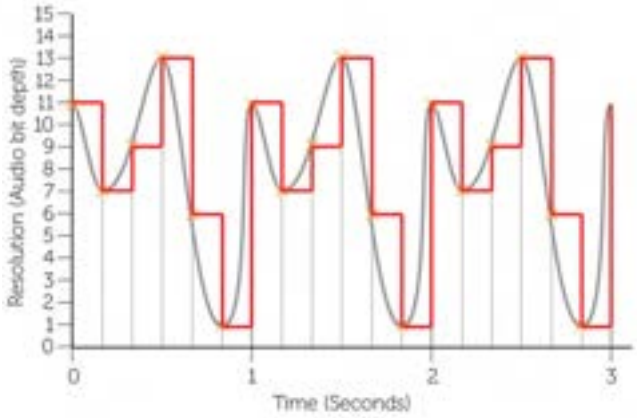
Representation of Data

Sound

- How sound can be sampled and stored in digital form
- How sampling intervals and other factors affect the size of a sound file and the quality of its playback:
  - Sample size
  - Bit rate
  - Sampling frequency.

Specification

Revised



Sampling – Measuring the amplitude (height of the wave) at regular intervals. Store each measurement in binary. The more bits used for the bit rate the more accurate the measurement

Bit Rate	The number of bits used to store each sample. – The more bits used the better the accuracy of the sound file.
Sampling Frequency	The time period between taking samples/measurements. – The more frequent the sound is sampled, the better the quality of playback
Sample Size	The length of the recording sampled

Size and Quality of the Sound File
Recording quality improves: – *the more frequently we sample the sound* – *the more accurately we record the wave height (larger bit rate)* *Increasing sampling rate and resolution means recording more data points which involved more data so the file size is therefore larger*

Size of a Sound File
Size = Bit Rate (bits) x Sample Frequency x sample size (duration in secs)

Representation of Data

Compression

- Need for compression
- Types of Compression:
  - Lossy
  - Lossless

Specification — Revised

Bitmap Images are represented as a series of **Pixels** which each pixel stored as a binary code.

- More bits per pixel = more colour combinations
  - 1 bit = 2 Colours
  - 2 bits = 4 Colours
  - 3 bits = 8 Colours
  - 4 bits = 16 Colours

Pixel	A **pixel (short for picture element) is one specific colour** Each pixel is stored as a binary value
Colour Depth	The number of bits used for each pixel determines how many colours we can use. This is known as the **colour depth.** The more bits per pixel the greater the colour depth and the more bits we need to store the image 4 bits can represent 16 colours, 8 bits can represent 256 colours,
Resolution	The resolution is the concentration of pixels. Usually measure in dots per inch (DPI). The higher the more pixels.
Metadata	Extra information stored about the image. Metadata includes data such as: *[The resolution, Width and height, Colour depth, Exposure, ISO, Aperture, File format]*

The effect of colour depth and resolution on the size of an image file	
A larger **colour depth** and **resolution** = higher quality image + larger file size	A smaller **colour depth** and **resolution** = lower quality image + smaller file size

Calculating the size of an image file
Size = resolution x colour depth (in bits) *To get the number of bytes divide the answer by 8*

OCR
Oxford Cambridge and RSA
An OCR endorsed teaching and learning tool

GCSE OCR

Computer Science
J277

Architecture of the CPU

Unit 1
Systems architecture

PG ONLINE

1

Objectives

- Understand the purpose of the CPU and the fetch-execute cycle
- Understand the following registers in the Von Neumann architecture:
  - MAR (Memory Address Register)
  - MDR (Memory Data Register)
  - Program Counter
  - Accumulator
- Understand common CPU components including:
  - ALU (Arithmetic Logic Unit)
  - CU (Control Unit)
  - Cache
  - Registers

Starter

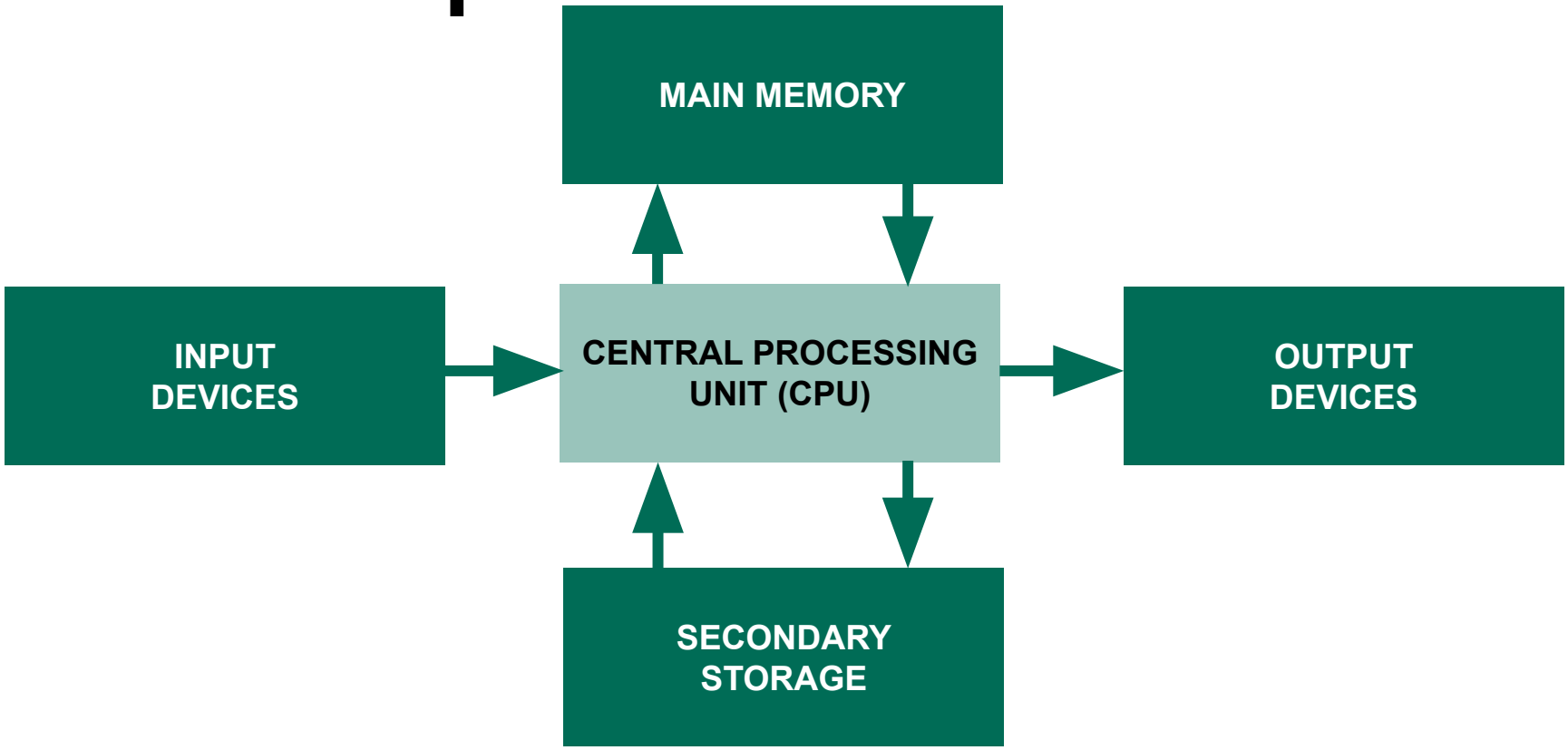
- A laptop and smartphone are both examples of computer systems
  - What are the input and output devices for these computer systems?
  - What component carries out the processing of the instructions?

Starter

Answers

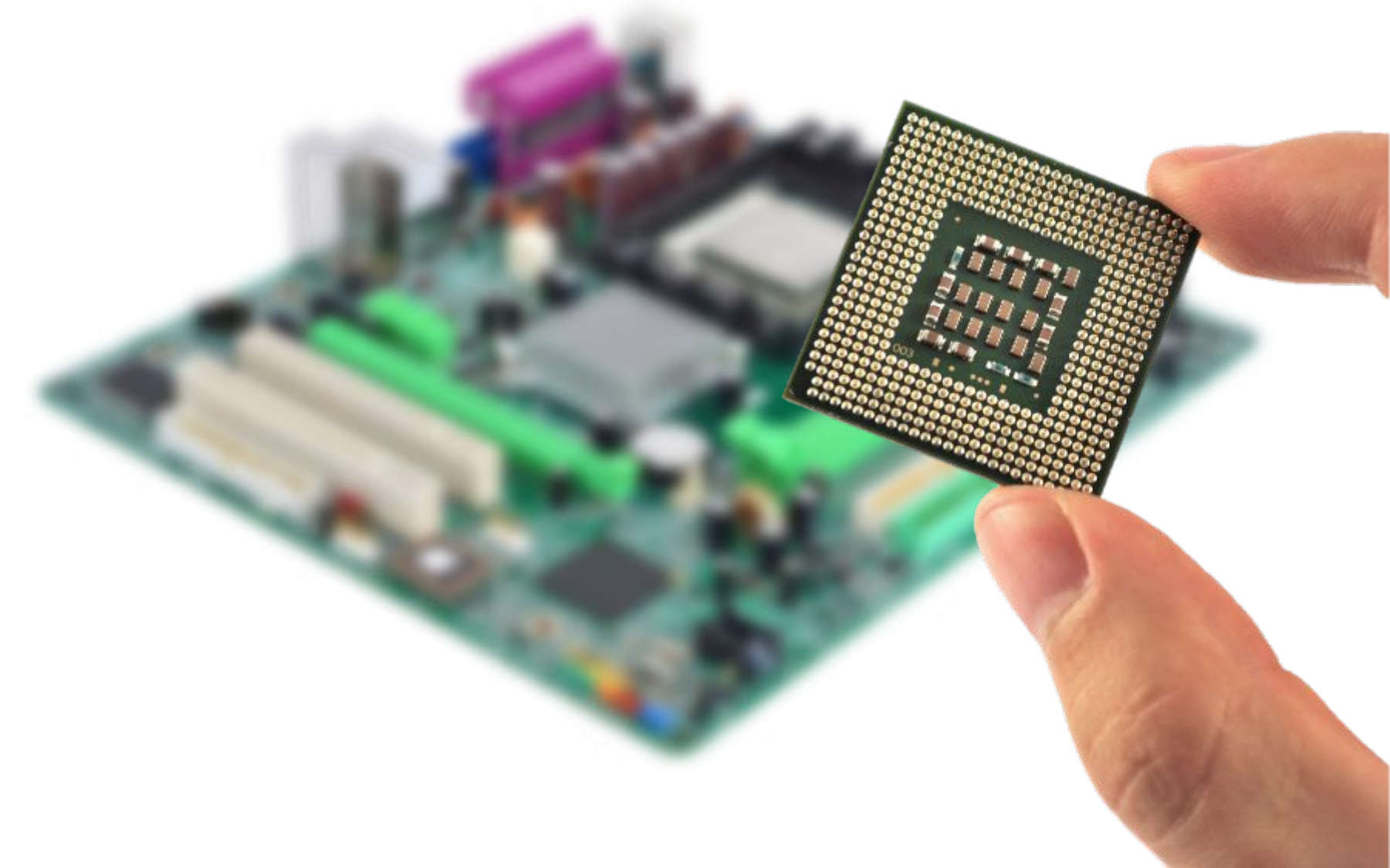
- Input devices:
  - Laptop: keyboard, buttons, trackpad, microphone
  - Smartphone: microphone, buttons, GPS sensor, gyroscopic sensor, touch part of the touchscreen
- Output devices:
  - Laptop and smartphone: speakers, display
- Processing:
  - CPU (Central Processing Unit)

The main components of a computer



What is this?

The CPU



Central Processing Unit (CPU)

- The Central Processing Unit or CPU is arguably the most important component of a computer
- What does it do?
  - What organ in the human body is it often compared to?
  - What are the similarities it has to this organ?

The purpose of the CPU

Answers

- The CPU processes instructions
  - When you run a program, it is the CPU which runs the instructions
- It is often thought of as being the ‘brains’ of the computer
  - The way that a brain works is very different to a CPU
  - A CPU runs one simple instruction at a time
  - It carries out billions of instructions per second

PG ONLINE

Stored program concept

- Before about 1943, early computers stored the data to be worked on in memory
- The program was not stored
  - Instructions were input one at a time using switches, or read in from punch cards and executed one at a time
- In 1943-44, mathematician von Neumann and his colleagues had the idea of storing the program instructions as well as the data in memory
- The **stored-program computer** was born!

Von Neumann architecture

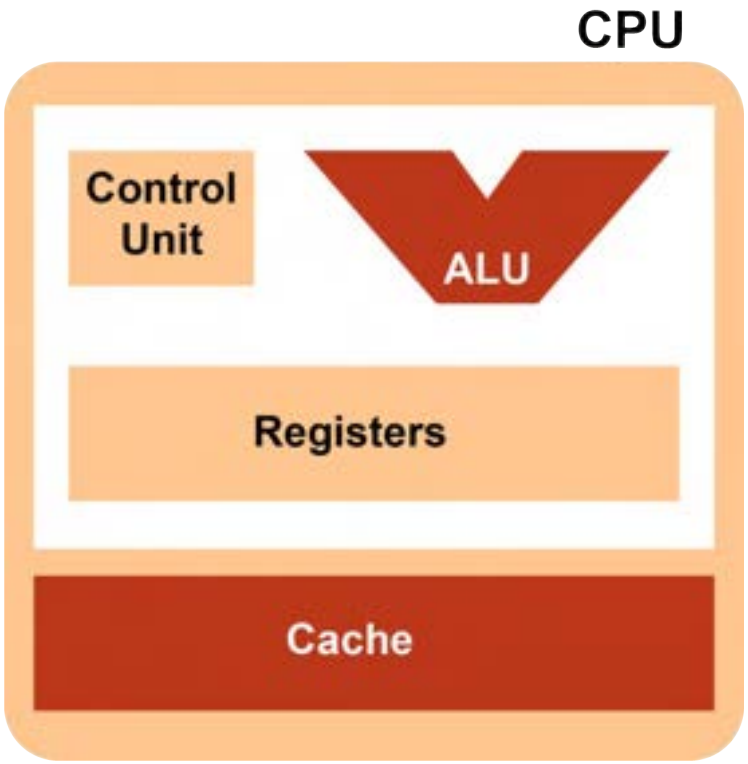
- Program **instructions** and the **data** the programs are using are both stored in the same memory
  - The CPU accesses both instructions and data from the same main memory (RAM or ROM)

Memory address	Instructions and data
0	Program instruction
1	Program instruction
2	Program instruction
3	Data
4	Data
5	Data
6	Data

Main memory

Components of the CPU

- Two of the major components of the CPU are the:
  - Control Unit
  - Arithmetic Logic Unit (ALU)
- There are also registers
  - These store data required for an instruction or the result of executing an instruction
  - A register is a very fast memory location in the CPU itself
  - Cache is located on the CPU – it is slower to access than registers but faster than RAM



Registers in the CPU

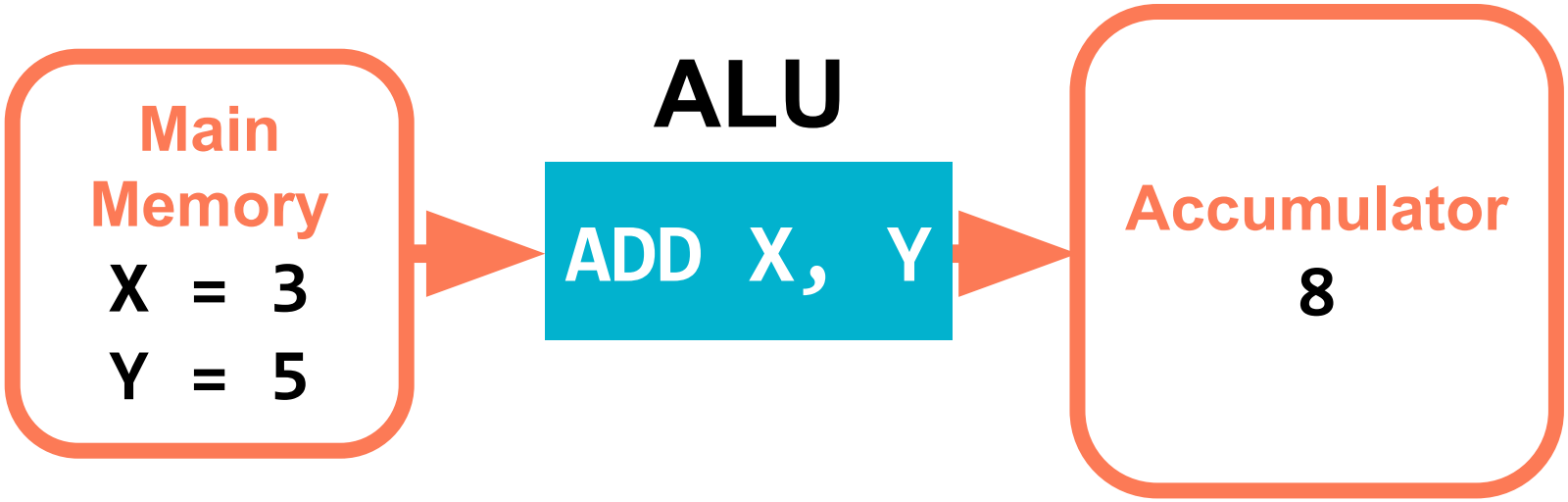
- **Program Counter (PC)**
  holds the address of the next instruction to be executed
- **Memory Address Register (MAR)**
  holds the memory address of the current instruction, and then the memory address of any data that the instruction uses, so that these can be fetched from memory
- **Memory Data Register (MDR)**
  holds the actual instruction fetched from memory, and then the data that has been fetched from, or written to memory
- **Accumulator**
  holds the results from the arithmetic logic unit after an instruction is executed before it is transferred to main memory

Control Unit

- The control unit coordinates and controls all of the activities taking place within the CPU
  - It decodes instructions and then sends signals to other components on the CPU so they know how to execute the instructions
  - It receives signals from the system clock
  - It controls the other parts of the CPU, much like the conductor of an orchestra

The Arithmetic Logic Unit

- The ALU or Arithmetic Logic Unit is where the actual arithmetic operations, such as ADD, are executed
  - It also carries out logical operations such as those including less than, greater than, equal to, AND, OR and NOT

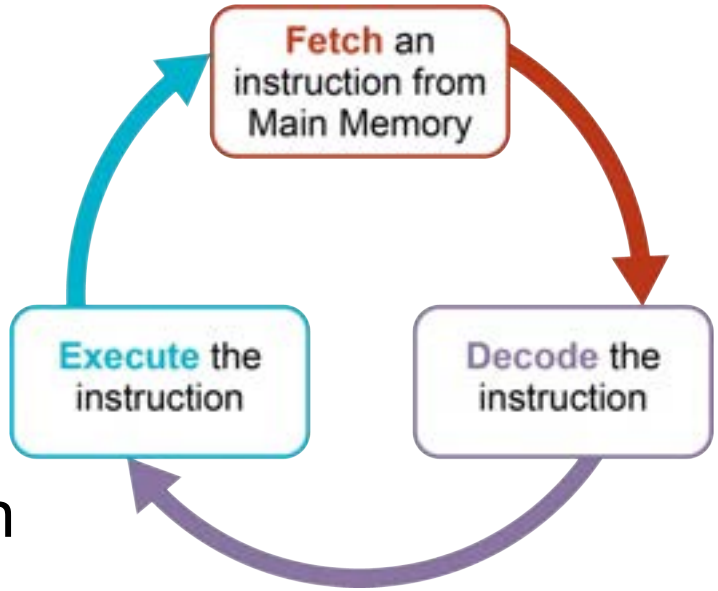


Worksheet 1

- Complete **Task 1** on **Worksheet 1**

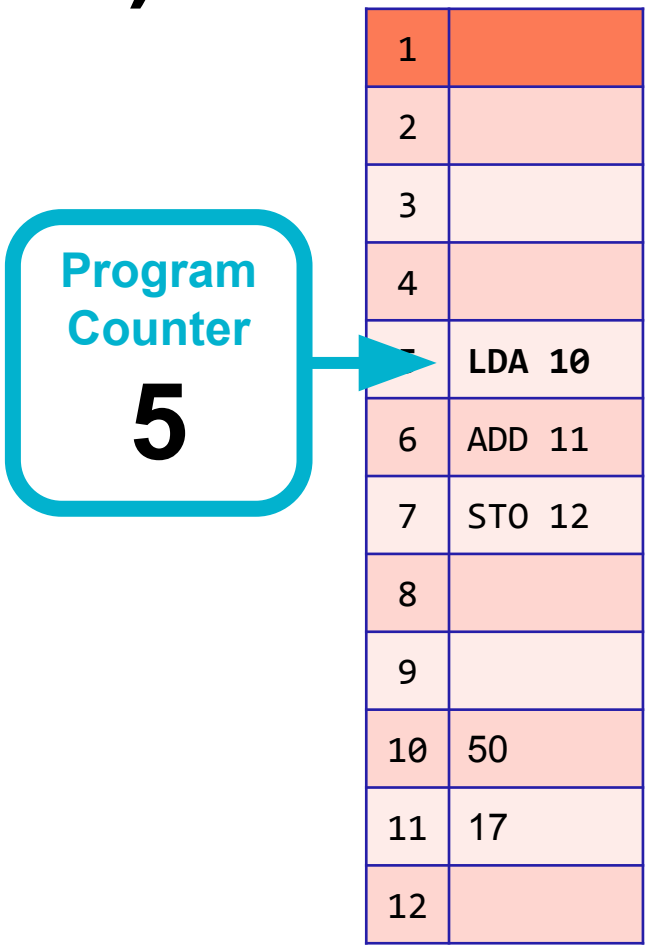
Fetch – Decode – Execute cycle

- The CPU operates by repeating three operations:
  - **FETCH** – causes the next instruction to be fetched from main memory
  - **DECODE** – decodes the instruction
  - **EXECUTE** – the instruction is executed
- This process is then repeated…



Program Counter (PC)

- The Program Counter holds the address of the next instruction to be executed
  - The Program Counter is incremented (increased by 1) as soon as that instruction has been fetched



Accumulator

- The accumulator (ACC) is where arithmetic and logic results are temporarily stored, much like the current result shown on a calculator

MAR and MDR

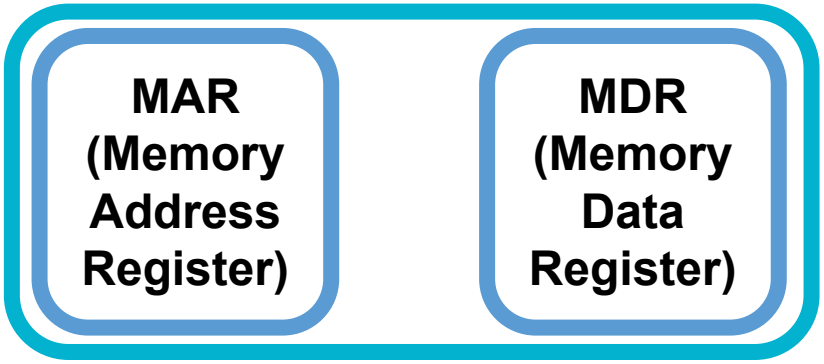
- In the FETCH stage of the F-E cycle
  - the address of the **instruction** to be executed is copied from the Program Counter (PC) to the Memory Address Register (MAR)
  - The **instruction** at that address is fetched from memory and copied to the Memory Data Register (MDR)
- The Control Unit decodes the instruction and decides if data needs to be fetched
- If so, the MAR is then used to hold the address of the **data** to be used in the instruction
- The **data** is fetched and copied to the MDR

Working together

MAR (Memory Address Register)

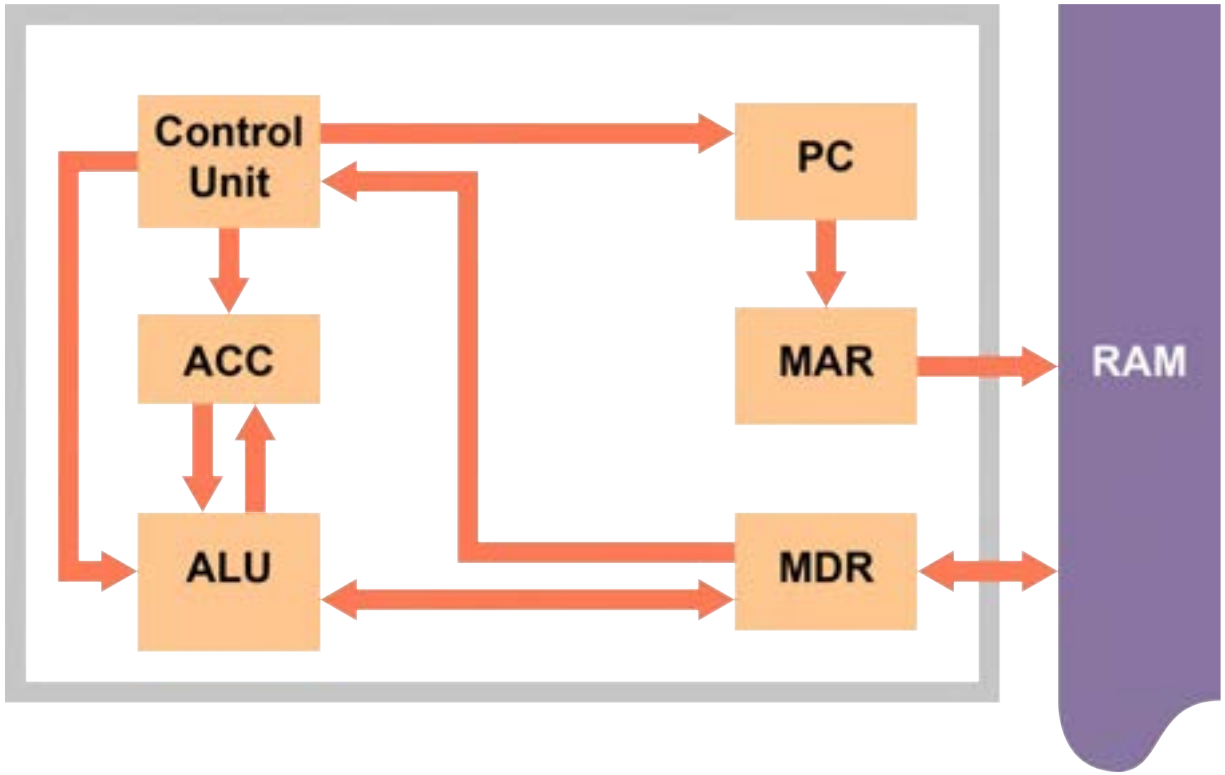
MDR (Memory Data Register)

The two work together; The MAR knows where to look for data in RAM, the MDR keeps hold of that data until it's ready to be used by the CPU



Processor architecture

- What does each acronym stand for?
  - What does each part do?



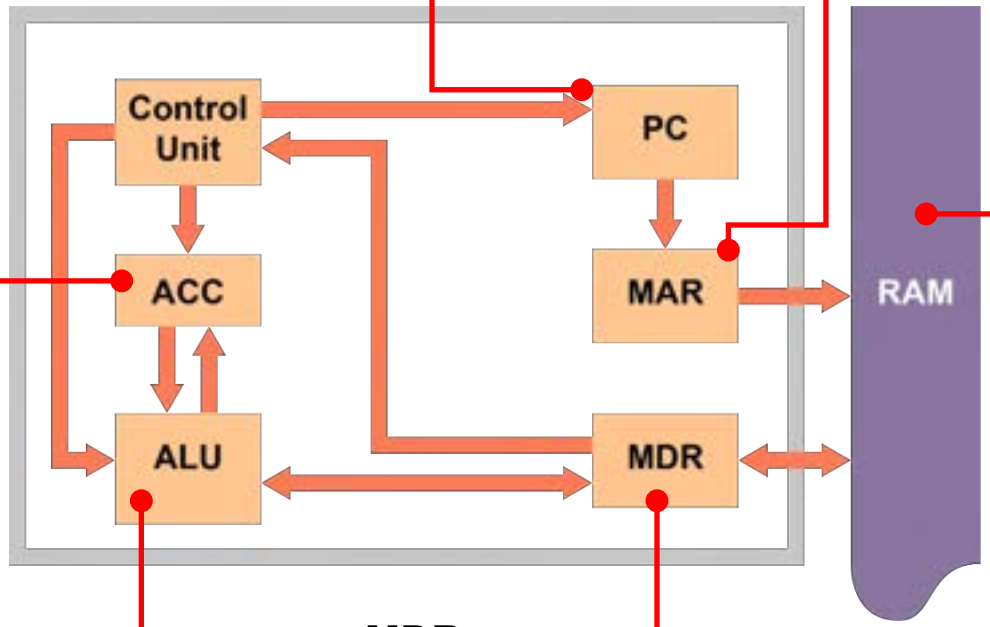
Processor architecture

Answers

PC:
Program Counter – points to the next instruction

MAR:
Memory Address Register – stores the address of the instruction to be executed or the address where data will be fetched from/written to

ACC:
Accumulator – Temporarily stores results from arithmetic or logical operations



RAM:
Random Access Memory

ALU:
Arithmetic Logic Unit – performs calculations and logic expressions

MDR:
Memory data register – Stores the data to be sent/received from memory

PG ONLINE

Example – Step 1 (Add 8 + 4)

1	
2	
3	
4	
5	LDA 10
6	ADD #4
7	STO 11
8	
9	
10	8
11	
12	

Program Counter
5

Acc
8

- PC points to next instruction in location 5
- Address 5 is passed to MAR and PC is incremented. It now holds 6
- Instruction at Address 5 is copied from memory to MDR
- Instruction is decoded by control unit
- Address 10 passed to MAR
- The value in location 10, i.e. 8, is passed from main memory to the MDR
- 8 is loaded into the accumulator (ACC)

Example – Step 2

1	
2	
3	
4	
5	LDA 10
6	**ADD #4**
7	STO 11
8	
9	
10	8
11	
12	

Program Counter
6

Acc
12

- PC now points to next instruction in location 6
- Address 6 is passed to the MAR and the PC is incremented to become 7
- Instruction at Address 6 passed to MDR
- The instruction is decoded
- No more data from memory is needed
- 4 is added to 8 by the ALU and the result is stored in the accumulator

Example – Step 3

1	
2	
3	
4	
5	LDA 10
6	ADD #4
7	STO 11
8	
9	
10	8
11	12
12	

Program Counter
7

Acc
12

- PC points to the next instruction in location 7
- Address 7 is passed to the MAR and the PC is incremented
- Instruction at Address 7 passed to MDR
- The instruction is decoded
- The instruction is executed causing the value 12 to be transferred from the accumulator to the MDR and the address 11 is transferred to the MAR
- The value 12 is stored in memory location 11

Worksheet 1

- Complete **Task 2** on **Worksheet 1**

Plenary

- In pairs, answer the following:
  - What is the purpose of the CPU?
  - What are **two** major components of the CPU, and what is the purpose of each?
  - What is a register?
  - Name **four** special registers in the CPU

Plenary

Answers

- What is the purpose of the CPU?
  - To fetch and execute instructions stored in memory
- What are **two** major components of the CPU, and what is the purpose of each?
  - The ALU, which carries out arithmetic and logic instructions
  - The Control Unit, which coordinates the CPU activities
- What is a register?
  - A fast memory location in the CPU
- Name **four** registers in the CPU
  - PC, MAR, MDR, Accumulator

Copyright

© 2024 PG Online Limited

The contents of this unit are protected by copyright.

This unit and all the worksheets, PowerPoint presentations, teaching guides and other associated files distributed with it are supplied to you by PG Online Limited under licence and may be used and copied by you only in accordance with the terms of the licence. Except as expressly permitted by the licence, no part of the materials distributed with this unit may be used, reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic or otherwise, without the prior written permission of PG Online Limited.

Licence agreement

This is a legal agreement between you, the end user, and PG Online Limited. This unit and all the worksheets, PowerPoint presentations, teaching guides and other associated files distributed with it is licensed, not sold, to you by PG Online Limited for use under the terms of the licence.

The materials distributed with this unit may be freely copied and used by members of a single institution on a single site only. You are not permitted to share in any way any of the materials or part of the materials with any third party, including users on another site or individuals who are members of a separate institution. You acknowledge that the materials must remain with you, the licencing institution, and no part of the materials may be transferred to another institution. You also agree not to procure, authorise, encourage, facilitate or enable any third party to reproduce these materials in whole or in part without the prior permission of PG Online Limited.

OCR
Oxford Cambridge and RSA
An OCR endorsed teaching and learning tool

GCSE OCR

Computer Science
J277

CPU performance

Unit 1
Systems architecture

PG ONLINE

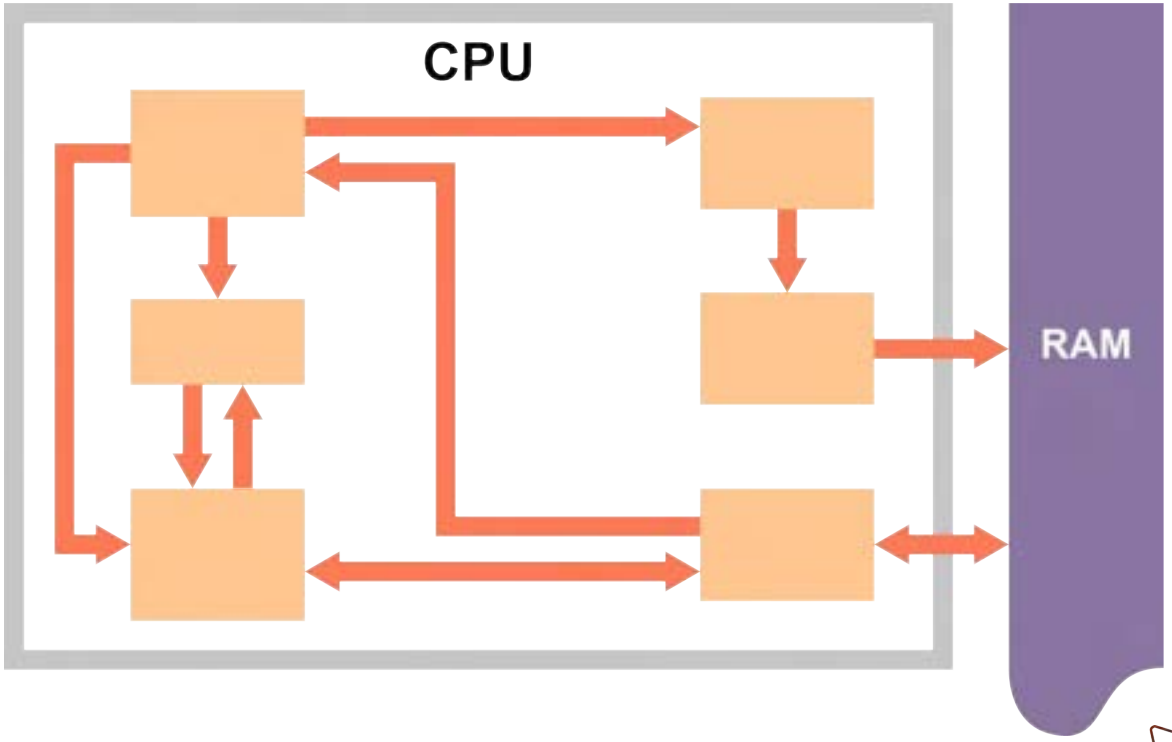
2

Objectives

- Understand the function of cache in the CPU
- Describe how common characteristics of CPUs affect their performance including:
  - Clock speed
  - Cache size
  - Number of cores
- Explain the purpose and give examples of embedded systems

Starter - recap

- Name at least **five** components inside a CPU
  - What is the purpose of each component?



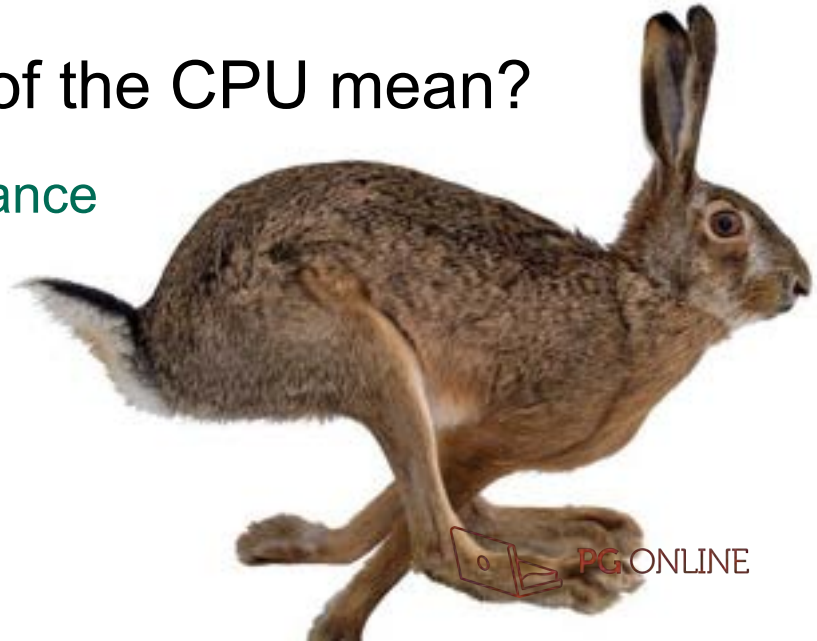
Starter

Answers

- Components in a CPU
  - **Control unit (CU)** – decodes the instructions; controls the timing of operations in the CPU
  - **Arithmetic Logic Unit (ALU)** – performs arithmetic and logic operations
  - **Registers** – including **PC** (program counter) for the address of the next instruction; **MAR** (memory address register) for the memory location of data to be fetched or written to main memory; **MDR** (memory data register) data retrieved from or written to main memory; **Accumulator** (ACC) stores the results from the ALU

What affects CPU performance?

- The following are three of the most common factors which can affect a CPU's performance
  - Clock speed
  - Processor cores
  - Cache
- What do each of these parts of the CPU mean?
  - How can they affect the performance of the CPU?



PG ONLINE

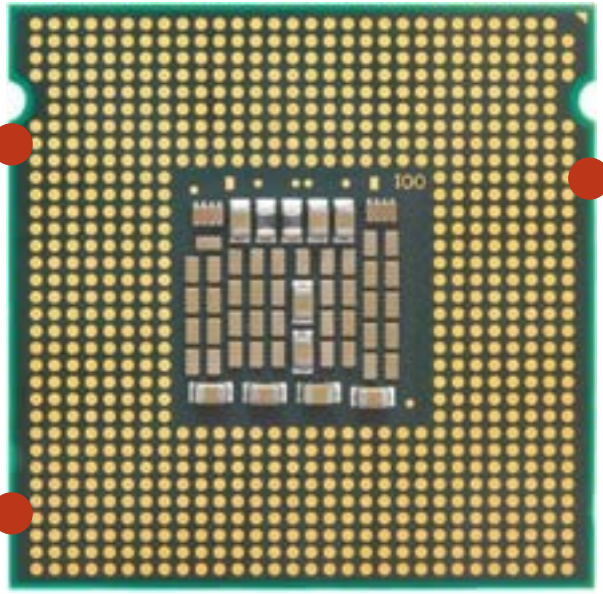
Factors affecting CPU performance

Clock speed

Cycles per second measured in hertz (Hz)

Number of cores

The number of duplicate processing units (cores) placed in one CPU



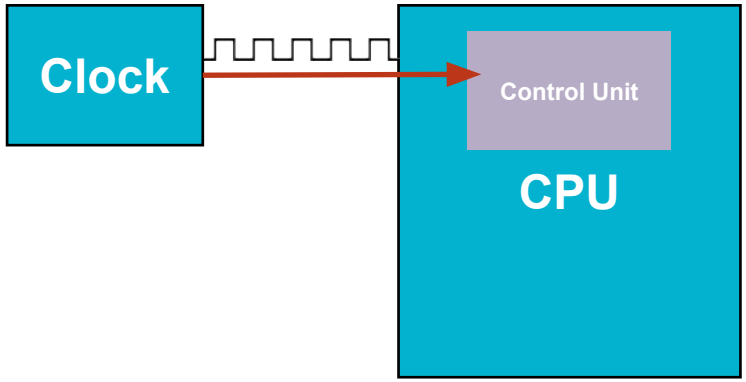
Cache size

Memory on the CPU that is faster than RAM but slower than registers

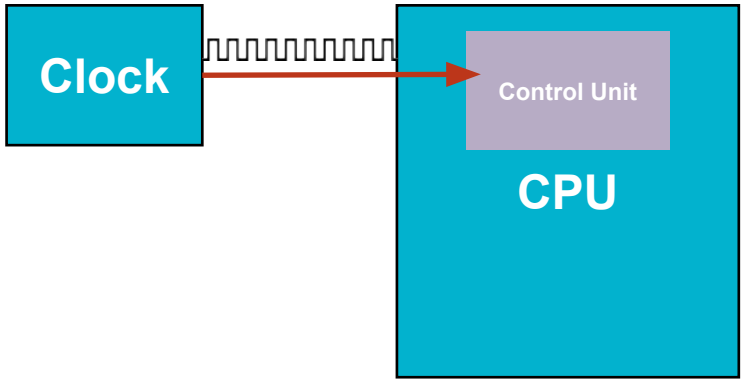
Clock speed

- Everything in a computer happens on the pulse of the internal clock
  - Therefore, the faster the clock speed, the faster the instructions are processed

Clock speed 1GHz



Clock speed 2GHz



Processor Speed

- One cycle per second = 1 hertz (Hz) = 1 instruction carried out each second
  - Some instructions may require more than 1 cycle to complete
- 1 kilohertz (kHz) = 1000 cycles per second
- 1 Megahertz (MHz) = 1,000,000 cycles per second
- 1 Gigahertz (GHz) = 1,000,000,000 cycles per second
  - How fast is your computer's processor?
  - Remember, a 1 GHz processor is performing one billion cycles per second

Multi-core processors

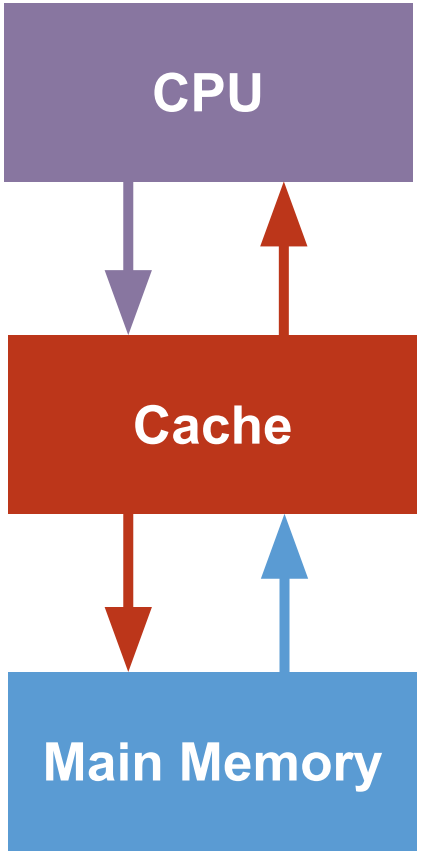
- A dual-core processor has two processors in the same integrated circuit, linked together
  - A dual-core processor has the potential to perform two instructions at the same time
  - This allows twice as many instructions to be executed, however, it doesn't always perform at this rate as software may not be able to take full advantage of both cores
- A quad-core processor has four linked processors
  - Which processor would you prefer, a dual-core 1GHz, or a 3GHz single core processor?

Effect on speed

- A quad-core processor working on many different tasks simultaneously, under ideal conditions can be up to four times faster than a single-core processor
  - If the computer is running a single program, it is not necessarily any faster, since the program may have been designed to only run on one core

Cache memory

- Cache is a small amount of very fast, expensive memory in the CPU
  - Cache can be accessed faster than regular main memory (RAM)
  - Recent or frequently used data and instructions are stored in cache
  - A larger cache will mean more instructions and data are stored in cache which increases the CPU performance as there is less need to access slower RAM or ROM



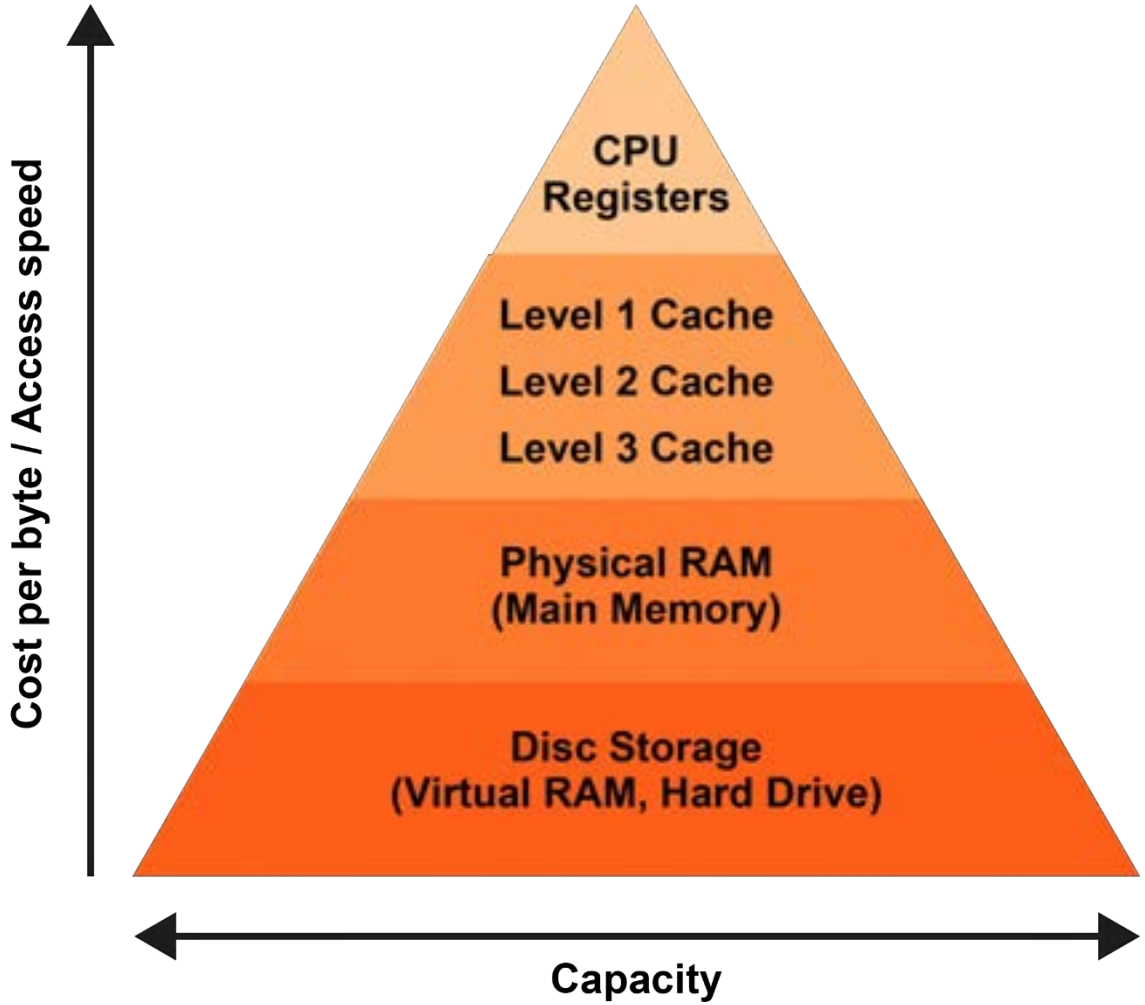
Effect on speed of cache

- RAM is fast to access, but is relatively slow memory to access compared to cache
  - It takes longer to retrieve an instruction or data from RAM than from cache
- Program instructions and data that are fetched are stored in cache in case they are needed again soon
  - If you have a ‘while’ loop in a program, for example, having all the instructions in the loop in cache speeds up execution

Levels of cache memory

- There are different ‘levels’ of cache:
  - **Level 1 cache** is extremely fast but small (between 2-256kB), located on the CPU. Each core will have its own level 1 cache
  - **Level 2 cache** is usually also given to each core. It is very fast, but a little slower than level 1 cache. The typical size is 256kB-8MB
  - **Level 3 cache** is the slowest type of cache, but still faster than RAM. It is usually located on the CPU and stores 4MB-50MB. The cache is shared between all the cores on the processor

Memory speeds by type



The benefits of cache

- The data used most often by the CPU is held in Level 1 cache so is available extremely quickly
  - In most systems, Level 1 cache is used about 50% of the time, with Level 2 cache being accessed about 90% of the time
  - This greatly reduces the time that the CPU has to wait for data from main memory
- The size of the Level 2 cache is a major factor in determining the performance of the CPU

Which is (theoretically) faster?

1. A dual core processor running at 2.7GHz with 4MB of shared cache memory
2. A quad core processor running at 1.5GHz with 8MB of shared cache memory
3. A single core processor running at 3.2GHz with 2MB of cache memory
4. A dual core processor running at 3.2GHz with 4MB of shared cache memory

Calculate the maximum number of cycles per second in each case and rank them in order of performance

Which is faster?

Answers

1. A dual core processor running at 2.7GHz with 4MB of shared cache memory
   2 * 2.7 = 5.4
2. A quad core processor running at 1.5GHz with 8MB of shared cache memory
   4 * 1.5 = 6
3. A single core processor running at 3.2GHz with 2MB of cache memory
   1 * 3.2 = 3.2
4. A dual core processor running at 3.2GHz with 4MB of shared cache memory
   2 * 3.2 = 6.4

Possible ranking: 4 (fastest), 2, 1, 3 (slowest)
2 may be faster than 4 due to the additional cache

Worksheet 2

- Complete **Task 1** on **Worksheet 2**

Embedded systems

- Do these devices have inputs, processes and outputs?
  - Could they be classed as computers?
  - Why or why not?

Embedded computers

- An embedded computer is a single microprocessor that includes RAM, ROM and a CPU
  - An embedded computer is frequently used to control a device using simple inputs

PG ONLINE

Embedded systems

- Without embedded systems, a digital device would not be able to perform specific functions
  - For example, a dishwasher wouldn’t know when to heat the water, or a satnav wouldn’t know how to communicate with a satellite
  - What embedded functions do cars use?

Car embedded systems

Answers

- Some examples of embedded systems in cars include:
  - Cruise control
  - Engine cooling fan
  - Interior fans and temperature
  - Interior lighting
  - Rear cameras and remote parking
  - Infotainment systems (entertainment / information / satnav)
  - Engine control unit (ECU)

Embedded vs general purpose

- The following table shows how the characteristics of embedded systems compare to general purpose machines

Characteristic	Embedded system	General purpose machine
CPU speed	Typically slow	Typically very fast
Software	Has one purpose and cannot install new software	New software can be installed
Storage	Programs stored on ROM	Programs stored on secondary storage, e.g. hard drive
Reliability	Typically very reliable – e.g. a microwave should not have a bug that changes defrost to full power	As it runs many software programs it may be less reliable and need restarting the device

Worksheet 2

- Complete **Task 2** on **Worksheet 2**

Plenary

1. Name **three** common factors which affect the performance of a CPU
2. What do each of these factors mean?
3. What is an embedded system?
4. Name **three** examples of an embedded system

Plenary

Answers

1. Clock speed, number of cores, cache size
2. **Clock speed** – the number of fetch-execute cycles per second; **number of cores** – the number of processing units on each CPU; **cache** – memory that is faster than RAM and slower than the registers, normally located on the CPU
3. A computer system that is used to control mechanical or electrical systems
4. Satnav, microwave, dishwasher, burglar alarm, washing machine, digital watch…

Copyright

© 2024 PG Online Limited

The contents of this unit are protected by copyright.

This unit and all the worksheets, PowerPoint presentations, teaching guides and other associated files distributed with it are supplied to you by PG Online Limited under licence and may be used and copied by you only in accordance with the terms of the licence. Except as expressly permitted by the licence, no part of the materials distributed with this unit may be used, reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic or otherwise, without the prior written permission of PG Online Limited.

Licence agreement

This is a legal agreement between you, the end user, and PG Online Limited. This unit and all the worksheets, PowerPoint presentations, teaching guides and other associated files distributed with it is licensed, not sold, to you by PG Online Limited for use under the terms of the licence.

The materials distributed with this unit may be freely copied and used by members of a single institution on a single site only. You are not permitted to share in any way any of the materials or part of the materials with any third party, including users on another site or individuals who are members of a separate institution. You acknowledge that the materials must remain with you, the licencing institution, and no part of the materials may be transferred to another institution. You also agree not to procure, authorise, encourage, facilitate or enable any third party to reproduce these materials in whole or in part without the prior permission of PG Online Limited.

OCR
Oxford Cambridge and RSA
An OCR endorsed teaching and learning tool

GCSE OCR

Computer Science
J277

Memory

Unit 1
Systems architecture

PG ONLINE

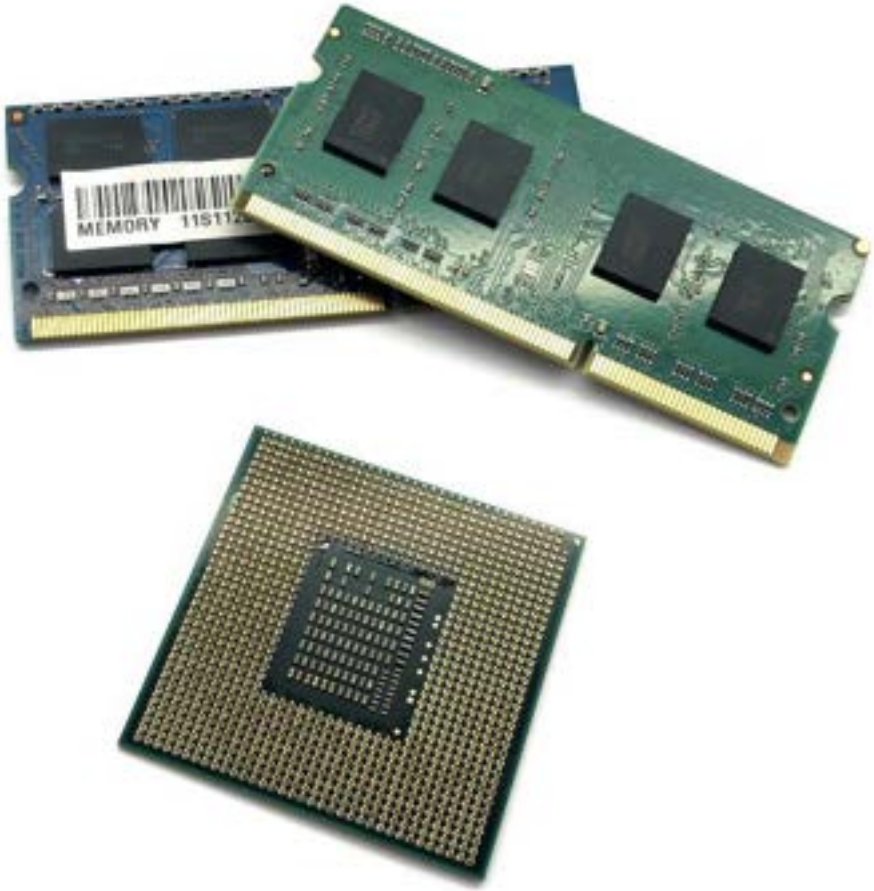
3

Objectives

- Explain the need for primary storage
- Describe the difference between RAM and ROM
- Describe the purpose of RAM and ROM in a computer system
- Explain the need for virtual memory

Starter

- What are **four** types of memory that are used in computers systems?



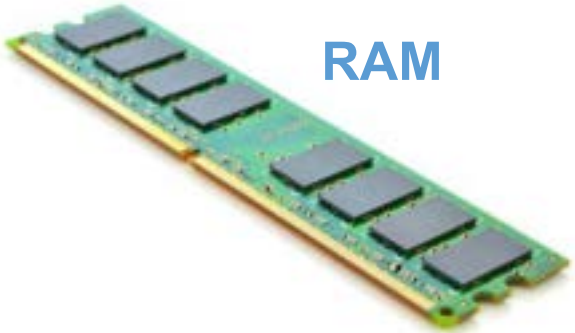
Starter

Answers

- From previous lessons:
  - CPU registers
  - Cache
- This lesson:
  - RAM
  - ROM
  - Virtual memory

Types of primary storage

- There are many types of memory used in computers
  - In the Von Neumann architecture, primary storage is needed to store programs that are currently running and need to be accessed by the CPU
- The two types used as primary storage are:
  - **RAM** (**R**andom **A**ccess **M**emory)
  - **ROM** (**R**ead **O**nly **M**emory)



RAM

ROM

RAM

- RAM is often referred to as:
  - main memory, primary memory, primary storage
- It can be read from and written to
  - Access to RAM is much faster than a hard drive
  - The name Random Access Memory comes from the ability of the CPU to access any part of the memory in the same amount of time
- At any one time it will normally store:
  - The operating system (or part currently in use)
  - The software currently in use
  - The data which the software is using

RAM

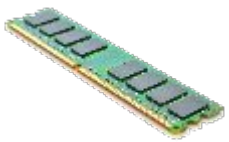
- The computer stores running programs and data in RAM when your computer is turned on
- When your computer is turned off, data stored in RAM is lost
  - RAM is volatile as it loses data if the power is off
- All your programs and data are stored permanently on secondary storage e.g. a hard drive
  - This is known as non-volatile storage as it doesn't lose the data if the power is off

Worksheet 3

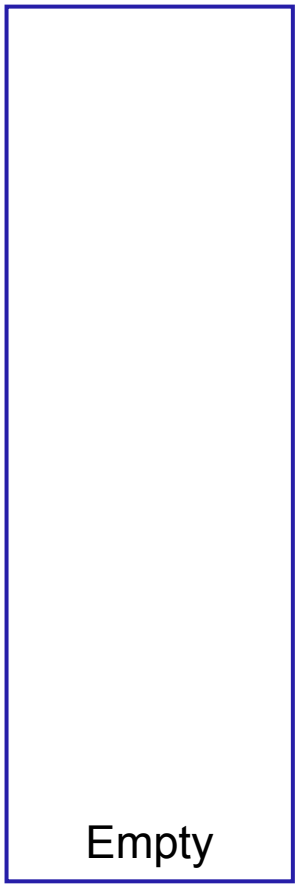
- Complete **Task 1** on **Worksheet 3**

RAM

- When the computer is first turned on, no data is stored in RAM

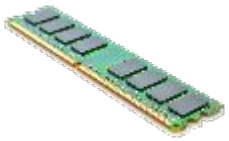


RAM

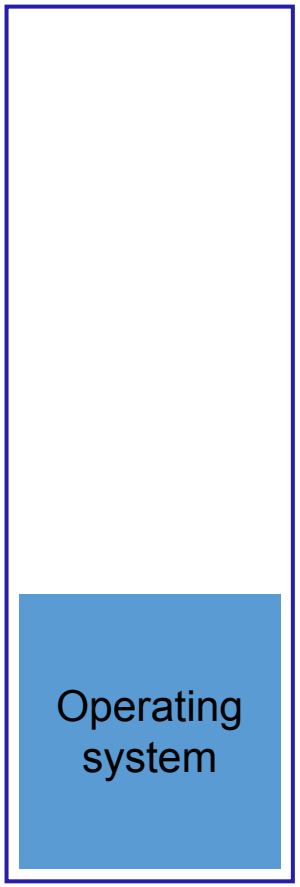


Operating system

- The computer first loads the operating system from secondary storage, such as a hard drive, into RAM

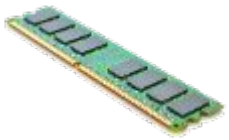


RAM

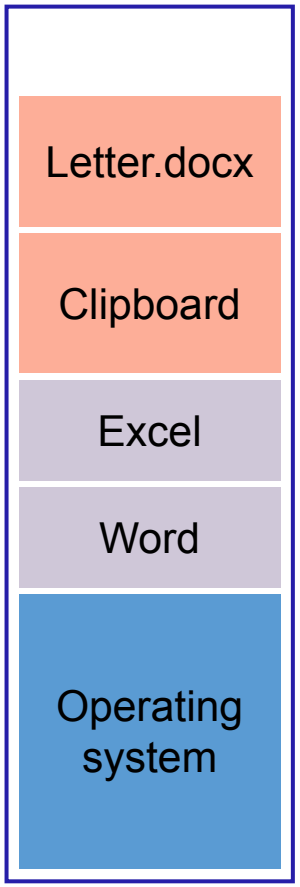


Programs and data

- When applications or programs are loaded, they are copied into RAM from the secondary storage
  - Documents and files (data) that are used with those programs are also opened by copying them into RAM
- RAM starts to fill up as all these programs, documents and files are copied

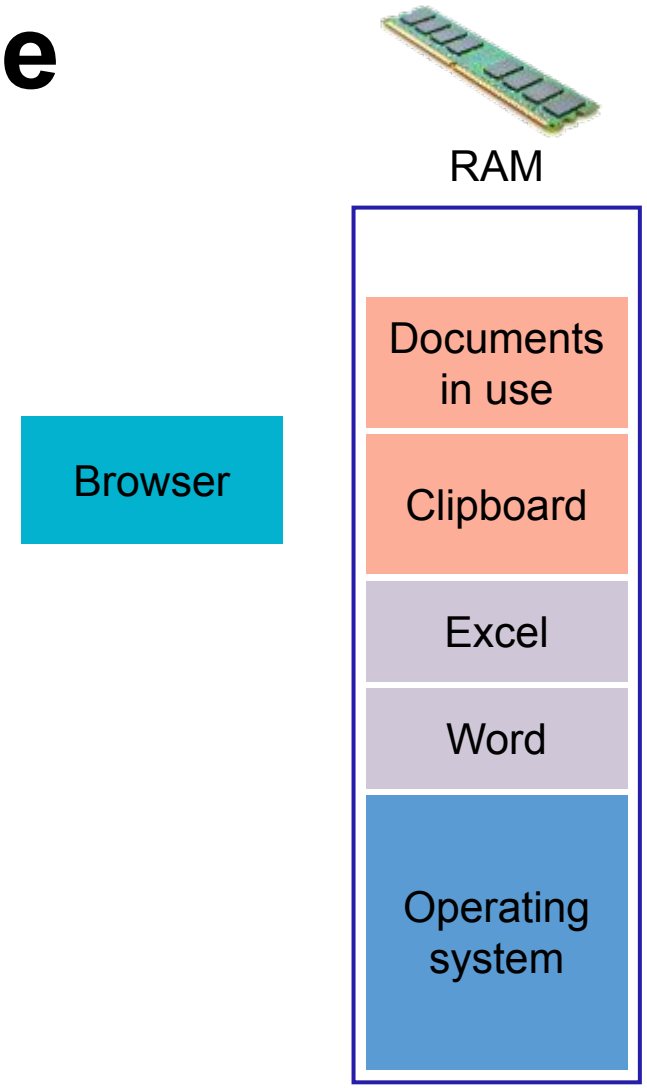


RAM



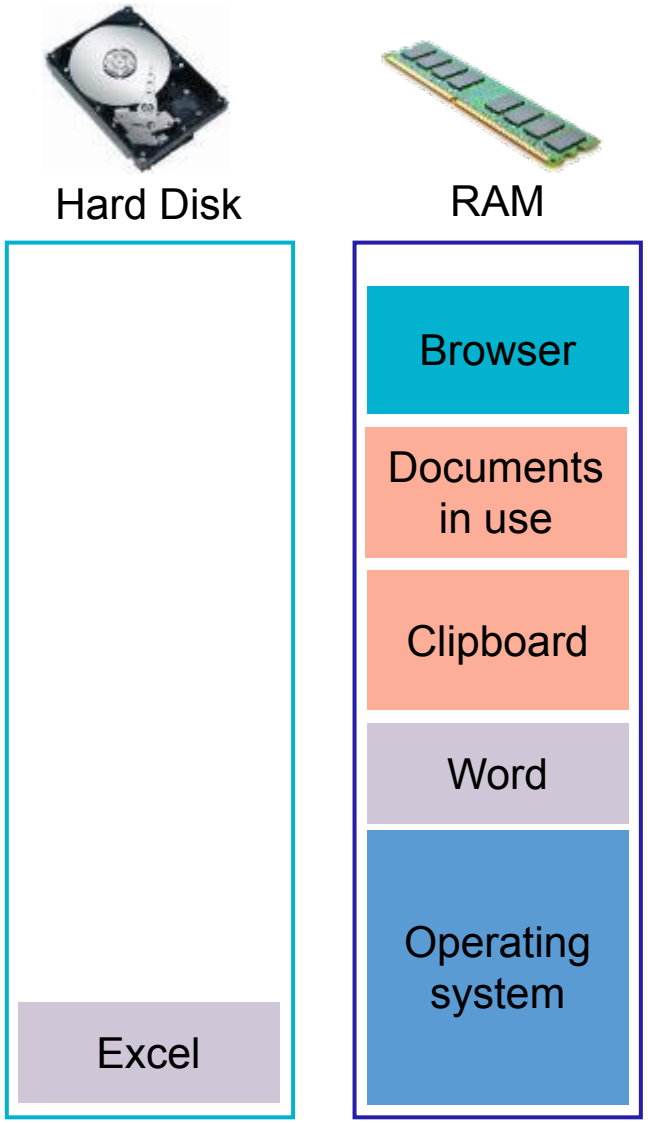
Running out of space

- You now want to open a browser to search the Internet
- The browser software needs more memory than you have free in RAM
  - What do you think happens?



Virtual memory

- Virtual memory is part of the secondary storage, such as a hard disk, is used as an extension to RAM
  - What are the advantages and disadvantages of using part of the secondary storage in this way?

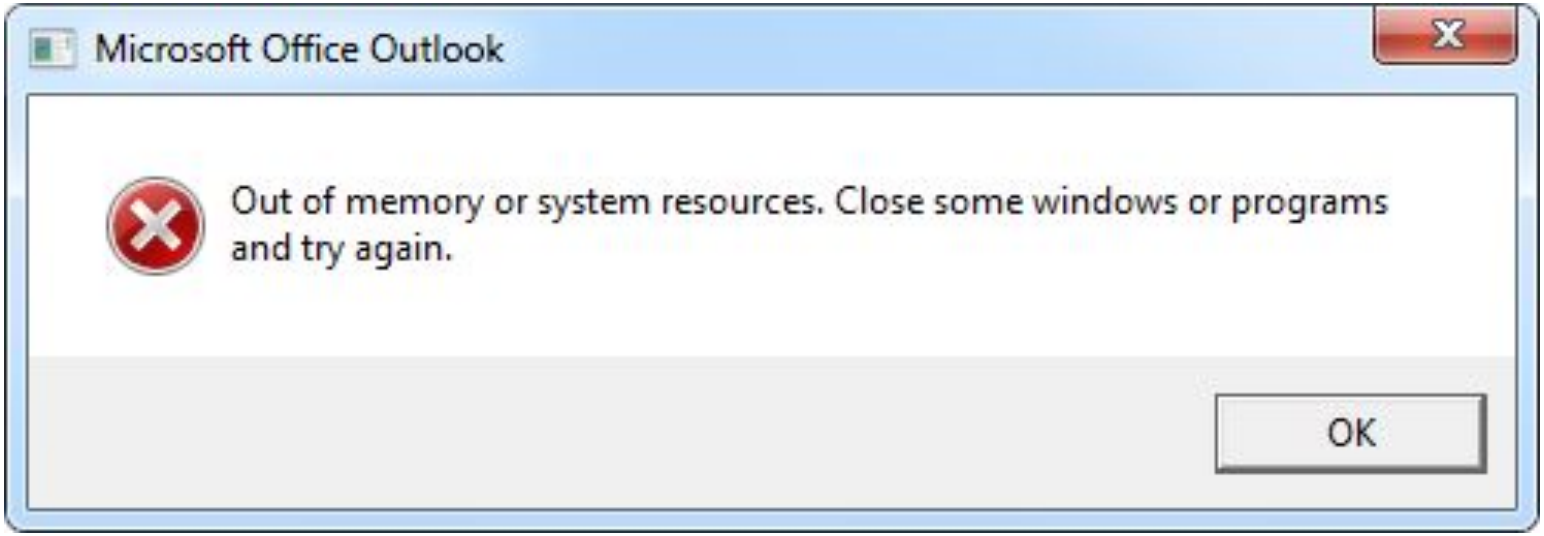


Virtual memory

Answers

- Advantages
  - Uses cheap secondary storage on the hard drive
  - Allows the running of programs or combinations of programs that would be too large to fit into the main memory
  - Prevents error messages saying ‘out of memory’ – the programs and files will still open
- Disadvantages
  - Accessing virtual memory is very slow
  - To access data, the existing data in RAM needs to be copied to the virtual memory, then data in virtual memory needs to be copied to RAM

What happens when RAM completely fills up?



Read-only memory (ROM)

- Some data needs to be permanently held in primary storage, even when a computer has no power
  - Read-only memory (ROM) is used to store this data
  - Data is read from ROM, but cannot be written to it
  - It is non-volatile as the data isn't lost if the power is off
- What is stored in ROM on a modern computer?

ROM

Answers

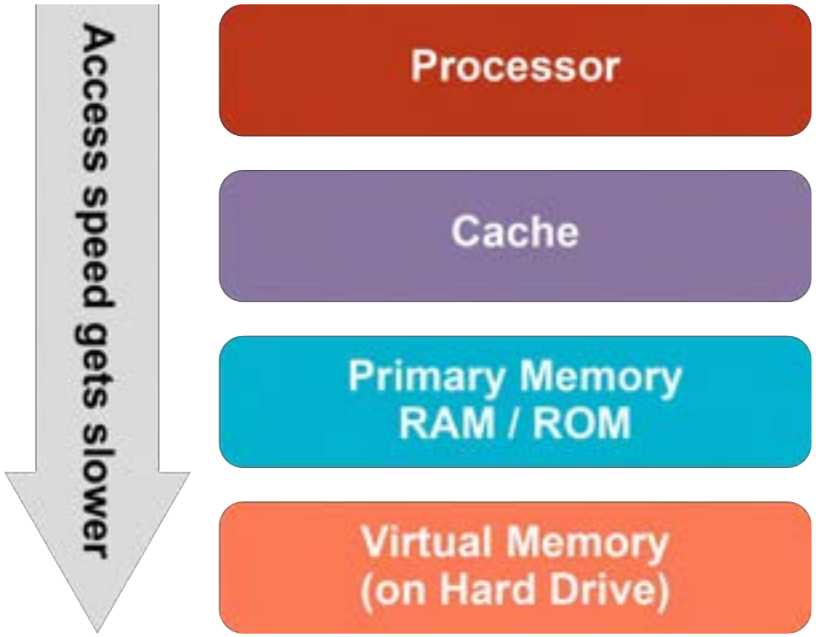
- ROM is used in modern computers to store:
  - The initial program that is run when the computer is turned on
  - This is known as the bootstrap
  - It tells the computer where it will find the operating system on the hard drive
- It also stores the Basic Input/Output System (BIOS)
  - The BIOS can run without a hard drive or other secondary storage being present
  - It controls basic technical configuration of the computer such as the processor speed and system time

RAM vs ROM differences

Characteristic	RAM	ROM
Size	Typically 4 GB – 32 GB	Typically 4 MB – 8 MB
Used to store	Running programs and operating system	BIOS and bootstrap
Read ability	Yes	Yes
Write ability	Yes	No
Volatile	Yes	No

Memory speed

- The speed that data can be accessed changes through the different components in a computer
  - The slower components are far cheaper for each byte of data stored



Worksheet 3

- Complete **Task 2** on **Worksheet 3**

Plenary

- In pairs test each other on the following:

1. What does RAM stand for?
2. What does ROM stand for?
3. State one difference between RAM and ROM
4. How much RAM is typically available in a modern personal computer?
5. Why is virtual memory much slower than RAM?

Plenary

Answers

- In pairs test each other on the following:

1. What does RAM stand for? Random Access Memory
2. What does ROM stand for? Read Only Memory
3. State one difference between RAM and ROM
   RAM can be written to, ROM can't
   RAM is volatile, ROM is non-volatile
4. How much RAM is typically available in a modern personal computer?
   Typically, a modern personal computer will have 8-32 GB of RAM available
5. Why is virtual memory much slower than RAM?
   Virtual memory is stored on secondary storage which is much slower than RAM

Copyright

© 2024 PG Online Limited

The contents of this unit are protected by copyright.

This unit and all the worksheets, PowerPoint presentations, teaching guides and other associated files distributed with it are supplied to you by PG Online Limited under licence and may be used and copied by you only in accordance with the terms of the licence. Except as expressly permitted by the licence, no part of the materials distributed with this unit may be used, reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic or otherwise, without the prior written permission of PG Online Limited.

Licence agreement

This is a legal agreement between you, the end user, and PG Online Limited. This unit and all the worksheets, PowerPoint presentations, teaching guides and other associated files distributed with it is licensed, not sold, to you by PG Online Limited for use under the terms of the licence.

The materials distributed with this unit may be freely copied and used by members of a single institution on a single site only. You are not permitted to share in any way any of the materials or part of the materials with any third party, including users on another site or individuals who are members of a separate institution. You acknowledge that the materials must remain with you, the licencing institution, and no part of the materials may be transferred to another institution. You also agree not to procure, authorise, encourage, facilitate or enable any third party to reproduce these materials in whole or in part without the prior permission of PG Online Limited.

OCR
Oxford Cambridge and RSA
An OCR endorsed teaching and learning tool

GCSE OCR

Computer Science
J277

Secondary storage

Unit 1
Systems architecture

PG ONLINE

4

Objectives

- Discuss the need for secondary storage including optical, magnetic and solid state storage
- Evaluate suitable storage devices and media for a given application using the following characteristics:
  - Capacity
  - Speed
  - Portability
  - Durability
  - Reliability
  - Cost

Starter

- RAM is a type of primary storage
  - It has a fast data rate but is volatile (the data is lost if there is no power)
  - What storage devices are used that are non-volatile?

Secondary storage

- Secondary storage is not directly accessible by the CPU
  - It is non-volatile, meaning it will keep data even if there is no power
  - Secondary storage devices may be internal or external to the computer
- What is secondary storage used for?

Uses of secondary storage

Answers

- It has many different uses – for example:
  - Programs and data are stored on hard drive
  - Blu-rays may be used to distribute films
  - Memory sticks may be used to transport data from one place to another
  - Magnetic tape or external hard drives may be used for backup
  - SD cards can be used for additional storage on cameras and smartphones – this is used for music, video and photos

Storage types

- Primary storage
  - RAM and ROM
- Secondary storage
  - Hard Disk Drive (HDD)
  - Solid State Drive (SSD)
- Offline secondary storage
  - Compact Disc (CD), Digital Versatile Disc (DVD) or BluRay
  - Flash memory, SD cards
  - Removable HDD or SSD
  - Magnetic tape

Storage methods

- Magnetic: Mechanical parts move over the disks surface to read and write data magnetically, or a drive head reads a magnetic tape
- Optical: Read and write data using light (laser)
- Solid State: Data is recorded onto solid memory chips without any moving parts

Worksheet 4

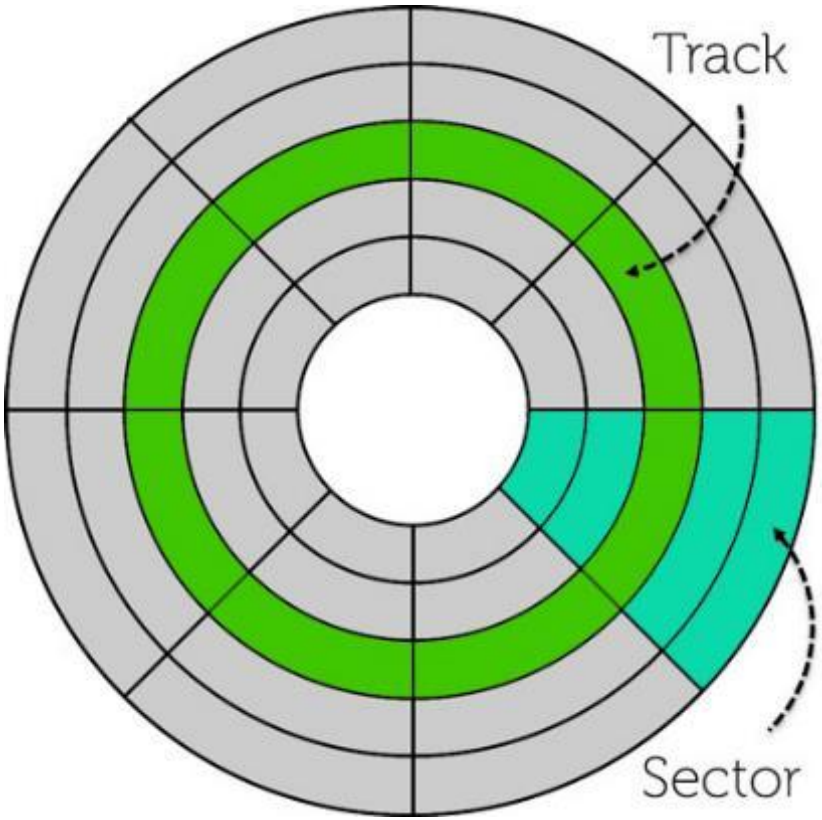
- Complete **Task 1** on **Worksheet 4**

Magnetic disks

Basic features:

- Disk contains concentric circles called tracks
- Each track is divided into sectors
- Disk heads mounted on mechanical arms read and write the data

- A disk with a solid platter is a ‘hard’ disk
  - Soft plastic disks are known as ‘floppy’ disks



Hard disks

- Parts of a hard disk

Magnetic storage: hard disks

- Fixed magnetic hard disks are still used in many PCs and laptops
  - They have a very large storage capacity, up to 12TB or more
  - They are a very cheap form of storage compared to solid state drives
- Portable hard disks can be connected to a computer via a USB port
  - They are used for backing up or transporting data

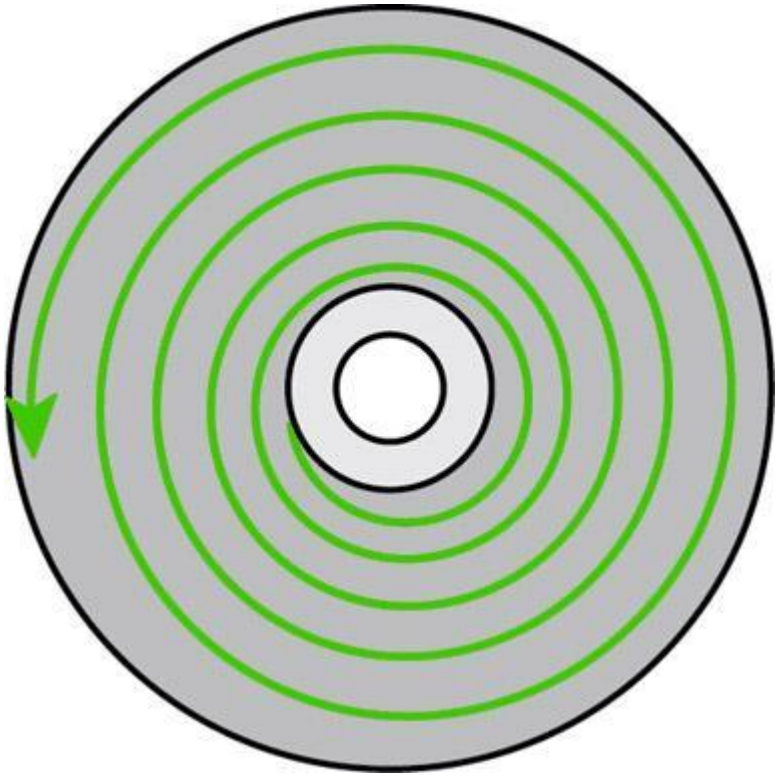
PG ONLINE

Magnetic storage

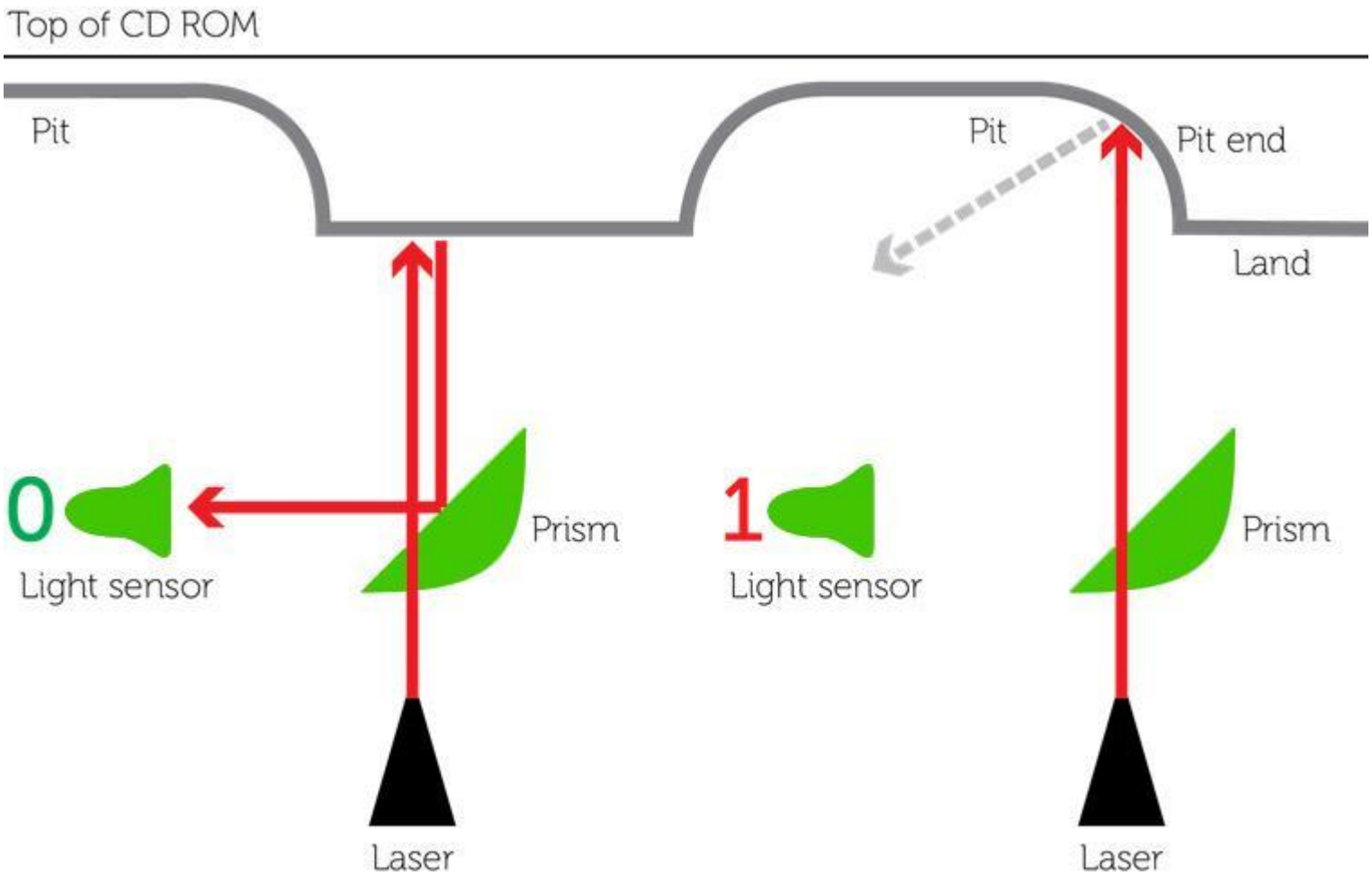
- Advantages:
  - Cheap, large storage capacities, relatively fast write speed
- Disadvantages:
  - Lots of mechanical parts, durability an issue, sealed unit due to disk head and platter precision and not very portable
- Uses:
  - Personal computers, storage of large quantities of data
- Capacity:
  - 500GB - 12TB or greater

Optical storage

- Basic features:
  - Data is stored as pits and lands burnt or pressed into a spiral track circulating outwards from the centre
  - A laser beam passes over the pits and lands the level of reflected light is measured
  - From this signal, 0s and 1s can be derived



How CDs work

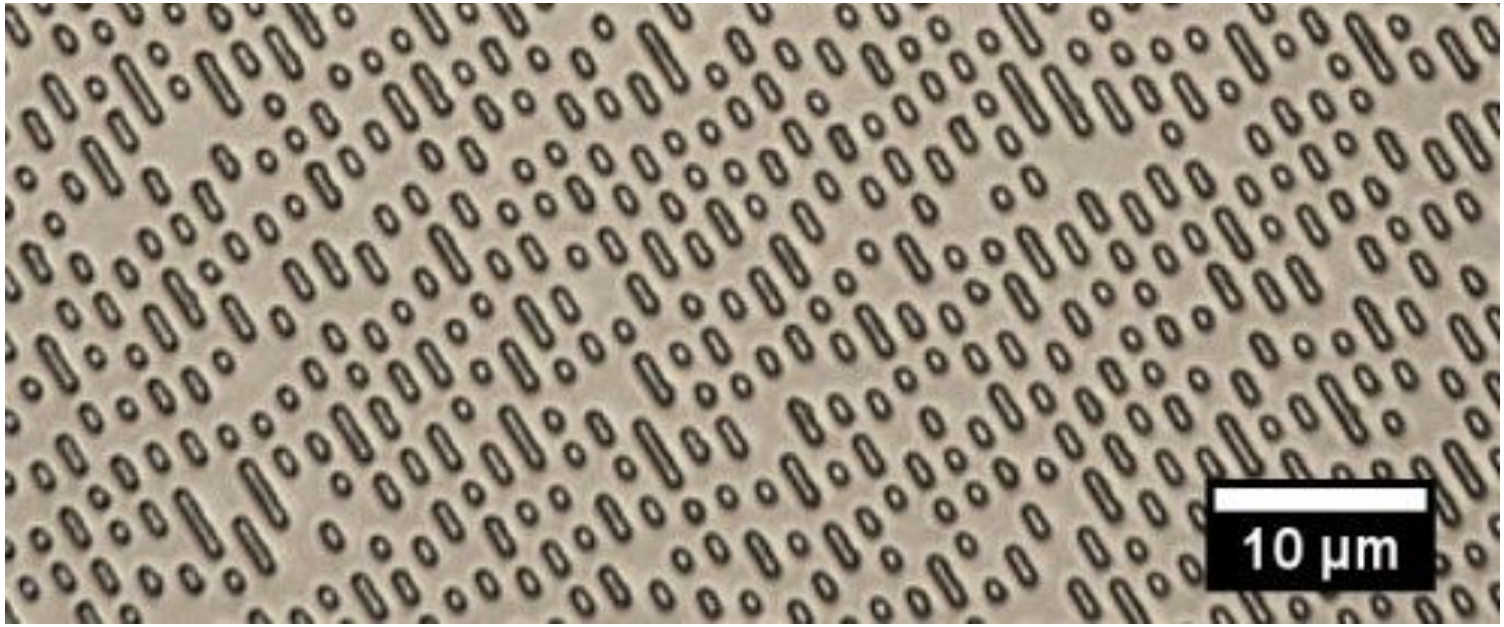

Top of CD ROM
Pit
Pit
Pit end
Land
0
Light sensor
Prism
1
Light sensor
Prism
Laser
Laser

Optical storage

- Advantages:
  - Cheap, very easily portable, takes up little space physically
- Disadvantages:
  - Less storage capacity compared to other types
  - Easily damaged / scratched, requires CD/DVD/Blu-ray reader
  - Slow write speeds
- Uses:
  - Songs, videos and other multi-media storage, backup and archiving of data
- Capacity:
  - CD-ROM – up to 720 MB
  - DVD – up to 8.4 GB (dual layered disk)
  - Blu-ray – up to 50 GB (dual layered disk)

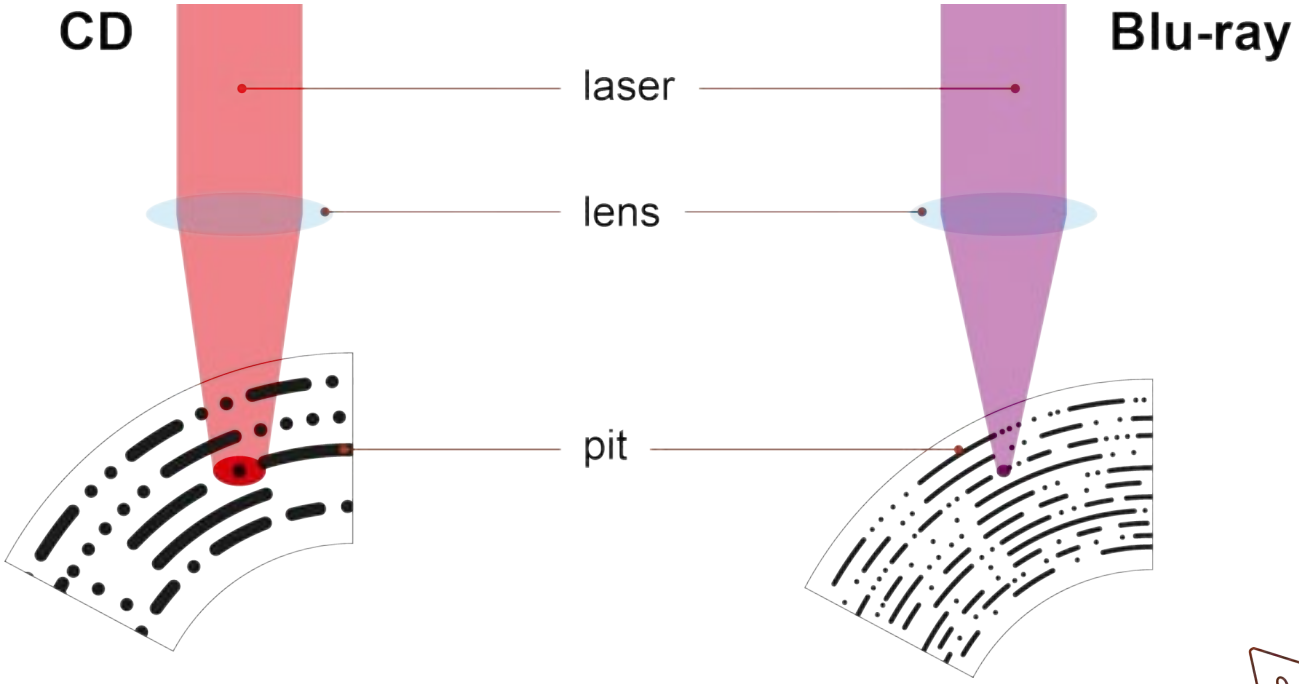
CDs, DVDs and BluRay

- Why are the capacities of these discs different given they are all the same physical size?
  - Microscopic view of the surface of a CD ROM



Pit size and laser wavelength

- A CD has bigger pits and lands as red light has a larger wavelength
  - The smaller Blu-ray pits and lands allow it to store more data

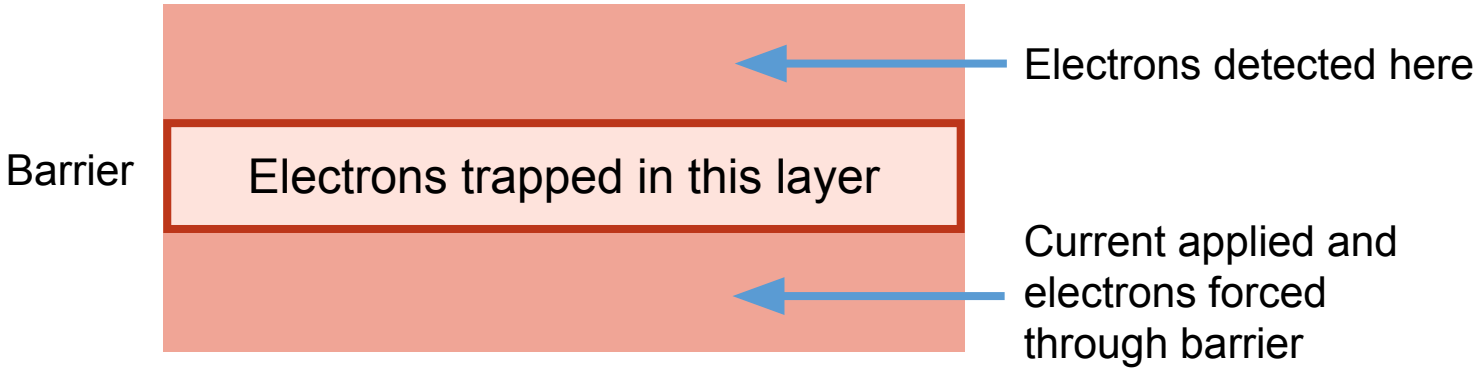


Solid State Drives (SSD)

- Basic features:
  - Solid-state disks use non-volatile flash memory to store information
  - Very fast read/write speeds as it doesn't need to wait for a disk to spin to the correct location and an arm to move
  - No mechanical or moving parts meaning these disks are very durable

How flash memory works

- Large electric current used to force electrons through a barrier and trap them on the other side



- They remain on the other side until ‘flashed’ with a new current, hence the name
  - Trapped (charged) or not trapped = 0 or 1

Advantages /disadvantages of SSDs

- Advantages:
  - Highly durable, no moving parts, very fast read/write speeds, no noisy fan or drive arm, faster start up times
- Disadvantages:
  - More expensive than magnetic hard disks, similar storage capacity as magnetic disks
- Uses:
  - Higher end computers
  - Laptops
  - Smartphones and tablets
- Capacity:
  - 100GB – 16TB

Flash memory

- Low cost, portable, no moving parts, durable
- This makes them ideal for a range of offline devices:
  - Cameras
  - Mobile phones
  - USB memory sticks

Worksheet 4

- Complete **Task 2** and **Task 3** on **Worksheet 4**

Other storage characteristics

- Storage devices have different characteristics that determine if they are a suitable choice
- What do each of the following characteristics mean?
  - Capacity
  - Speed
  - Portability
  - Durability
  - Reliability
  - Cost

PG ONLINE

Storage characteristics

Answers

- Characteristics of storage devices
  - **Capacity**: How much data can be stored – e.g. 700 MB, 50 GB or 2 TB
  - **Speed**: The rate (usually in MB/s) at which data can be read or written
  - **Portability**: how easy it is to carry – is the device small?
  - **Durability**: will the device break if dropped? How well does it work with extreme temperatures or magnetic fields?
  - **Reliability**: how likely is the data (or some of it) to be lost?
  - **Cost**: what is the cost of a device? How much is it to store 1 MB of data

Data capacity

- Different storage devices have a range of storage capabilities
- A higher capacity will allow greater data storage
  - How do you decide which storage device to use?
  - How do you calculate storage requirements?

Data capacity

- When you know how much data you need to store, you can decide which storage device(s) would be most appropriate
  - If you have 300GB of data to store, is optical storage appropriate?
  - If not, why not? What could you use instead?
  - Why is this particular device appropriate?

Calculating data capacity

- Knowing the capacity required will enable us to make an informed decision as to which device to use
- If we wish to store 5000 photos and each photo has a file size of 10 MB, we need a total of 50 GB
  - Which storage devices are suitable for this amount of data?
  - Which storage device would be suitable for storing this data on a smartphone?

Worksheet 4

- Complete **Task 4** on **Worksheet 4**

Plenary

- In pairs, give answers to the following:
  - 5 examples of storage devices
  - 4 other characteristics of storage devices other than cost
  - 3 technologies that storage devices use
  - 2 words that describe data that is or isn't lost when power is turned off
  - 1 word that means 1000 Gigabytes

Plenary

Answers

- Examples of storage devices:
  - Hard disk drive, floppy disk drive, tape drive, CD drive, DVD drive, Blu-ray drive, USB flash drive, SSD drive, SD card
- Characteristics of storage devices:
  - Capacity, speed, portability, durability, reliability (and cost)
- Technologies used in storage devices:
  - Magnetic, optical, solid state
- Volatile – data is lost; non-volatile – data is not lost
- 1 Terabyte (TB) = 1000 Gigabytes

Copyright

© 2024 PG Online Limited

The contents of this unit are protected by copyright.

This unit and all the worksheets, PowerPoint presentations, teaching guides and other associated files distributed with it are supplied to you by PG Online Limited under licence and may be used and copied by you only in accordance with the terms of the licence. Except as expressly permitted by the licence, no part of the materials distributed with this unit may be used, reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic or otherwise, without the prior written permission of PG Online Limited.

Licence agreement

This is a legal agreement between you, the end user, and PG Online Limited. This unit and all the worksheets, PowerPoint presentations, teaching guides and other associated files distributed with it is licensed, not sold, to you by PG Online Limited for use under the terms of the licence.

The materials distributed with this unit may be freely copied and used by members of a single institution on a single site only. You are not permitted to share in any way any of the materials or part of the materials with any third party, including users on another site or individuals who are members of a separate institution. You acknowledge that the materials must remain with you, the licencing institution, and no part of the materials may be transferred to another institution. You also agree not to procure, authorise, encourage, facilitate or enable any third party to reproduce these materials in whole or in part without the prior permission of PG Online Limited.

OCR
Oxford Cambridge and RSA
An OCR endorsed teaching and learning tool

GCSE OCR

Computer Science
J277

1

Units of data storage and binary numbers

Unit 2
Data representation

PG ONLINE

Objectives

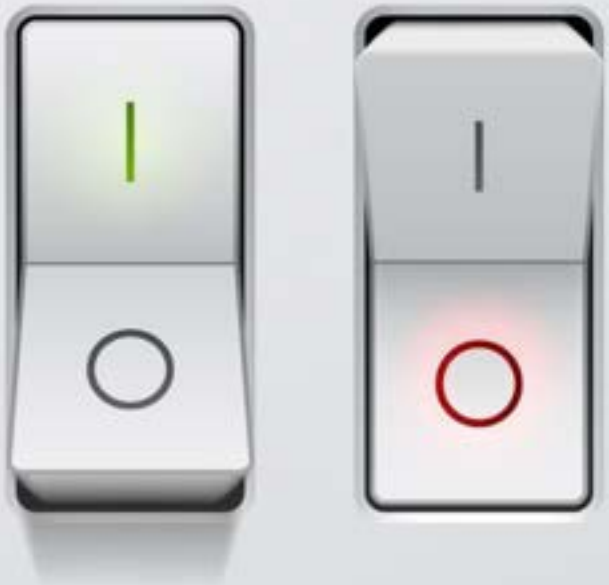
- Define the terms bit, nibble, byte, kilobyte, megabyte, gigabyte, terabyte and petabyte
- Understand that data needs to be converted into a binary format to be processed by a computer
- Convert positive denary whole numbers (0-255) into 8-bit binary numbers and vice versa

Starter

- What is this symbol?
  - What numbers is it made from?
  - How many states does it control?

Understanding binary

- Computers understand only two states: power on, or power off
- This is represented by switches, and computers are essentially calculators made up of billions of switches
  - Power on = 1
  - Power off = 0

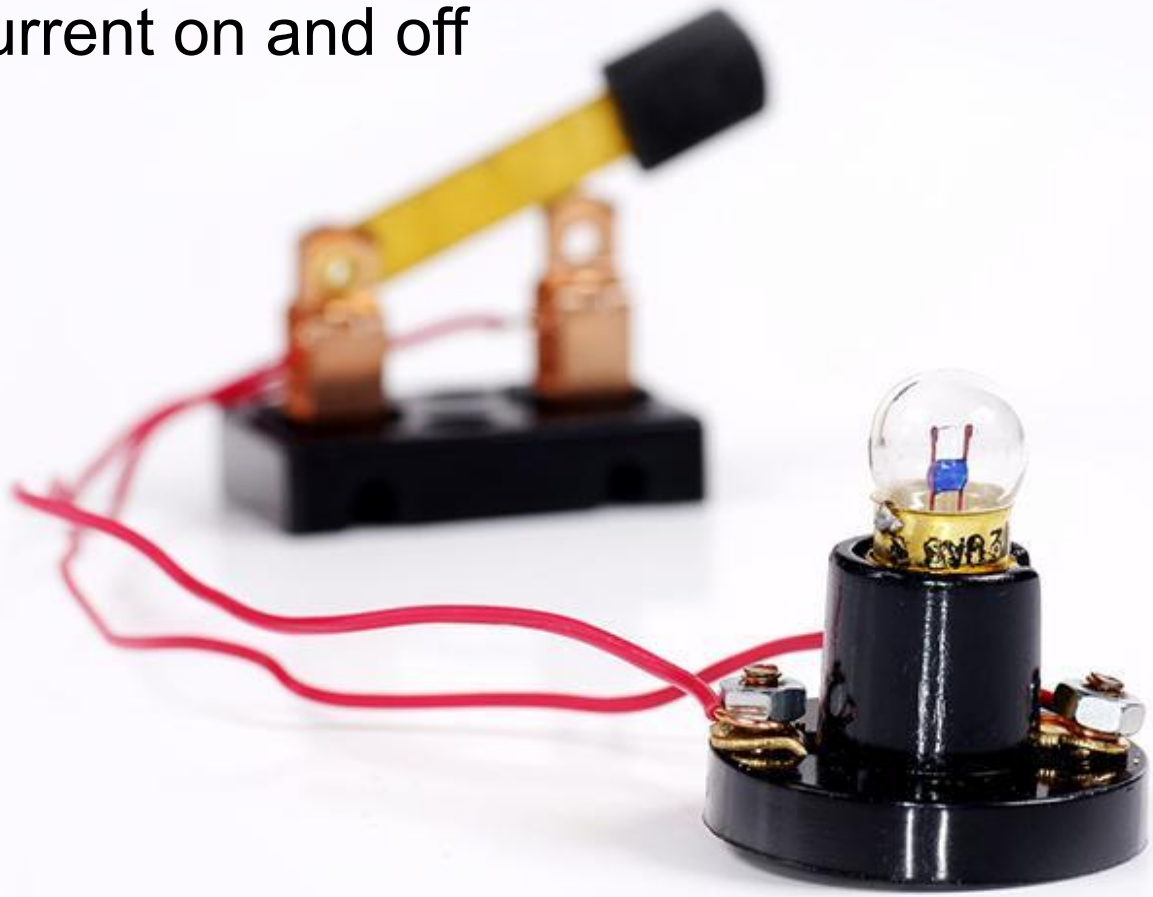


Binary number system

- Computers use a binary number system consisting of only 0s and 1s
  - Everything that a computer needs to process must be converted into a binary format
  - This format is used for storing numbers, text, images, sound and program instructions

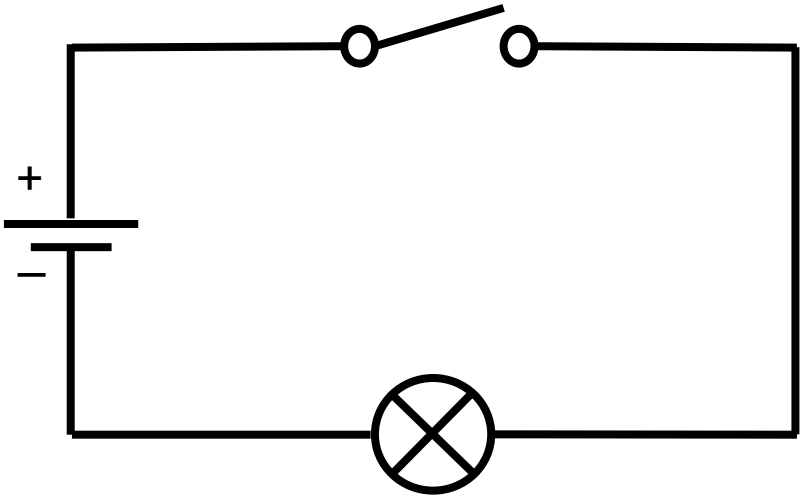
Creating a circuit

- Computers comprise billions of switches to turn current on and off



Circuits

- Use the keywords below to explain how this electrical circuit works:



Battery **Switch** **Lamp**

Bits and bytes

- Many units are used for data storage in computers
  - 1 bit = a single 0 or 1
  - 1 byte = 8 bits = 1 character of text
  - (A nibble = 4 bits or half a byte)
  - 1 kB (kilobyte) = 1000 bytes
  - 1 MB (megabyte) = 1000 kB or 1000x1000 bytes
  - 1 GB (gigabyte) = 1000 MB
  - 1 TB (terabyte) = 1000 GB
  - 1 PB (petabyte) = 1000 TB
- Make a mnemonic for k, M, G, T and P

Units

Answers

- One suggestion
  - **K**ites – kilo
  - **M**ake - mega
  - **G**reat – giga
  - **T**oy – tera
  - **P**lanes - peta

PG ONLINE

A sense of scale

File	Size
One character of text	1 byte
A full page of text	30 kB
One small digital colour photograph	3 MB
Music CD capacity	650 MB
DVD capacity	4.5 GB
Hard disk capacity	1 TB

Binary representation

- How many combinations or states can each of the number of switches given produce?

Number of Switches (Bits)	Possible combinations or states
1	2
2	4
3	
4	
5	
6	
7	
8	

Binary representation

Number of Switches (Bits)	Possible combinations or states
1	2
2	4
3	8
4	16
5	32
6	64
7	128
8	256

Binary and denary number systems

- Denary is a base 10 number system with 10 digits 0-9
  - Why do we usually use 10 digits?
- Binary is a base 2 number system with 2 digits, 0 and 1

Numbers with the denary system

Thousands, Hundreds, Tens and Units

1000		100		10		1	
2		**7**		**0**		**3**	
2x1000	+	7x100	+	0x10	+	3x1	= 2703

Binary works in exactly the same way

128	64	32	16	8	4	2	1
1	**0**	**0**	**1**	**0**	**1**	**1**	**0**
1x128	+		1x16	+	1x4	+ 1x2	= ?

Binary to denary conversion

- What is:

128	64	32	16	8	4	2	1
0	0	1	1	1	0	0	1

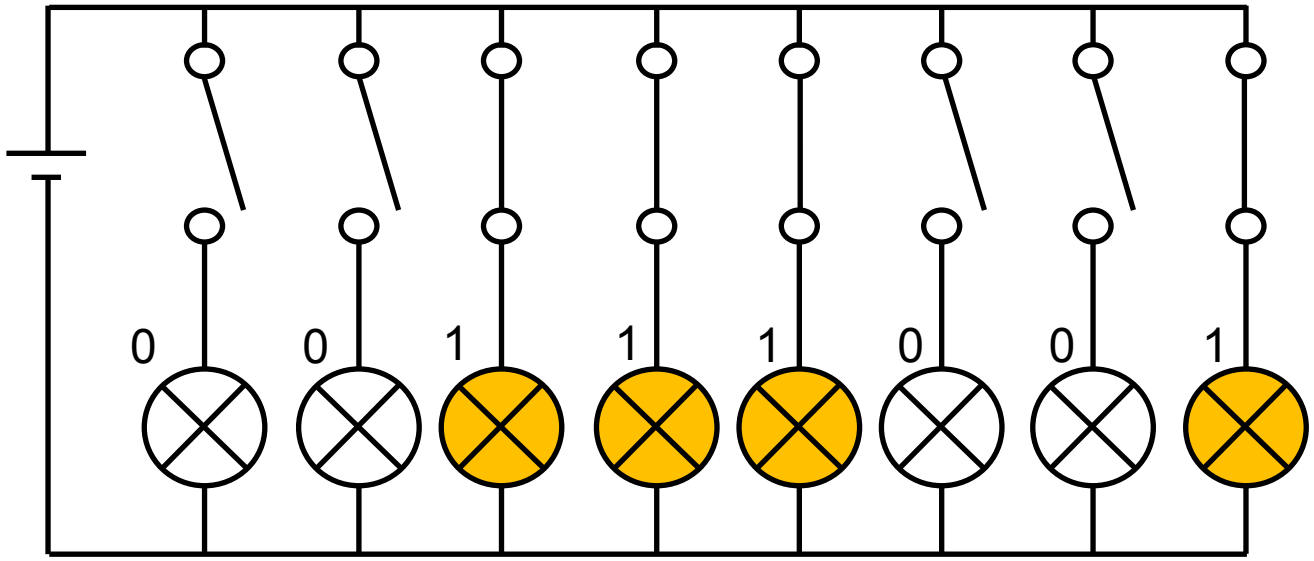
Most and least significant bit

- The most significant bit (MSB) is the bit with the largest value
  - This is the bit that is furthest to the left
- The least significant bit (LSB) is the bit with the smallest value
  - This is the bit that is furthest to the right

128	64	32	16	8	4	2	1
1	0	0	1	1	1	0	1

Representing values

- Consider the same byte value 0011 1001
  - This is represented in denary as 57
  - As an electrical circuit this could be represented as:



Worksheet 1

- Complete **Task 1** on **Worksheet 1**

Denary to binary

- How do you convert 28 to binary?
- Method
  - Working right to left, write out the numbers 1, 2, 4, 8 and so on, doubling each time to 128

128	64	32	16	8	4	2	1
0	0	0	1	1	1	0	0

  - 128, 64 and 32 are all greater than 28, so put a zero for these
  - Put a 1 in the 16 column, 28-16=12
  - Put a 1 in the 8 column, 12-8 = 4
  - Put a 1 in the 4 column, 4-4=0 so put zero in other columns

Representing large integers

- 1 byte (8 bits) can represent the numbers between 0 (0000 0000) and 255 (1111 1111)

$$2^8 - 1 = 255$$

- What is the largest number that can be held in 16 bits?
- What is the largest number that can be held in 32 bits?

Larger numbers

Answers

- The largest number that can be held with 16 bits is:
  - $2^{16} – 1 =$
  - 65 536 – 1 =
  - <u>65 535</u>
  - (The range of numbers from 0 to 65 535 can be stored in a 16 bit number)
- The largest number that can be held with 32 bits is:
  - $2^{32} – 1 =$
  - 4 294 967 296 – 1 =
  - <u>4 294 967 295</u>

Representing large integers

- Computers used to use two bytes to represent an integer
  - Older versions of Microsoft Excel used to have only 65,536 rows (from 1 to 65,536)
  - The rows will have been addressed with a two byte integer
- Nowadays, they mostly use four bytes
  - Programmers need to consider the maximum numbers that can be stored by the type of integer they are using

Worksheet 1

- Complete **Task 2** on **Worksheet 1**

Plenary

- In pairs:
  - Which numbers are stored in one bit?
  - How many bits are there in a nibble and a byte?
  - What are the units after a kilobyte?
  - What is zero in binary?
  - What is 255 in binary?
  - What is 39 in binary?

Plenary

Answers

- In pairs:
  - Which numbers are stored in one bit? 0 and 1
  - How many bits are there in a nibble and a byte?
    Nibble = 4 bits, byte = 8 bits
  - What are the units after a kilobyte?
    megabyte, gigabyte, terabyte, petabyte
  - What is zero as an 8 bit binary number? 0000 0000
  - What is 255 in binary? 1111 1111
  - What is 39 as an 8 bit binary number? 0010 0111

Copyright

© 2024 PG Online Limited

The contents of this unit are protected by copyright.

This unit and all the worksheets, PowerPoint presentations, teaching guides and other associated files distributed with it are supplied to you by PG Online Limited under licence and may be used and copied by you only in accordance with the terms of the licence. Except as expressly permitted by the licence, no part of the materials distributed with this unit may be used, reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic or otherwise, without the prior written permission of PG Online Limited.

Licence agreement

This is a legal agreement between you, the end user, and PG Online Limited. This unit and all the worksheets, PowerPoint presentations, teaching guides and other associated files distributed with it is licensed, not sold, to you by PG Online Limited for use under the terms of the licence.

The materials distributed with this unit may be freely copied and used by members of a single institution on a single site only. You are not permitted to share in any way any of the materials or part of the materials with any third party, including users on another site or individuals who are members of a separate institution. You acknowledge that the materials must remain with you, the licencing institution, and no part of the materials may be transferred to another institution. You also agree not to procure, authorise, encourage, facilitate or enable any third party to reproduce these materials in whole or in part without the prior permission of PG Online Limited.

OCR
Oxford Cambridge and RSA
An OCR endorsed teaching and learning tool

GCSE OCR

Computer Science
J277

2

Binary arithmetic and hexadecimal

Unit 2
Data representation

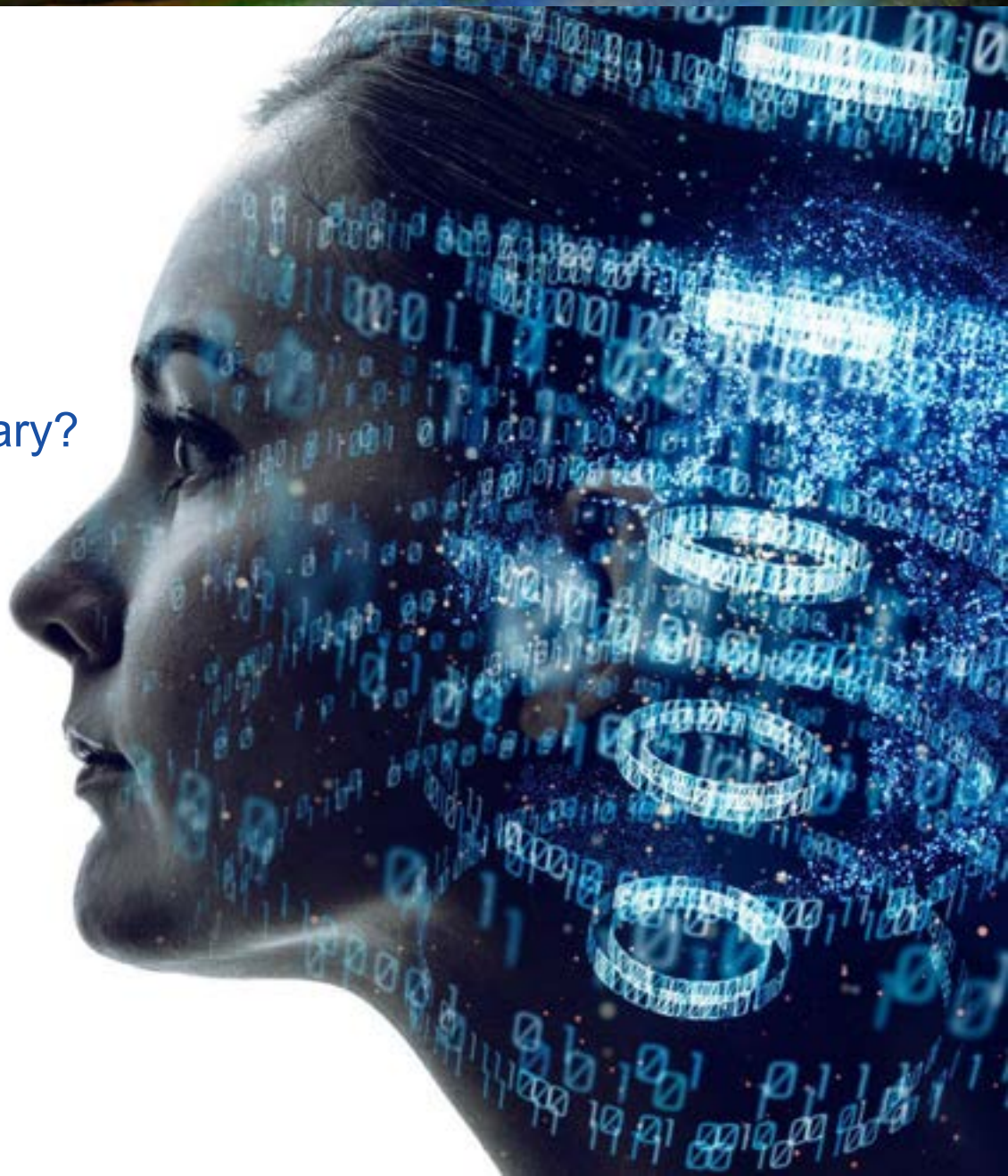
PG ONLINE

Objectives

- Convert positive denary whole numbers (0-255) into 2-digit hexadecimal numbers and vice versa
- Convert between binary, denary and hexadecimal equivalents of the same number
- Add two 8-bit binary integers and explain overflow errors which may occur
- Understand the use of binary shifts

Starter

- Review:
  - What is 7 in binary?
  - What is 1001 in denary?
- What is the result of the following binary calculation: 1001 + 110 ?



Starter

Answers

- What is 7 in binary?
  - 111
- What is 1001 in denary?
  - 9
- What is the result of the following binary calculation:

Binary	Denary
1001+	9+
110	6
1111	=15

Binary to denary conversion

- Practise converting binary to denary and back again
  - Convert each of the following to binary or denary

128 0	64 1	32 0	16 1	8 1	4 0	2 0	1 1
1	1	0	0	0	0	1	0
14							
105							

How can you quickly tell which is odd and even in any binary number?

Binary to denary conversion

Answers

- 0101 1001 = 89
- 1100 0010 = 194
- 14 = 1110
- 105 = 110 1001
- Even binary numbers will have a 0 at the right end
- Odd binary numbers will have a 1 at the right end

Hexadecimal

- Hexadecimal (or hex) is a number system which uses base 16
- As we only have 10 digits, it uses 0-9 and then the letters A to F

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F

- What is **F** in denary?
- What is **10** in hex?

Hex to denary conversion

- You will only need to translate one- or two-digit hexadecimal numbers

16s		Units	
2		**A**	
2x16	+	10	= 42 in denary

- Multiply the left-hand digit by 16, then add the units
- What is hex **27** in denary?

Denary to hex conversion

- Divide the denary number by 16 to get the number of 16s (the left-hand hex digit)
- The remainder gives you the units

Denary **18** becomes:

18 / 16 = 1 remainer 2 so
the hex value for **18** is **12**

(*Spoken, 'One Two', not 'Twelve'*)

- What is denary **27** in hex?
- What is denary **44** in hex?

Hex to denary conversions

Answers

- What is hex 27 in denary?
  - 2 x 16 + 7 = 39
- What is denary 27 in hex?
  - **1** x 16 remainder 11 (11 is **B** in hex) = 1B
- What is denary 44 in hex?
  - **2** x 16 remainder 12 (12 is **C** in hex) = 2C

Worksheet 2

- Complete **Task 1a** on **Worksheet 2**

Binary to hex conversion

- Take a binary word of 8 bits

  1 1 1 0 0 1 0 1

- Divide into two nibbles of 4 bits

  1 1 1 0 0 1 0 1

- Convert each nibble into its hex value and rejoin

  1 1 1 0 = 14 = E in Hex + 0 1 0 1 = 5 in Hex

  So 1 1 1 0 0 1 0 1 = E5 in Hex

Hex to binary conversion

- What is **3B** in binary?

  Split the two hex characters

  3 = 0011 in binary and **B = 1011**

  So **3B = 0011 1011** in binary

  What is hex **21** in binary?

  What is hex **A5** in binary?

Hex to binary conversion

Answers

- What is hex 21 in binary?
  - 2 = 0010, 1 = 0001
  - 21 in hex = 0010 0001 in binary
- What is hex A5 in binary?
  - A = 10 = 1010, 5 = 0101
  - A5 in hex = 1010 0101 in binary

Worksheet 2

- Complete **Task 1b** on **Worksheet 2**

Why use hex?

- There are advantages for programmers and computer scientists in using hex rather than binary
  - It is much simpler to remember a hex value than a binary value
  - It is quicker to write or type since a hex digit only takes up one digit rather than 4 bits
  - People are less likely to make an error with fewer digits
  - It is easy to convert between hex and binary

Binary addition

- Use binary to add two numbers together

1 + 1 = 10?

Simple denary addition

Work right to left :

1. **Add the Units**

$$
\begin{array}{r}
14 \\
+\ 17 \\
=\ 1 \\
1
\end{array}
$$

Simple denary addition

Work right to left :

1. **Add the Units**
2. **If Over 9, *Carry* Tens**

	1	
	1	4
+	1	7
=		1

Simple denary addition

Work right to left :

1. **Add the Units**
2. **If Over 9, *Carry* Tens**
3. **Add Tens**

$$
\begin{array}{rrr}
 & 1 & \\
 & 1 & 4 \\
+ & 1 & 7 \\
\hline
= & 3 & 1
\end{array}
$$

The rules of binary addition

Work right to left and apply these simple rules:

1. **0 + 0 = 0**
2. **0 + 1 = 1**
3. **1 + 0 = 1**
4. **1 + 1 = 0** *Carry **1***
5. **1 + 1 + 1 = 1** *Carry **1***

	1 1	
	1 1 1 0	14
+	1 1 0 0	12
=	**1 1 0 1 0**	**26**

Carry Bit
Rule 5
Rule 4
Rule 2 or 3
Rule 1

Adding binary values

Carry		1	1	1	
	1	0	1	0	1
+			1	1	1
=	**1**	**1**	**1**	**0**	**0**

Adding numbers

- Computers work with a fixed number of bits at a time
  - This can cause problems
  - What problem will arise when adding the following bytes and storing the result in one byte?

1 1 1 1 0 0 0 0

+ 1 0 0 1 1 1 1 1

Overflow error

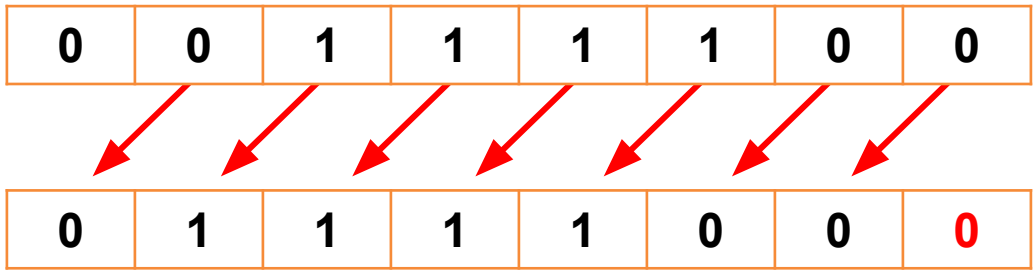
- When the result of an addition is too large for the number of bits the computer works with there will be an **overflow error**

```
  1 1 1 1
    1 1 1 1 0 0 0 0
    1 0 0 1 1 1 1 1
  -----------------
+ 1 1 0 0 0 1 1 1 1
```

9th Bit

Logical binary shift operations

- A binary shift left of one bit moves all the bits one place to the left
  - The vacant bit spaces are filled with zeros



- Looking at the table above, what effect does a shift left of one place have on the binary value?
  - What effect would a shift right of two places have?

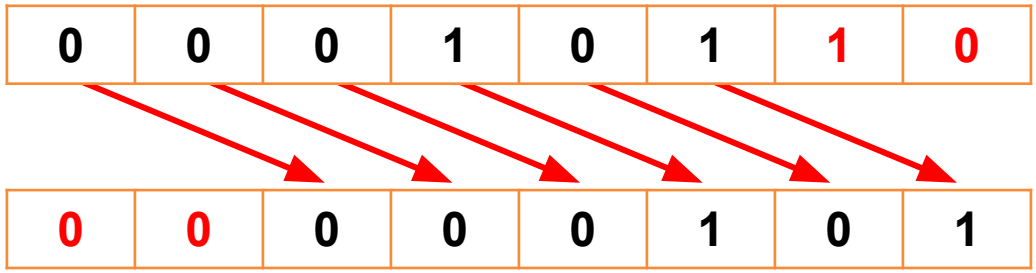
Shift operations

Answers

- A binary shift left of one bit doubles the number
  - E.g. 1000=8
    Shift left once, 10000=16
- A binary shift right of two places results in halving the number twice and rounding down each time
  - Example 1: 1000=8
    Right shift once, 100 = 4
    Right shift again, 10 = 2
  - Example 2: 1001 = 9
    Right shift once, 100 = 4
    Right shift again, 10 = 2

Effects of shifts

- Logical shifts can very quickly multiply or divide a binary number by a factor of two
  - Left shifts multiply
  - Right shifts divide
- A loss of accuracy can occur if 1 bits are removed:
  - 22 / 4 is not exactly 5



Worksheet 2

- Complete **Task 2**

Plenary

- Work out the answers to each of the following then compare your answers with a partner
  - Convert C in hexadecimal to denary
  - Convert 2B in hexadecimal to binary
  - Convert 1010 0011 in binary to hexadecimal
  - Calculate
    1001 0101 +
    0010 0110
  - Calculate the left shift of 0110 1011
  - Explain the effect of a left shift

Plenary

Answers

- C in hexadecimal = 12 in denary
- 2B in hexadecimal = 16*2 + 1*B = 32 + 11 = 43
- 1010 0011 in binary = A3 in hexadecimal
- 1001 0101 + 0010 0110 = 1011 1011 in binary
- Left shift 0110 1011 (107) = 1101 0110 (214)
- The left shift doubles the number

Copyright

© 2024 PG Online Limited

The contents of this unit are protected by copyright.

This unit and all the worksheets, PowerPoint presentations, teaching guides and other associated files distributed with it are supplied to you by PG Online Limited under licence and may be used and copied by you only in accordance with the terms of the licence. Except as expressly permitted by the licence, no part of the materials distributed with this unit may be used, reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic or otherwise, without the prior written permission of PG Online Limited.

Licence agreement

This is a legal agreement between you, the end user, and PG Online Limited. This unit and all the worksheets, PowerPoint presentations, teaching guides and other associated files distributed with it is licensed, not sold, to you by PG Online Limited for use under the terms of the licence.

The materials distributed with this unit may be freely copied and used by members of a single institution on a single site only. You are not permitted to share in any way any of the materials or part of the materials with any third party, including users on another site or individuals who are members of a separate institution. You acknowledge that the materials must remain with you, the licencing institution, and no part of the materials may be transferred to another institution. You also agree not to procure, authorise, encourage, facilitate or enable any third party to reproduce these materials in whole or in part without the prior permission of PG Online Limited.

OCR
Oxford Cambridge and RSA
An OCR endorsed teaching and learning tool

GCSE OCR

Computer Science
J277

3

Characters

Unit 2
Data representation

PG ONLINE

Objectives

- Understand the use of binary codes to represent characters
- Understand the term ‘character set’
- Explain the relationship between the number of bits per character in a character set, and the number of characters that can be represented using:

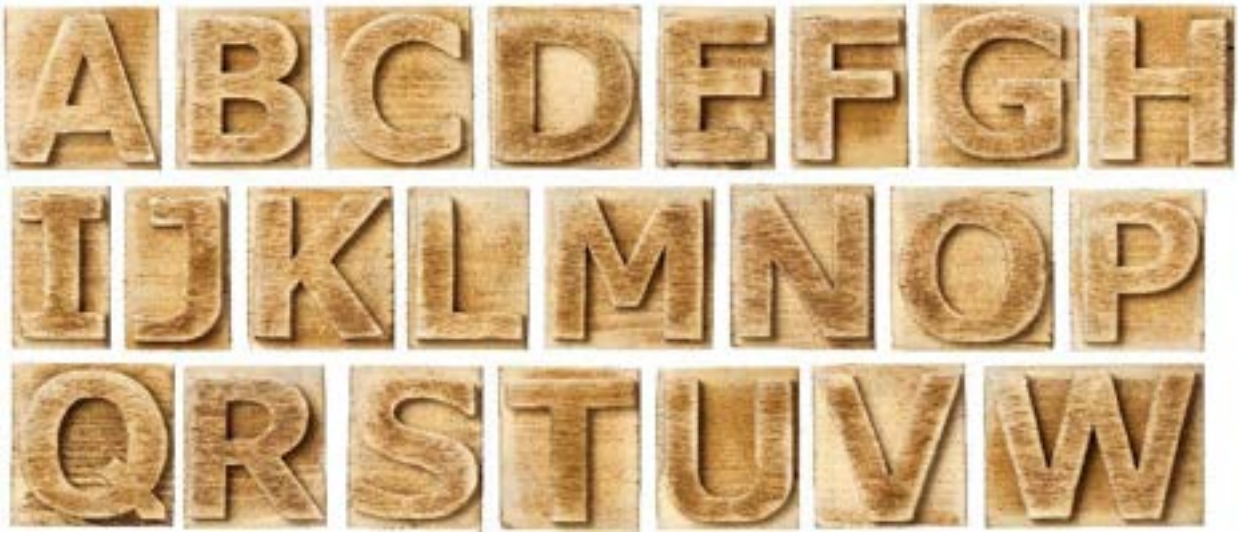
  ASCII

  Extended ASCII

  Unicode

Starter

- A computer's memory and storage only hold binary 1s and 0s
  - How might it be possible to store letters with only binary?



Representing text characters

- If a computer understands only 1s and 0s, what happens when the ‘M’ key is pressed on the keyboard?

0 1 0 0 1 1 0 1

Representing characters in binary

- Every character is represented by a binary value
  - Uppercase letters (capitals) have different values from lowercase characters
  - Punctuation symbols have their own character code
- How many characters are there on a standard keyboard?
  - How many bits would be required to represent this many combinations?

Characters in binary

Answers

- A keyboard needs to contain
  - 26 lowercase letters
  - 26 uppercase letters
  - 10 numbers
  - (around) 36 other characters
- There are around 98 unique characters that are available on a keyboard
  - 6 bits give 64 different combinations – this isn't enough
  - 7 bits give 128 different combinations which can represent 128 different characters

Character sets

- A character set is a set of letters, symbols and digits that can be represented by a computer
- There are two major character sets in use today
  - ASCII
  - Unicode

The ASCII character set

- **ASCII** (American Standard Code for Information Interchange) has become the standard code, used worldwide
  - It was originally developed in the 1960s for representing the English alphabet
  - It encodes 256 characters into 8-bit binary codes
- Characters include numbers 0 to 9, uppercase and lowercase letters A-Z, a-z, punctuation symbols and the space character

The ASCII character set

- What happens if you press ***ALT+65*** on a keyboard?
  - What character is represented by 0100000 (32)?
- What is the ASCII character for the number 7? Is this the same as the binary value for 7?
  - Why not? What is happening? What does this mean?

Decimal	Binary	Character	Decimal	Binary	Character	Decimal	Binary	Character
32	00100000	space	64	01000000	@	96	01100000	'
33	00100001	!	65	01000001	A	97	01100001	a
34	00100010	"	66	01000010	B	98	01100010	b
35	00100011	£	67	01000011	C	99	01100011	c
36	00100100	$	68	01000100	D	100	01100100	d
37	00100101	%	69	01000101	E	101	01100101	e
38	00100110	&	70	01000110	F	102	01100110	f

ASCII groups and sequences

- Character codes are commonly grouped and run in sequence
  - Numeric characters 0 to 9 run consecutively from 48 to 57 on the ASCII table
- A-Z characters are from 65-90 or 01000001 to 01011010
  - What range does lowercase characters a-z use?
  - If you know Capital A is 65 or 01000001, what is Capital E?

ASCII character set

Answers

- ASCII character 32 (010 0000) represents a space
- The ASCII character code for ‘7’ is 55
  - 55 (011 0111) is the ASCII character code that represents the character ‘7’
  - In programming this is very different to the integer 7 which is represented by 0000 0111 (7)
- Lowercase characters a-z use 97-122
- If A is 65 (0100 0001) then E is 69 (0100 0101)

Bit length and characters

- Numerous different codes for representing characters have been created, but ASCII is commonly used on PCs
- Originally only seven bits were used, but now an eighth bit is used allowing for many more characters such as ©, ® etc.
  - How many different characters can be encoded using 7-bit, 8-bit or 16-bit character sets?

Character codes

Answers

- A 7-bit character set (like ASCII) has 128 different characters that can be encoded
  - An 8-bit character set (like extended ASCII) has 256 different characters that can be encoded
  - A 16-bit character set has 65 536 different characters that can be encoded

Using the eighth bit

- Sometimes it is useful to be able to type special characters like á, à, ®
- Here are the codes for some of them:

©	Alt+0169
®	Alt+0174
á	Alt+0225
à	Alt+0224
â	Alt+0226
ä	Alt+0228

- Try out these different character codes

Worksheet 3

- Complete **Task 1** and **Task 2** on **Worksheet 3**

Programming with text and numbers

- The ASCII code for ‘7’ is 011 0111
  - The binary code for the digit 7 is 0000 0111
- When you write a program in Python, for example, you have to specify whether a variable is text or integer
  - You cannot do arithmetic with characters
  - If the character represents a number it must first be converted to an integer before any arithmetic can be carried out

Working with string input

- In Python, two strings can be concatenated, or joined together, using the + symbol

```
firstname = input("Please input your first name: ")
secondname = input("Please input your second name: ")
fullname = firstname + " " + secondname
print("Your full name is " + fullname)
```

- If you enter **Mike** for a first name and **Bell** for a second name, the computer will display

```
Your full name is Mike Bell
```

ASCII representation of numbers

- Try typing ***ALT + 55***
- What is the binary representation of the ASCII character 7? Is this the same as the binary value for 7?
  - Why not? What does this mean?

Decimal	Binary	Character	Decimal	Binary	Character	Decimal	Binary	Character
48	00110000	0	53	00110101	5	58	00111010	:
49	00110001	1	54	00110110	6	59	00111011	;
50	00110010	2	55	00110111	7	60	00111100	<
51	00110011	3	56	00111000	8	61	00111101	=
52	00110100	4	57	00111001	9	62	00111110	>

Converting ASCII to pure binary

- Clearly, we cannot do arithmetic with ASCII characters
- Programming languages deal with the input of numbers in different ways
- In some languages, variables have to be declared as type char, string, integer, real etc. at the beginning of the program
  - In other languages such as Python, all data is input as string, and if it is to be regarded as an integer, it has to be converted using an built-in function

e.g.

```
xString = input ("Enter an integer: ")
x = int(xString)
```

Using different alphabets

- To represent other characters for different languages, a new code allowing for many more characters is needed
  - **Unicode** was developed to use 16 bits, 65 536 possible combinations
  - The 32-bit version gives 4 294 967 296 (over 4 billion) possible combinations

Unicode

- In Japanese, konnichiwa is used as a greeting meaning “good day”
- In Unicode this is written as three 16-bit characters

今日は

- How many bytes does the English ‘good day’ require in ASCII?
- How many bytes does the Japanese require in Unicode?

Unicode

Answers

- ‘good day’ requires 8 bytes to store
- 今日は requires 6 bytes to store (3 characters x 2 bytes)
- Unicode is also used to store emoji
  - ‘e’ is Japanese for picture
  - ‘moji’ is Japanese for character or alphabet

Smiling face with sunglasses
Unicode: 1F60E

Worksheet 3

- Complete **Task 3** on **Worksheet 3**

Plenary

- Work in a pair to answer the following questions
  - How many bits are in extended ASCII?
  - How many characters does this allow for?
  - How many bytes are used for a Unicode character?
  - If ‘f’ has the ASCII code 102, what is the ASCII code for ‘g’?
  - How many bytes are needed to store “Hello everyone.”?

Plenary

Answers

- How many bits are in extended ASCII? 8 bits
- How many characters does this allow for? 256
- How many bytes are used for a Unicode character?
  16-bit has 2 bytes, 32-bit has 4 bytes
- If ‘f’ has the ASCII code 102, what is the ASCII code for ‘g’?
  103
- How many bytes are needed to store “Hello everyone.”?
  15 characters (remember space and the full stop)
  This requires 15 bytes assuming that 8-bit ASCII is used

Copyright

© 2024 PG Online Limited

The contents of this unit are protected by copyright.

This unit and all the worksheets, PowerPoint presentations, teaching guides and other associated files distributed with it are supplied to you by PG Online Limited under licence and may be used and copied by you only in accordance with the terms of the licence. Except as expressly permitted by the licence, no part of the materials distributed with this unit may be used, reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic or otherwise, without the prior written permission of PG Online Limited.

Licence agreement

This is a legal agreement between you, the end user, and PG Online Limited. This unit and all the worksheets, PowerPoint presentations, teaching guides and other associated files distributed with it is licensed, not sold, to you by PG Online Limited for use under the terms of the licence.

The materials distributed with this unit may be freely copied and used by members of a single institution on a single site only. You are not permitted to share in any way any of the materials or part of the materials with any third party, including users on another site or individuals who are members of a separate institution. You acknowledge that the materials must remain with you, the licencing institution, and no part of the materials may be transferred to another institution. You also agree not to procure, authorise, encourage, facilitate or enable any third party to reproduce these materials in whole or in part without the prior permission of PG Online Limited.

OCR
Oxford Cambridge and RSA
An OCR endorsed teaching and learning tool

GCSE OCR

Computer Science
J277

4

Images

Unit 2
Data representation

PG ONLINE

Objectives

- Understand how a bitmap graphic is made up of individual pixels
  - Explain how each pixel is represented in binary
  - Understand that the number of bits per pixel determines the number of available colours for an image
- Explain the need for image metadata
- Explain the relationship between file size and image resolution

Starter

- Digital images can be stored on a computer in many different file formats
  - What are five different file formats that can be used?

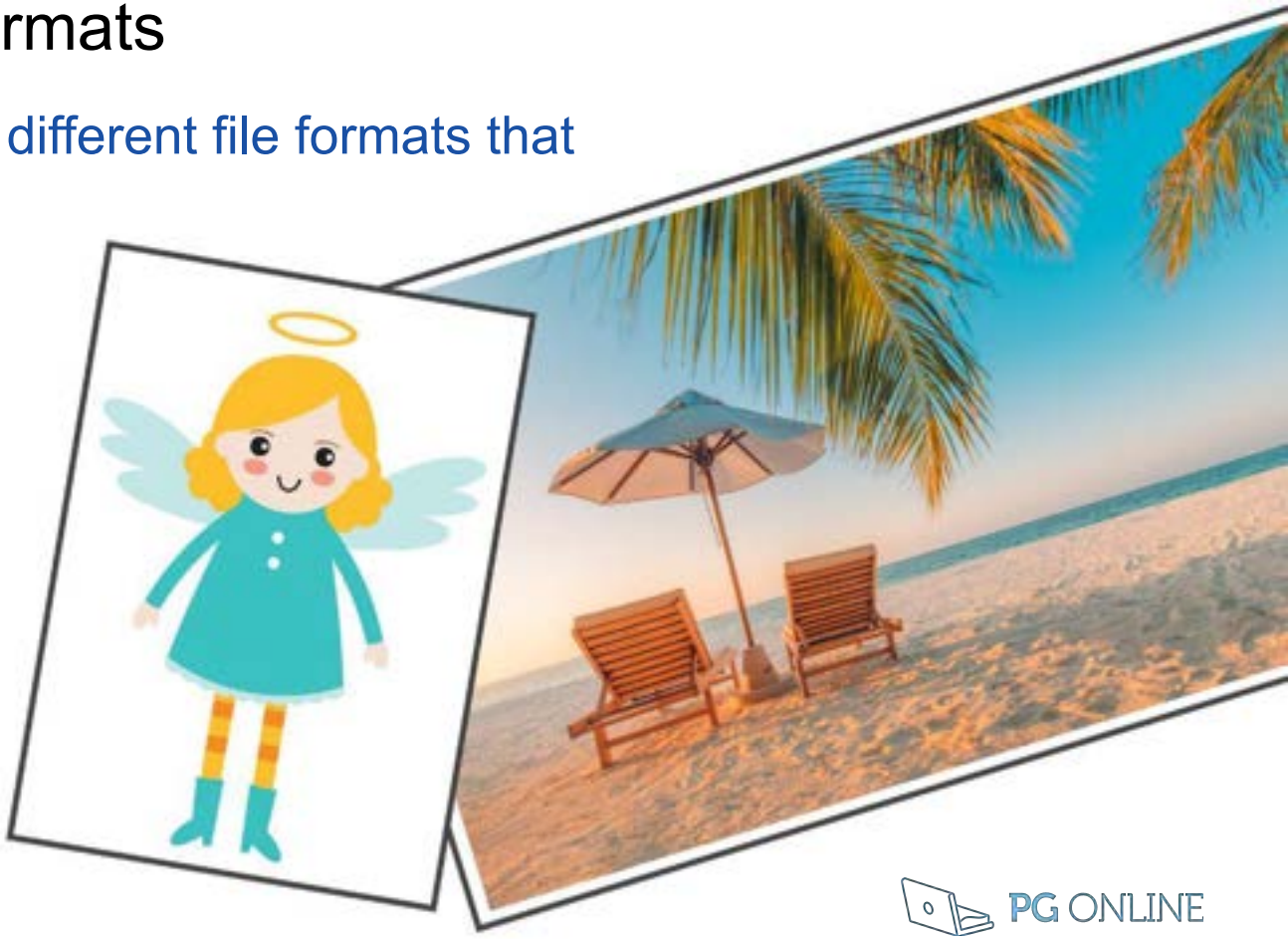


Image file types

Answers

- Bitmap image file formats:
  - BMP
  - JPG
  - GIF
  - PNG
  - TIFF
- Vector image file format:
  - SVG
- There are also editing formats such as Illustrator (AI) and Photoshop (PS)

Bitmap (or Raster) images

- Bitmap images are made up of **PIC**ture **EL**ements or **pixels**
  - A pixel is the smallest identifiable area of an image
  - Each pixel is a single colour and is given a binary value which represents that colour e.g. 111100000000 might equal red
  - A pixel's colour can be changed by changing this value

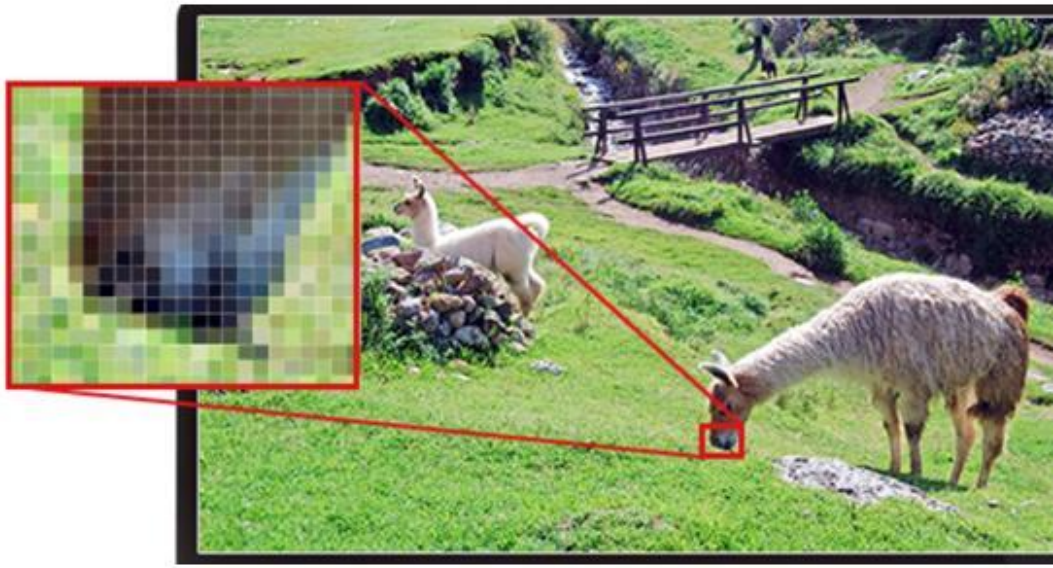
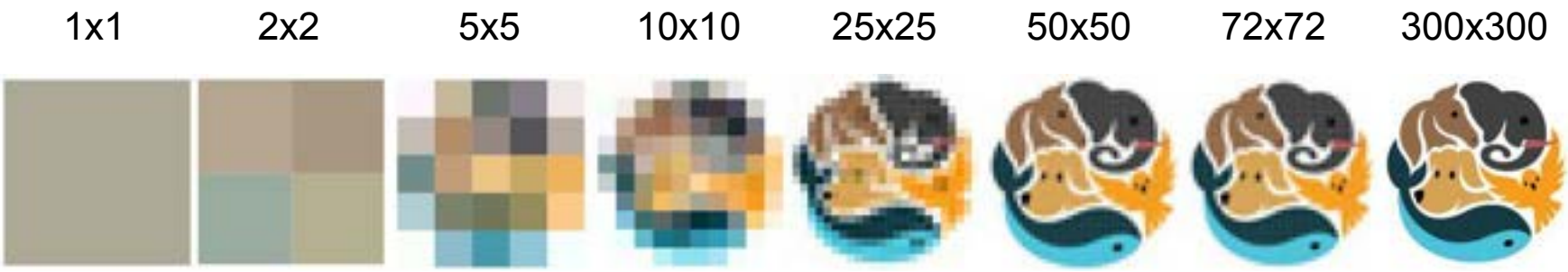


Image resolution

- Resolution is the concentration of pixels within a specific area
- The area is defined by the image width and height in pixels e.g. 1920x1080
  - 72 PPI (Pixels Per Inch) = typical screen resolution
  - 300 DPI (Dots Per Inch) = print quality resolution
  - Smartphones may have very high resolutions 300+ PPI



Creating an Image

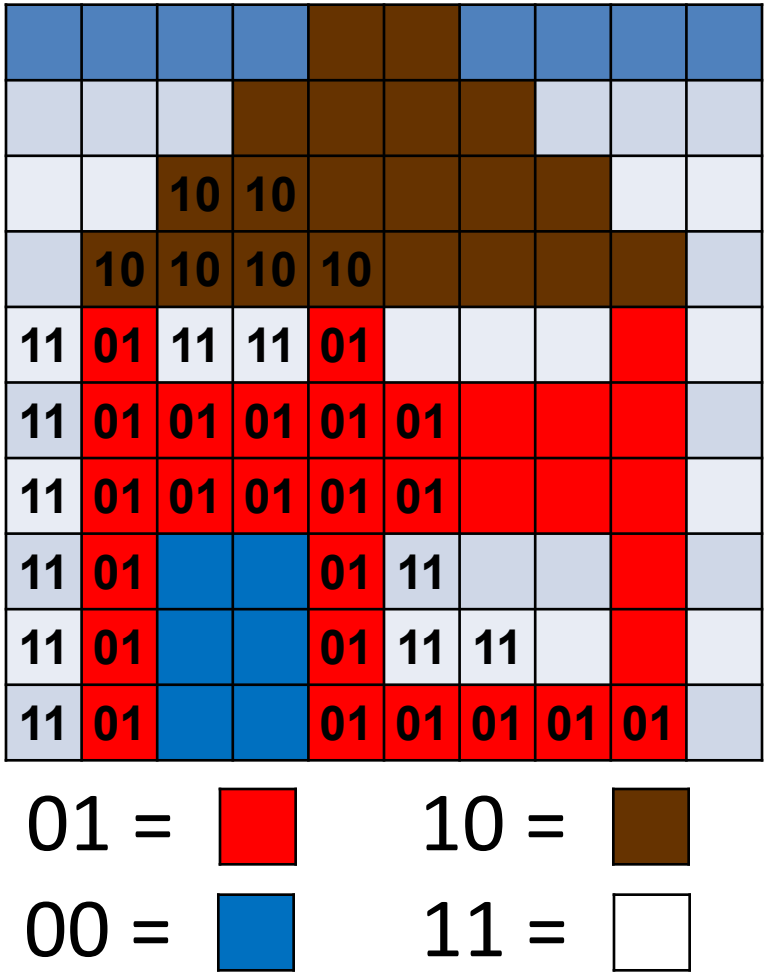
- Each pixel is given a binary value
- Each value represents a different colour
- Using one bit per pixel allows only 2 values, 0 and 1

  1 = Black, 0 = White

0	0	0	0	1	0	0	0	0	0
0	0	0	1	1	0	0	0	0	0
0	0	1	1	1	0	1	0	0	0
0	1	1	1	1	0	1	1	0	0
1	1	1	1	1	0	1	1	1	0
0	0	0	0	1	0	1	0	0	0
1	1	1	1	1	1	1	1	1	1
0	1	1	1	1	1	1	1	1	0
0	0	1	1	1	1	1	1	1	0
0	0	0	0	0	0	0	0	0	0

Increasing the number of colours

- More bits per pixel = more colour combinations
  - 1 bit = 2 colours
  - **2 bits = 4 colours**
  - 3 bits = 8 colours
  - 4 bits = 16 colours
- How many bits per pixel required for 256 colours?



Colour or colour depth

- Each pixel can represent a finite number of colours
  - Each pixel is attributed a number of bits, *n*
  - The number of bits dictates the colour depth (bit depth) with the number of colours that can be represented being 2^n
  - A higher colour depth gives a greater range of colours and a better quality of image

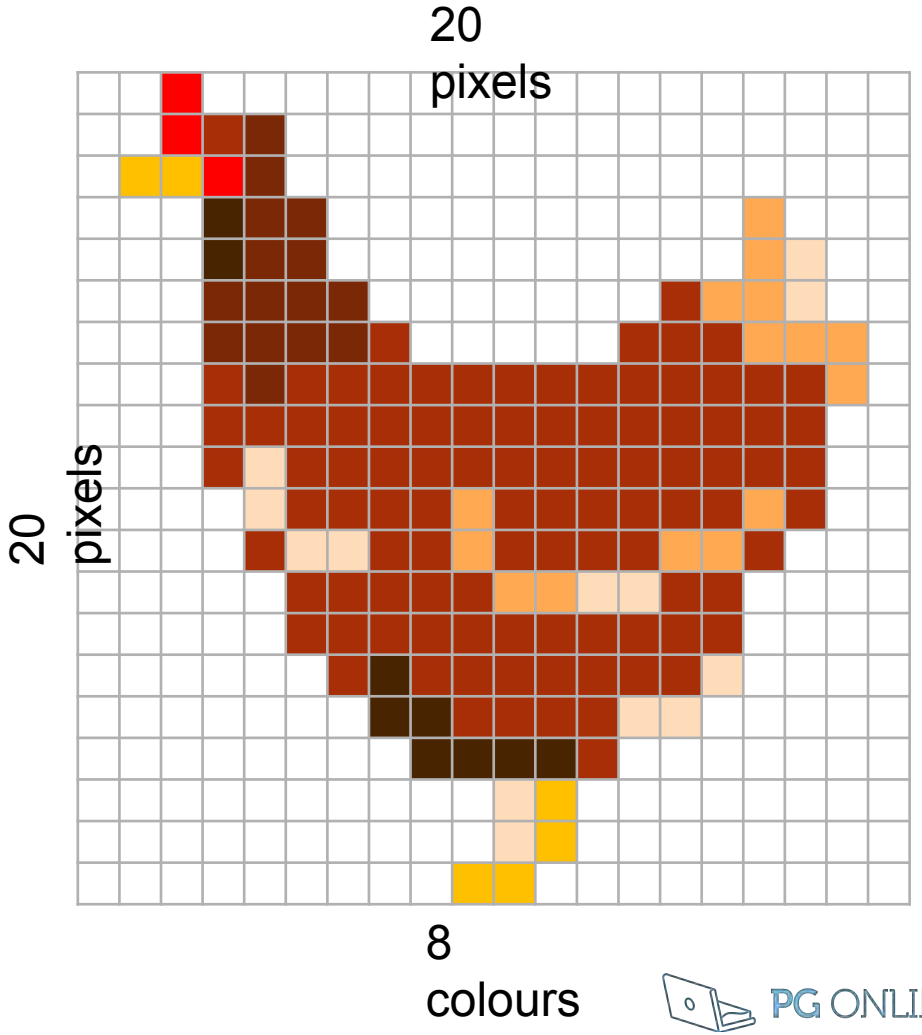
8 bits per pixel = 2^8 = 256 colours

16 bits per pixel = 2^{16} = 65,536 colours

24 bits per pixel = 2^{24} = 16,777,216 colours

Colours and resolution vs File Size

- How does the number of colours affect file size?
- How does the size of the image affect file size?



Effects of varying colour depth

- Changing the colour depth of an image will affect the number of colours it can display, as shown below:

2 Colours | 4 Colours | 8 Colours | 16 Colours | 256 Colours | 16.7m Colours

PG ONLINE

Worksheet 4

- Complete **Task 1** on **Worksheet 4**

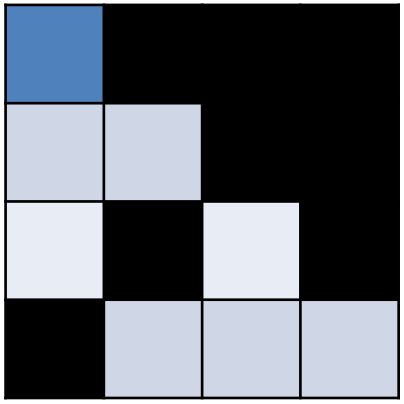
PBM monochrome images

- Store image dimensions
- Change ‘colours’ by changing binary values

```
10 10
0 0 0 0 1 0 0 0 0 0
0 0 0 1 1 0 0 0 0 0
0 0 1 1 1 0 1 0 0 0
0 1 1 1 1 0 1 1 0 0
1 1 1 1 1 0 1 1 1 0
0 0 0 0 1 0 1 0 0 0
1 1 1 1 1 1 1 1 1 1
0 1 1 1 1 1 1 1 1 0
0 0 1 1 1 1 1 1 1 0
0 0 0 0 0 0 0 0 0 0
```

Representing a monochrome image in binary

- A monochrome image can be represented in binary as a bit pattern
  - The following icon can be represented by the corresponding bit pattern:

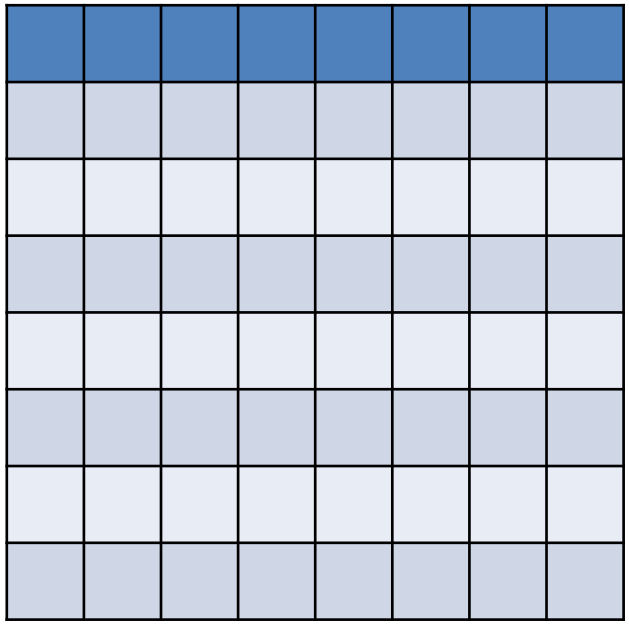


```
0111,
0011,
0101,
1000
```

Converting a bit pattern into a monochrome image

- Convert the following bit pattern into an 8x8 icon:

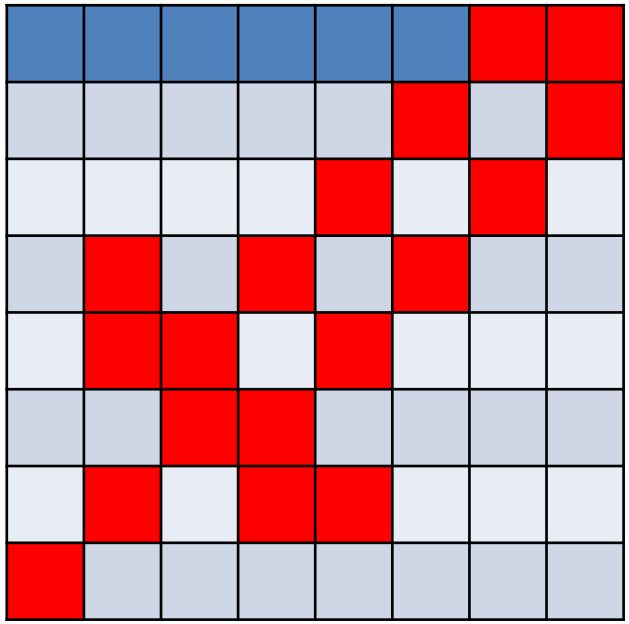
```
00000011
00000101
00001010
01010100
01101000
00110000
01011000
10000000
```



Converting a bit pattern into a monochrome image

- Convert the following bit pattern into an 8x8 icon:

```
00000011
00000101
00001010
01010100
01101000
00110000
01011000
10000000
```

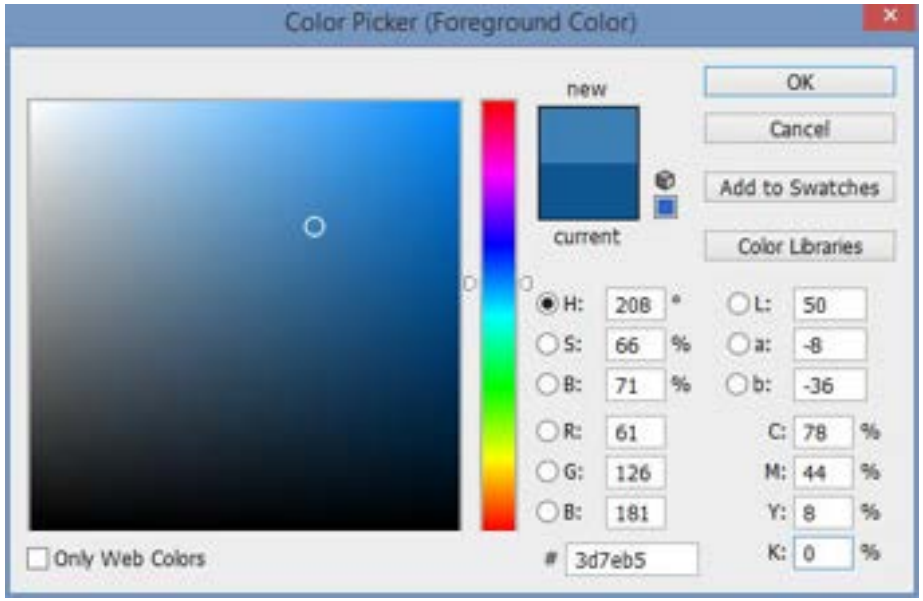


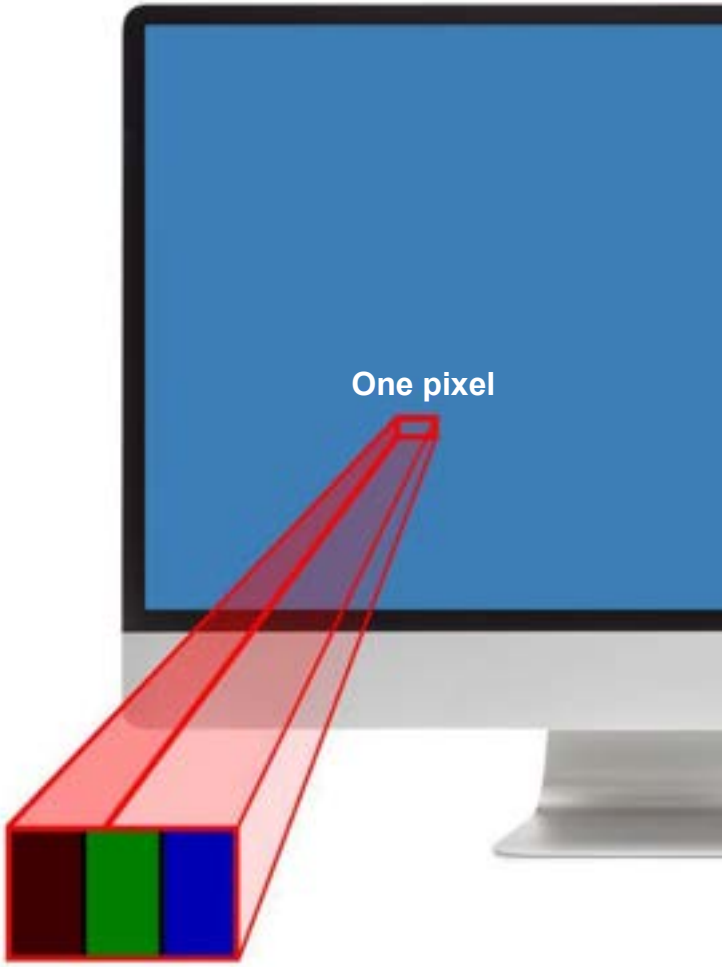
Worksheet 4

- Complete **Task 2** on **Worksheet 4**

Colours

- Each pixel has a proportion of red, green and blue
  - From a distance, it makes up one colour





61 126 181
Red Green Blue

Looking at colour values

- Colour values of individual pixels are expressed in software as denary RGB values or in hexadecimal
  - Why isn't binary used in this instance?
- RGB (Red, Green and Blue) values range between 0-255
  - How many bits are required for each of the red, green and blue components of a pixel?
  - How many bits are required for the whole pixel?
  - In 32-bit colour what are the last 8 bits for?

Looking at colour values

Answers

- Software colour pickers are used by designers and developers
  - They use denary and hex values as these are easier for people to read, write and remember and reduce errors if copied
- RGB values between 0-255
  - Each colour requires 8 bits to store the range 0-255
  - The whole pixel requires 3*8 = 24 bits (16.8 million colours)
  - In 32 bits per pixel 8 bits are given to a transparency channel (to allow a background to show through)

Why file sizes don’t always add up

- DucksBMP has a resolution of 1000 x 750px
- The colour depth is 24 bit = 16 million colours
  - What is the file size in bytes and MB?
  - Why is there a difference in the actual file size of 56 bytes?

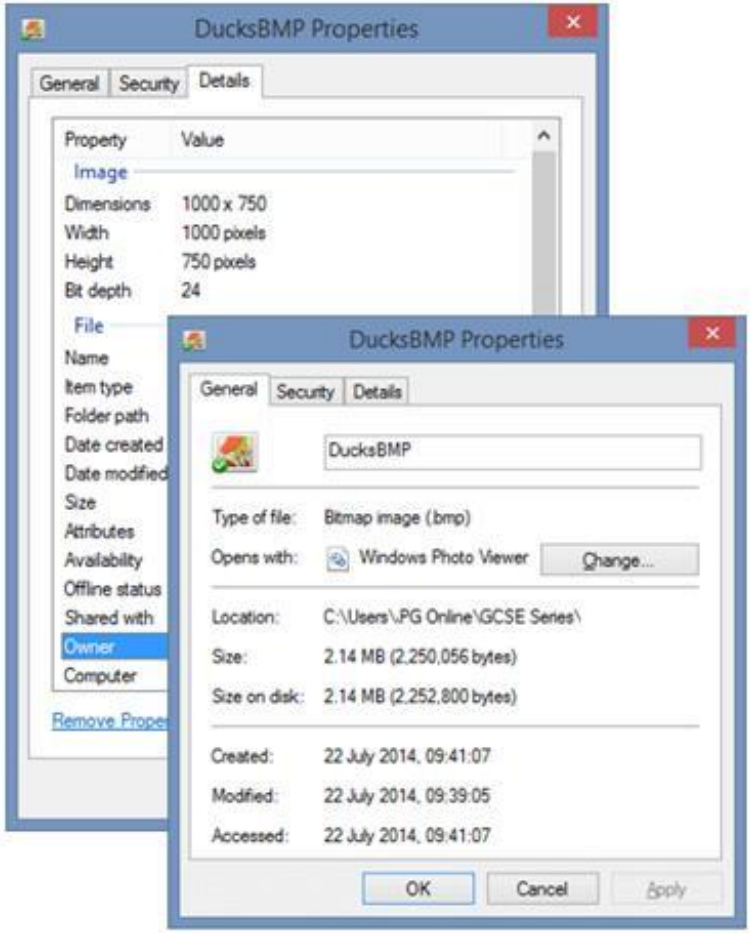


Image metadata

- Metadata is data about data
- It is information other than image data that is stored with a file
- This will include:
  - Colour depth in bits per pixel
  - Resolution (height and width in pixels)
  - Date created
  - Author
- Why would the file DucksJPG be a different file size?

Plenary

- Fill in the gaps in each of the below sentences:
  - A bitmap graphic is made up of _____
  - Each _____ is represented in binary
  - The _____ determines the number of available colours for an image
  - Image _____ holds data about the image such as
  - _____
  - The greater the _____ of the image, the greater the file size

Plenary

Answers

- A bitmap graphic is made up of pixels
- Each pixel is represented in binary
- The number of bits per pixel determines the number of available colours for an image
- Image metadata holds data about the image, such as colour depth, resolution, date created, author
- The greater the resolution/colour depth of the image, the greater the file size

Copyright

© 2024 PG Online Limited

The contents of this unit are protected by copyright.

This unit and all the worksheets, PowerPoint presentations, teaching guides and other associated files distributed with it are supplied to you by PG Online Limited under licence and may be used and copied by you only in accordance with the terms of the licence. Except as expressly permitted by the licence, no part of the materials distributed with this unit may be used, reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic or otherwise, without the prior written permission of PG Online Limited.

Licence agreement

This is a legal agreement between you, the end user, and PG Online Limited. This unit and all the worksheets, PowerPoint presentations, teaching guides and other associated files distributed with it is licensed, not sold, to you by PG Online Limited for use under the terms of the licence.

The materials distributed with this unit may be freely copied and used by members of a single institution on a single site only. You are not permitted to share in any way any of the materials or part of the materials with any third party, including users on another site or individuals who are members of a separate institution. You acknowledge that the materials must remain with you, the licencing institution, and no part of the materials may be transferred to another institution. You also agree not to procure, authorise, encourage, facilitate or enable any third party to reproduce these materials in whole or in part without the prior permission of PG Online Limited.

OCR
Oxford Cambridge and RSA
An OCR endorsed teaching and learning tool

GCSE OCR

Computer Science
J277

Sound

Unit 2
Data representation

PG ONLINE

5

Objectives

- Understand how sound is sampled and stored in digital form
- Be able to represent a short sound file in binary
- Explain how sampling intervals and resolution affect the size of a sound file using the terms:

  Sample rate

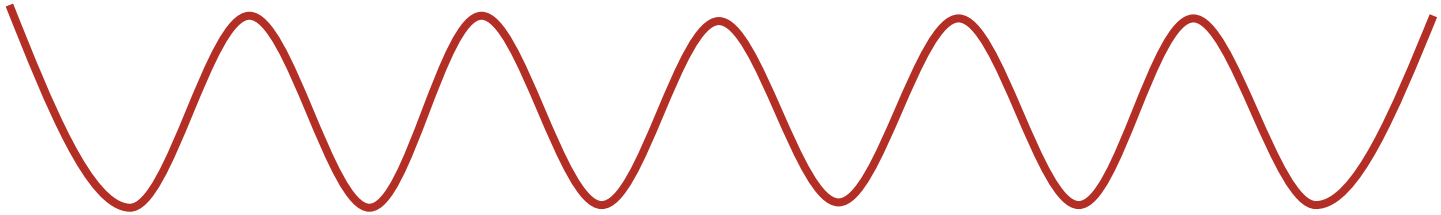
  Bit depth
- Explain the trade-off between file size and the quality of playback

Starter

- A computer uses binary 1s and 0s
  - How do you think this is able to store sound?

Analogue to digital conversion

- Analogue sound signals are continuous



- Digital signals are discrete

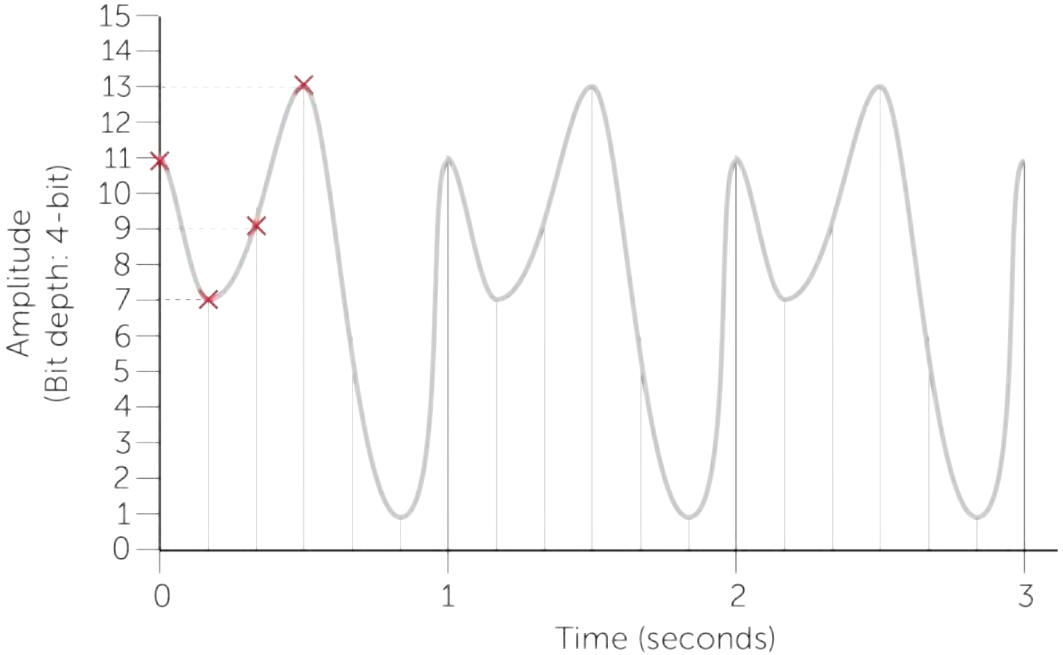
- Sound is digitised by repeatedly measuring and recording the sound wave

Analogue to digital convertors

- Sounds must be converted into a digital form in order to be stored and processed by a computer
  - An Analogue to Digital Convertor (ADC) is used to convert inputs to digital signals
  - A Digital to Audio Convertor (DAC) is used to convert digital signals to outputs

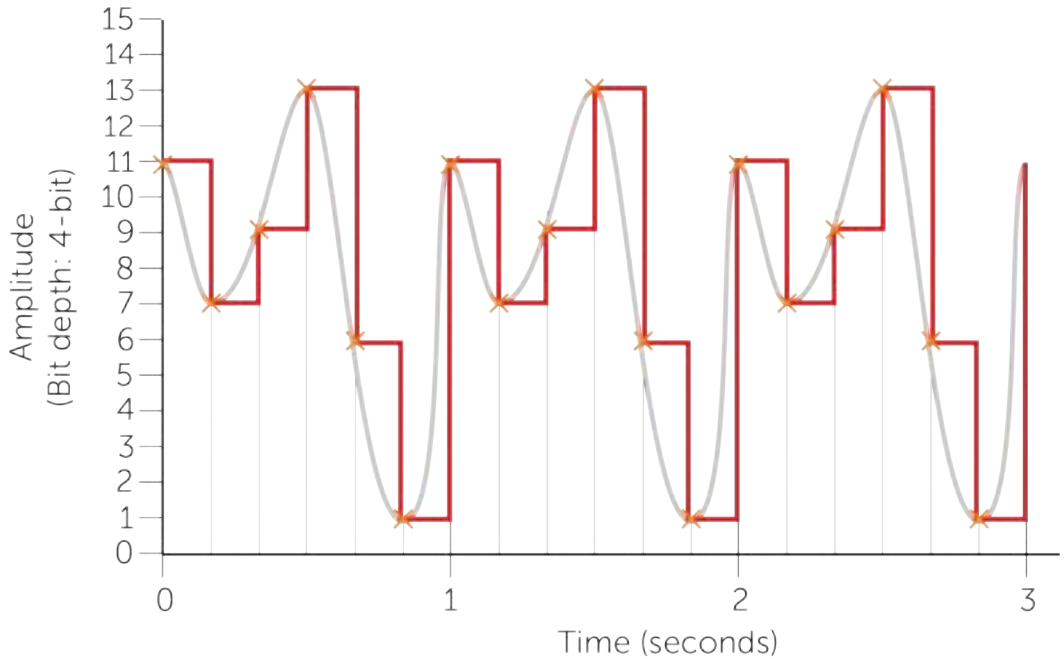
Sound sampling

- A sound sample is a measurement of amplitude of a sound wave at a given time
  - Sound is sampled using a bit depth and sample rate
  - The **bit depth** determines how closely the wave is sampled on the y-axis



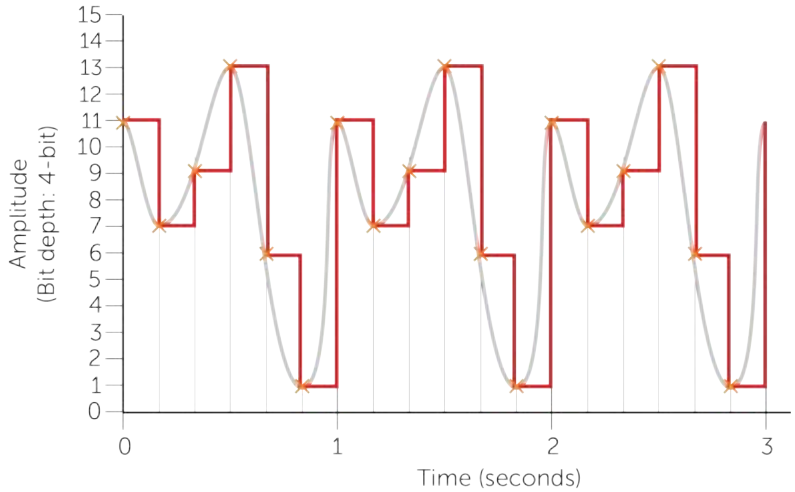
Sound sampling

- The sample rate is the number of samples taken per second
  - It is measured in hertz (Hz)

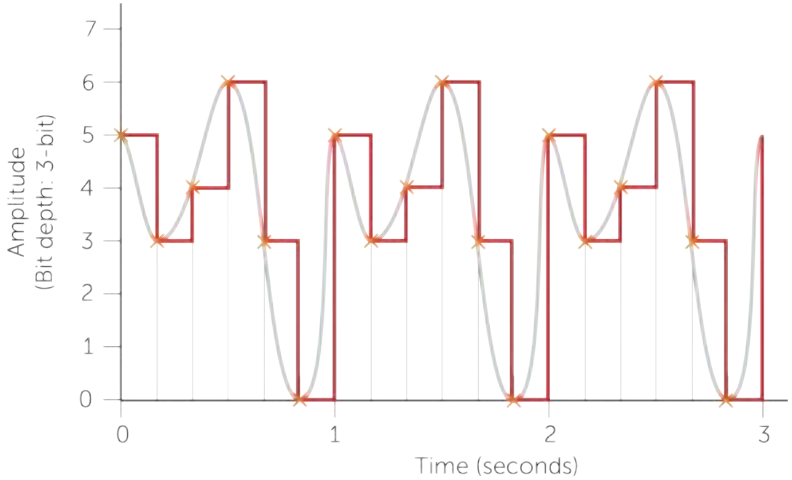


Bit depth

- The number of bits used to record each measurement is known as the bit depth
  - More bits used per sample enables the height of the wave to be more accurately measured but increases file size



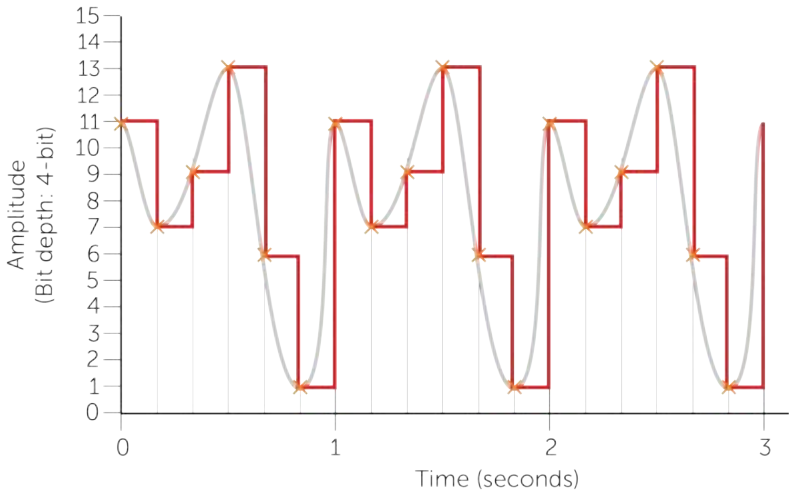
Higher (4-bit) bit depth



Lower (3-bit) bit depth

Hertz

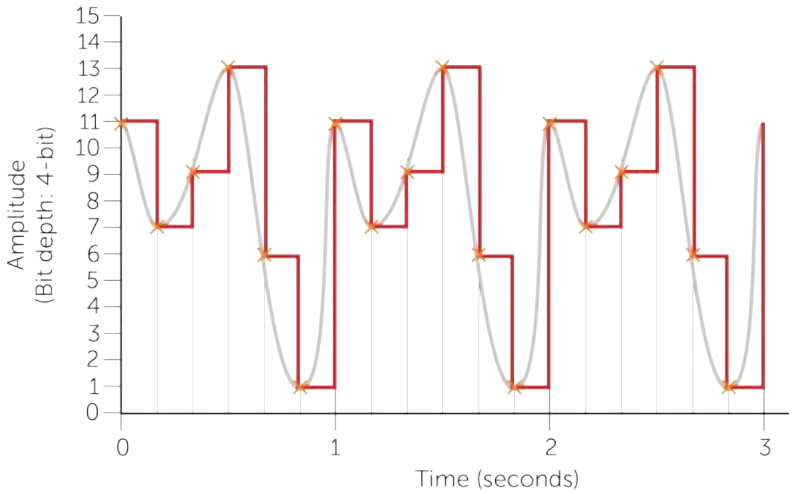
- Sample rate is usually measured in Hertz (Hz)
- 1Hz = 1 sample per second
- CD-quality music is sampled at 44,100Hz (44.1kHz)
  - What is the sample rate in Hertz of the following sample?



Hertz

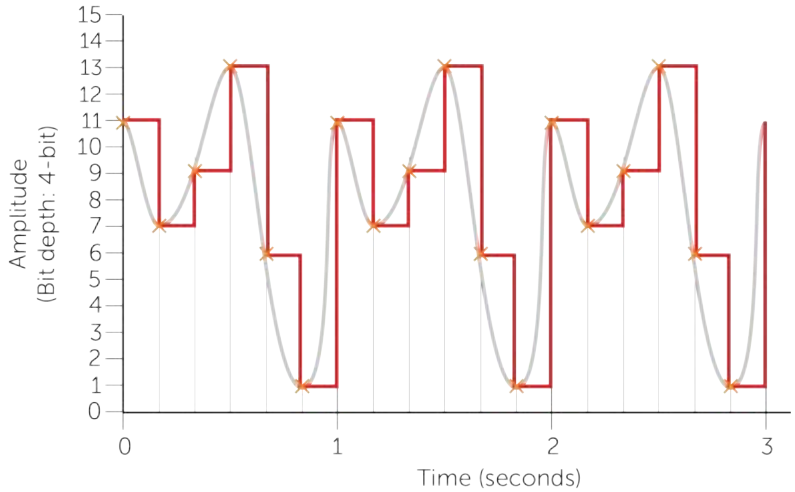
Answers

- What is the sample rate in Hertz of the following sample?
  - The sample rate is 6 Hertz
    (6 samples are taken in each second)

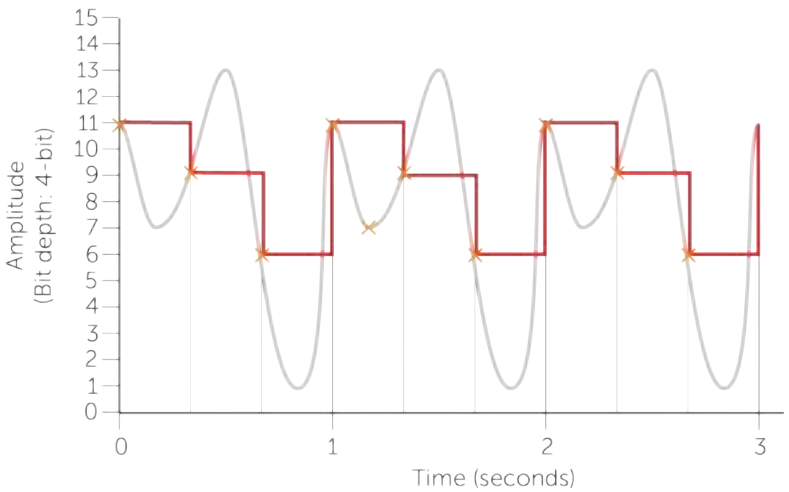


Sampling rate

- The frequency or sample rate per second affects the level of detail in the digital representation
- The greater the frequency, the greater the accuracy, and file size

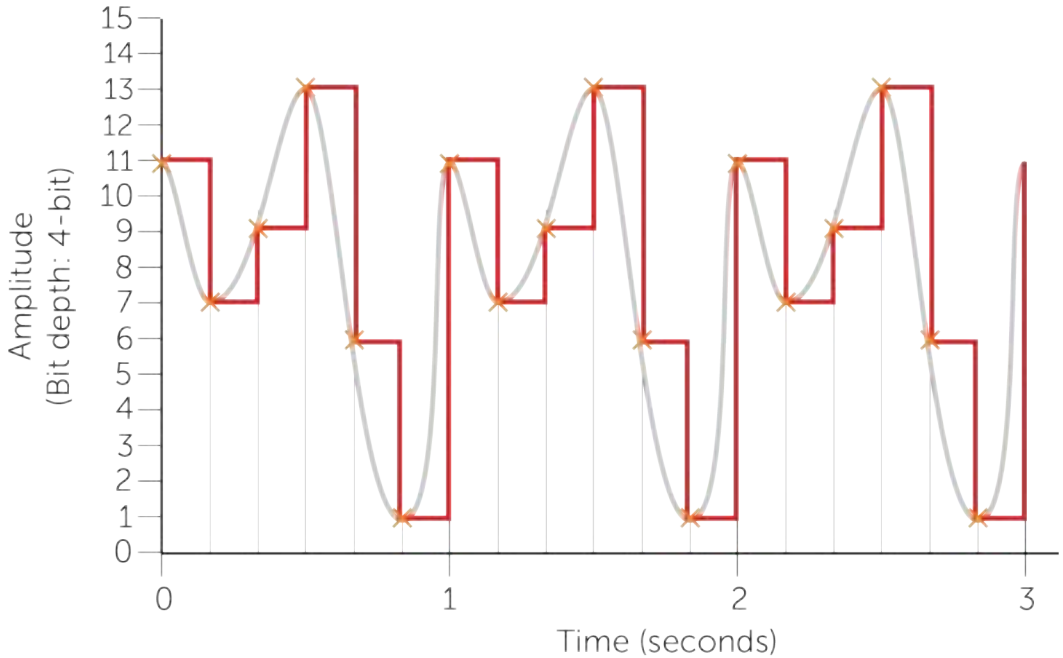


Higher frequency



Lower frequency

Calculating sound file sizes



- File size (bits) = sample rate x bit depth x duration
  - 6 samples per second x 4-bit bit depth x 3 seconds =
  - 72 bits / 8 = 9 bytes

Worksheet 5

- Complete **Task 1** on **Worksheet 5**

Digitised sound quality

- Recording quality improves:
  - the more frequently we sample the sound
  - the more accurately we record the wave height
- Increasing the **sampling rate** (frequency) means recording more data points
- Increasing the **bit depth** improves the accuracy of each data point
  - What happens to the size of the sound file if the frequency and bit depth are increased?

Worksheet 5

- Complete **Task 2** on **Worksheet 5**

Our hearing range

- We can hear sounds between 20-20,000 Hz
  - Younger people can hear sounds at a higher frequency
- Try the test sound and see how high you and your class can hear

PG ONLINE

Lossy compression – MP3

- Lossy compression removes sounds that we can't easily hear or that least affect the perceived playback quality
  - Lossy compression leaves out some data from the original so can negatively affect the sound quality
  - However, a minute of music can be stored in 1 MB of an MP3 file, but needs 10 MB of an uncompressed WAV file
  - This is useful for storing, downloading or streaming

PG ONLINE

Lossless compression

- Lossless compression formats are able to reduce the file size when compressed but do not lose any information
- The following music file format are lossless:
  - FLAC (Free Lossless Audio Codec
  - ALAC (Apple Lossless Audio Code
  - WMA Lossless (Windows Media Audio)

Other file formats

- **WAV** and **AIFF** files are uncompressed audio files
- Lossy file formats are still very popular as they reduce the file size well
  - **MP3** (MPEG-1 Audio Layer III)
  - **AAC** (Advanced Audio Coding) – typically higher quality than MP3

PG ONLINE

Plenary

- In pairs take turns to test each other on the following questions
  - The sample rate is measured in what unit?
  - The number of bits available to store each sample is known as what?
  - What is the formula for calculating the size of an uncompressed music file?
  - Track1.wav uses a sample rate of 20 kHz, Track2.wav uses a sample rate of 10 kHz. Which file is the higher quality? Which is the larger file size?

Plenary

Answers

- In pairs take turns to test each other on the following questions
  - The sample rate is measured in what unit? **Hertz (Hz)**
  - The number of bits available to store each sample is known as what? **Bit depth**
  - What is the formula for calculating the size of an uncompressed music file?
    Sample rate x bit depth x duration
  - Track1.wav uses a sample rate of 20 kHz, Track2.wav uses a sample rate of 10 kHz. Which file is the higher quality? Which is the larger file size?
    Track1 is higher quality and a larger file size
    (assuming the same bit depth and track length)

Copyright

© 2024 PG Online Limited

The contents of this unit are protected by copyright.

This unit and all the worksheets, PowerPoint presentations, teaching guides and other associated files distributed with it are supplied to you by PG Online Limited under licence and may be used and copied by you only in accordance with the terms of the licence. Except as expressly permitted by the licence, no part of the materials distributed with this unit may be used, reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic or otherwise, without the prior written permission of PG Online Limited.

Licence agreement

This is a legal agreement between you, the end user, and PG Online Limited. This unit and all the worksheets, PowerPoint presentations, teaching guides and other associated files distributed with it is licensed, not sold, to you by PG Online Limited for use under the terms of the licence.

The materials distributed with this unit may be freely copied and used by members of a single institution on a single site only. You are not permitted to share in any way any of the materials or part of the materials with any third party, including users on another site or individuals who are members of a separate institution. You acknowledge that the materials must remain with you, the licencing institution, and no part of the materials may be transferred to another institution. You also agree not to procure, authorise, encourage, facilitate or enable any third party to reproduce these materials in whole or in part without the prior permission of PG Online Limited.

OCR
Oxford Cambridge and RSA
An OCR endorsed teaching and learning tool

GCSE OCR

Computer Science
J277

Compression

Unit 2
Data representation

PG ONLINE

6

Objectives

- Explain the need for compression
- Describe the difference between lossy and lossless compression

Starter

- HD TV has a resolution of 1920px x 1080px
  - This results in 2.1 megapixels per frame
  - Each pixel uses 24-bit colour
  - This means each frame needs about 50 megabits
  - There are 25 frames per second resulting in a total amount of data being 1.25 gigabits per second
- How can we possibly stream this through internet connections that are typically under 70 Mb/s in the UK?

Compression techniques

- Compression is the name given to algorithms which reduce file sizes
  - Decompression is the process where compressed data is restored to its original format
- Compression is heavily used with sound, image and video files
- There are two types of compression:
  - Lossy compression (JPG, GIF, MP3)
  - Lossless compression (PNG, TIFF)

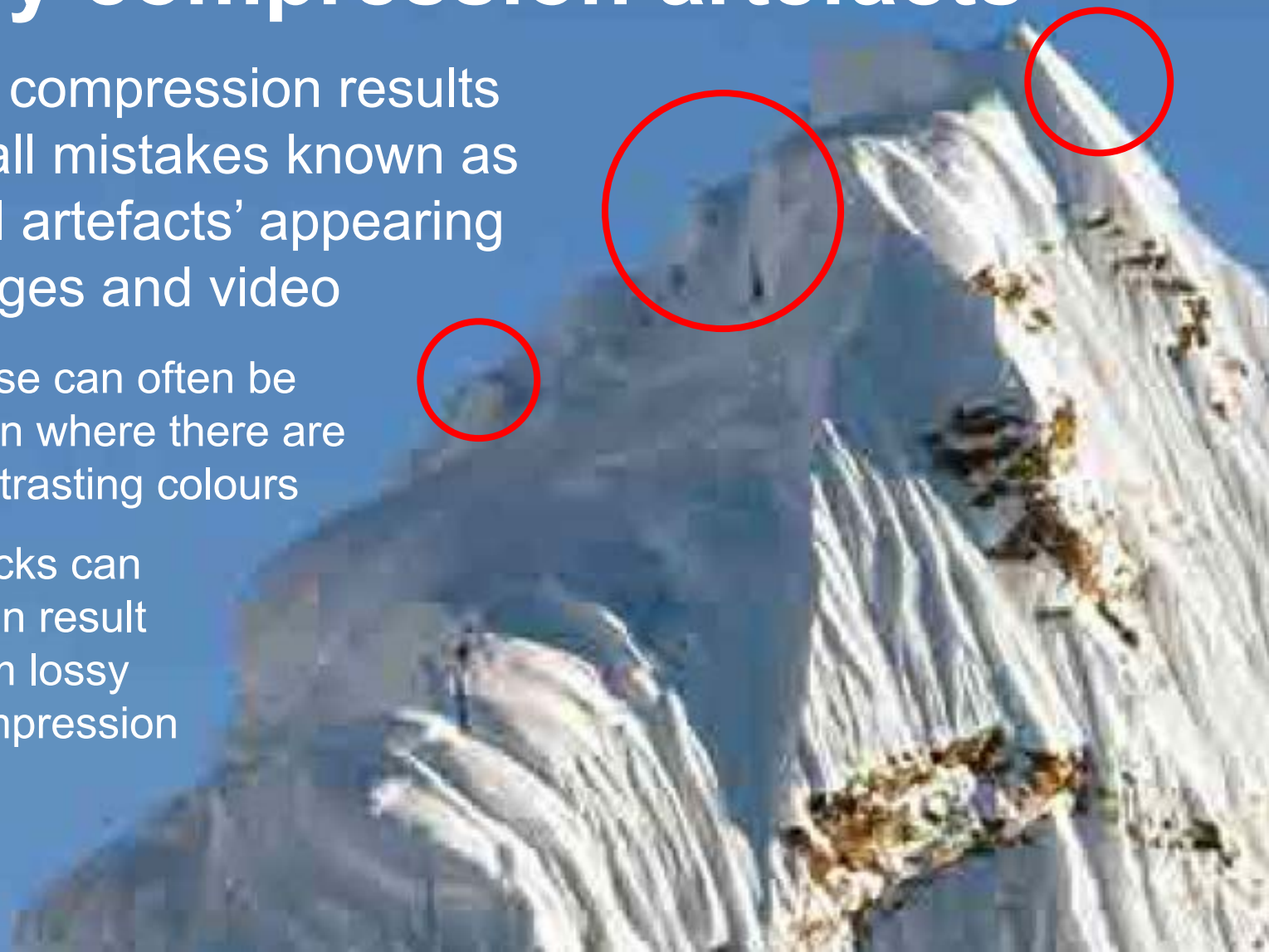
Lossy compression

- Lossy compression permanently loses some data
  - Would a lossy compression algorithm work for compressing a computer program?

```
# S CTIO  2
w ile gues    =    rr ctPas  or :
    guess = i put "Try to guess t      sswor    ")
    g   ses   gue  es + 1

 rint("Pa swor   uessed c r  ctly
```

Lossy compression artefacts

- Lossy compression results in small mistakes known as ‘digital artefacts’ appearing in images and video
  - Noise can often be seen where there are contrasting colours
  - Blocks can often result from lossy compression



Lossless image compression

- Lossless image compression will not lose any of the original data
  - One way to achieve this is using **run length encoding**. This finds groups of repeating data and records the data once along with the number of repetitions

= 12 x , 6 x

- When data is uncompressed it is restored exactly as it was in the original

Lossless text compression

- Finds patterns in the original text
- Encodes each pattern in a dictionary

An eye for an eye,

a tooth for a tooth.

.	0	0000
An_	1	0001
eye	2	0010
for	3	0011
an_	4	0100
,_	5	0101
a_tooth	6	0110

PG ONLINE

Dictionary compression

- 38 characters including spaces = 38 bytes (assuming an 8-bit ASCII table is used)
- 44 bits = 5.5 bytes = 14% of original size (plus codes)

.	0	0000
An_	1	0001
eye	2	0010
for	3	0011
an_	4	0100
,_	5	0101
a_tooth	6	0110

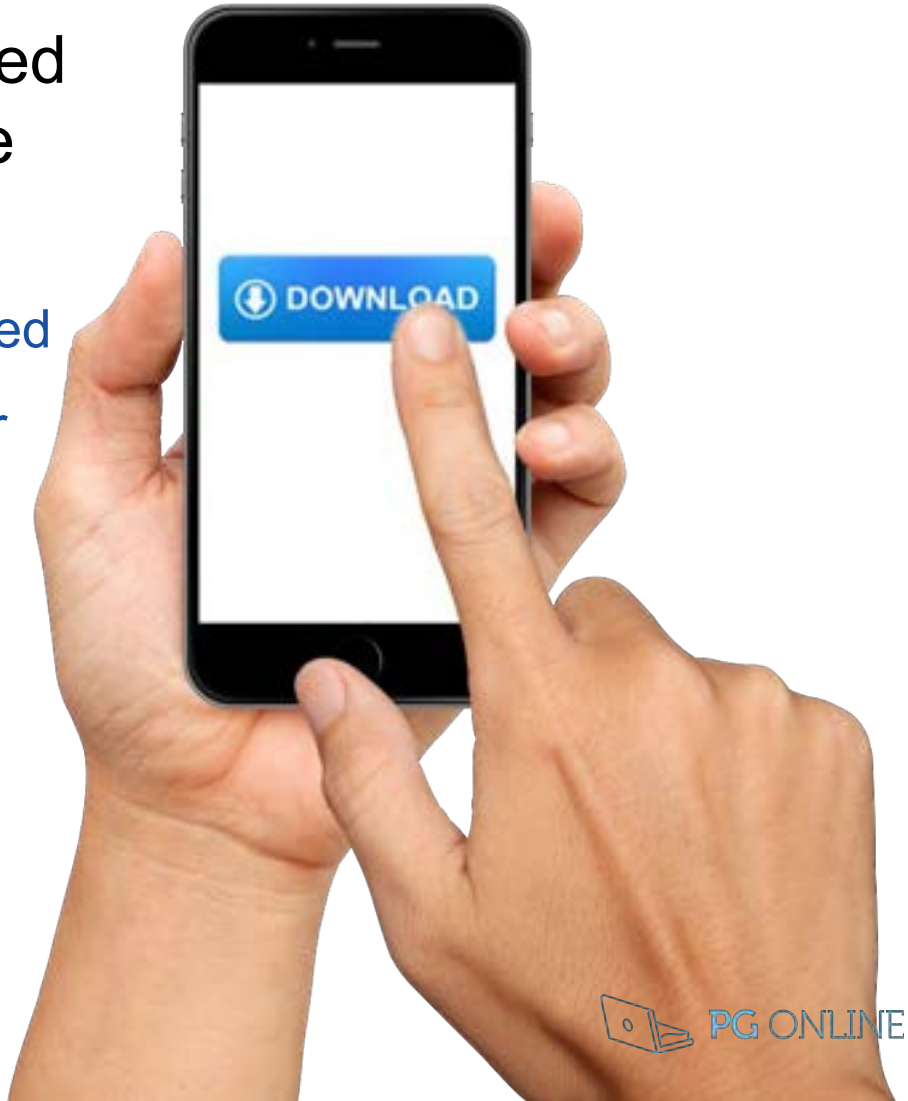
1	2	3	4	2	5	6	7	3	6	0
0001	0010	0011	0100	0010	0101	0110	0111	0011	0110	0000

Worksheet 6

- Complete **Task 1** on **Worksheet 6**

Using compression

- Compression is often used for files and data that are sent via the Internet
  - Download times are reduced
  - Data allowances go further
  - It is possible to transmit video and music data streams as fast as they are playing



PG ONLINE

Downloading a music track

- Dancing Queen by ABBA® = 3m 51sec = 231 seconds
- MP3 quality = 128kbps
- CD quality = 1411kbps
  - 231 x 128kbps =
    29,568kbs / 1024 / 8 = 3.6MB

  OR:

  - 231 x 1411kbps =
    325,941kbs / 1024 / 8 = 39.8MB
- 11.5 times faster with a compressed file
  - 36MB less download data used

Worksheet 6

- Complete **Task 2** on **Worksheet 6**

Benefits of compression

- Smaller files = fewer packets = faster transmission time
  - Quicker to complete transmission
  - Reduces traffic over the Internet
- Reduces download times of video, sound (including speech used for VOIP systems) and image files
  - Streaming is also possible as the data can be sent as fast as the rate it is played
- Images inside web pages appear faster
- Reduces space on disk / servers

Buffering

- Video or music streaming causes buffering if the download speed is slower than the playback speed

- How could you reduce the chances of people experiencing buffering when they are playing video?

Video streaming

Answers

- The files for video streaming are compressed
  - The TV or computer needs to decompress the data as it is playing the video
  - Before the video begins playing, it will buffer a sufficient amount to allow for the network occasionally being slow
  - If the website or server detects that the connection is slow, it may be able to start sending an alternative lower quality file

File formats

- What are five compressed file formats used for images, sound, video or documents?
  - Which are lossy and which are lossless?

Common file standards

Answers

File format	Type of compression	File usage
JPEG	Lossy	Photos
PNG	Lossless	Images/photos, may include transparency
ZIP	Lossless	Document compression
GIF	Lossless	Simple images, may include animation
MP4	Lossy	Video file format
MP3	Lossy	Music format
FLAC / ALAC	Lossless	Lossless Music format (Free / Apple)

Worksheet 6

- Complete **Task 3** on **Worksheet 6**

Plenary

- With a partner, take it in turns to explain the following:
  - The need for compression
  - Lossy compression
  - Some of the effects of lossy compression on images and sound files
  - Lossless compression

Plenary

Answers

- The need for compression
  - Compression reduces the size of files, which makes them smaller to store and transmit – this enables services such as streaming music and video
- Lossy compression
  - The compression technique will lose some of the original information
  - Noise at edges and blocks are digital artefacts from lossy compression
- Lossless compression
  - No data is lost by using a lossless compression algorithm

Copyright

© 2024 PG Online Limited

The contents of this unit are protected by copyright.

This unit and all the worksheets, PowerPoint presentations, teaching guides and other associated files distributed with it are supplied to you by PG Online Limited under licence and may be used and copied by you only in accordance with the terms of the licence. Except as expressly permitted by the licence, no part of the materials distributed with this unit may be used, reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic or otherwise, without the prior written permission of PG Online Limited.

Licence agreement

This is a legal agreement between you, the end user, and PG Online Limited. This unit and all the worksheets, PowerPoint presentations, teaching guides and other associated files distributed with it is licensed, not sold, to you by PG Online Limited for use under the terms of the licence.

The materials distributed with this unit may be freely copied and used by members of a single institution on a single site only. You are not permitted to share in any way any of the materials or part of the materials with any third party, including users on another site or individuals who are members of a separate institution. You acknowledge that the materials must remain with you, the licencing institution, and no part of the materials may be transferred to another institution. You also agree not to procure, authorise, encourage, facilitate or enable any third party to reproduce these materials in whole or in part without the prior permission of PG Online Limited.

OCR
Oxford Cambridge and RSA
An OCR endorsed teaching and learning tool

GCSE OCR

Computer Science J277

1

The Internet and wide area networks

Unit 3 Networks, connections and protocols

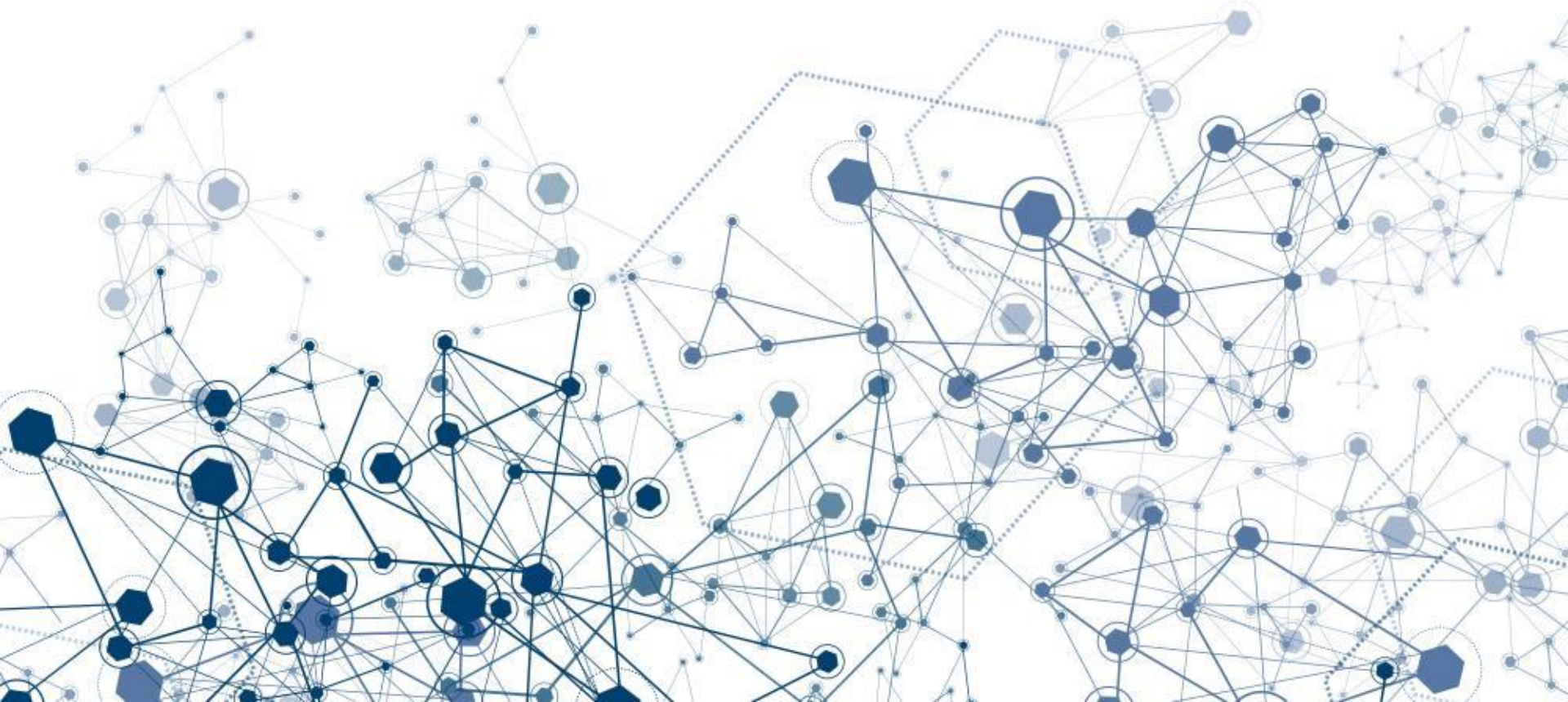
PG ONLINE

Objectives

- Define a Wide Area Network
- Describe the nature of the Internet as a worldwide collection of computer networks
- Explain the need for IP addressing of resources on the Internet and how this can be facilitated by the role of DNS
- Understand the need for Network Interface Cards and the uses of MAC addressing
- Explain packet switching

Starter

- What is the Internet?
  - How is it different from the World Wide Web?



Starter

Answers

- The Internet is a collection of **inter**-connected **net**works
  - Many services make use of the Internet such as the web, voice over IP (VoIP), email and file sharing
- The World Wide Web is one service that makes use of the Internet
  - The World Wide Web is the information system of interlinked web pages and other resources
  - It is accessed from a web browser such as Chrome, Safari, Edge or Firefox
  - It makes use of HTTP to send and receive data

Wide Area Networks

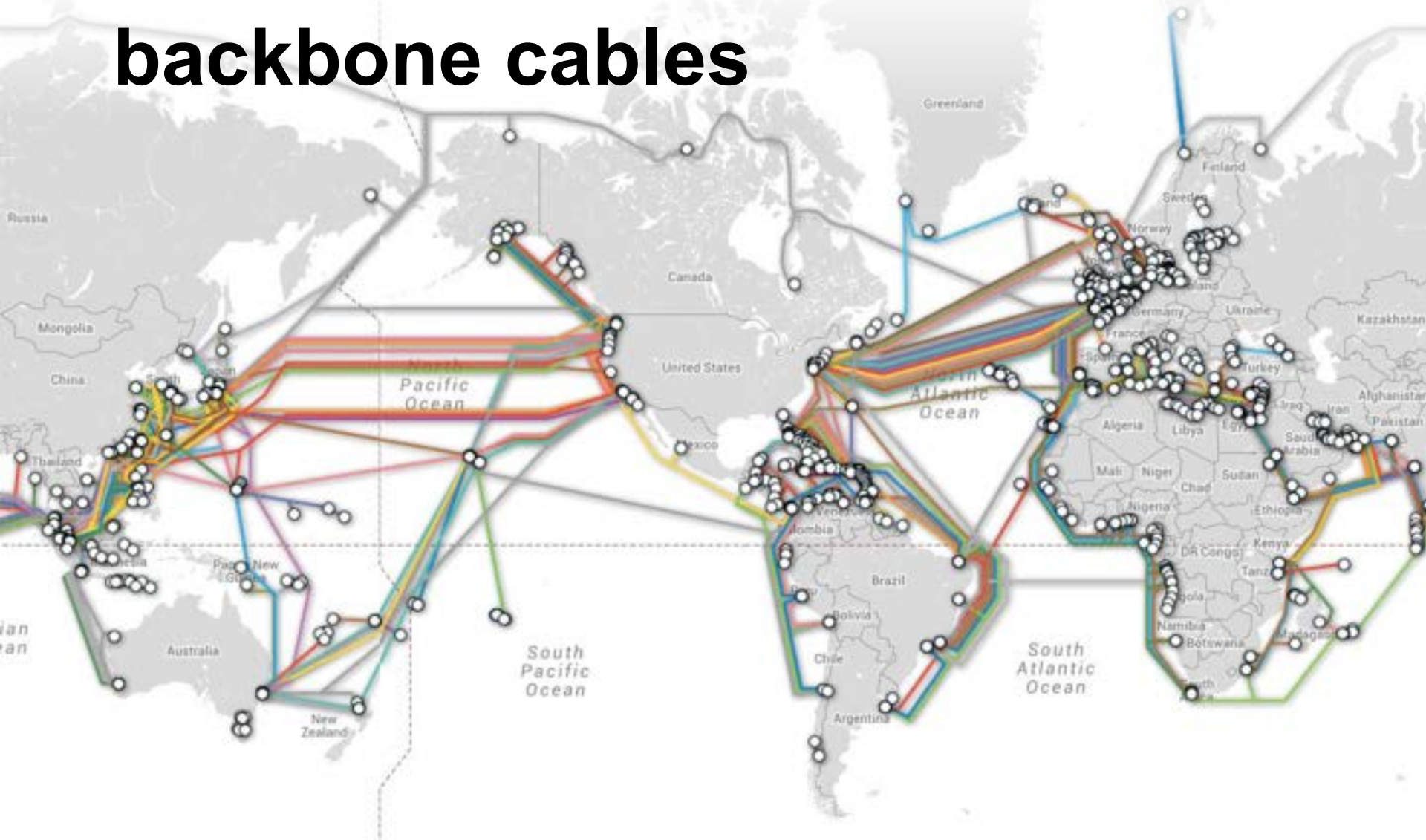
- **Any** network in which the computers communicate using resources supplied by a ‘third party carrier’ such as BT, is a WAN
- A WAN uses cables, telephone lines, satellites or radio waves to connect two or more remote locations
- The Internet is the largest Wide Area Network (WAN) in the world
  - Name **three** organisations that may be using a WAN to connect locations

Wide Area Networks

Answers

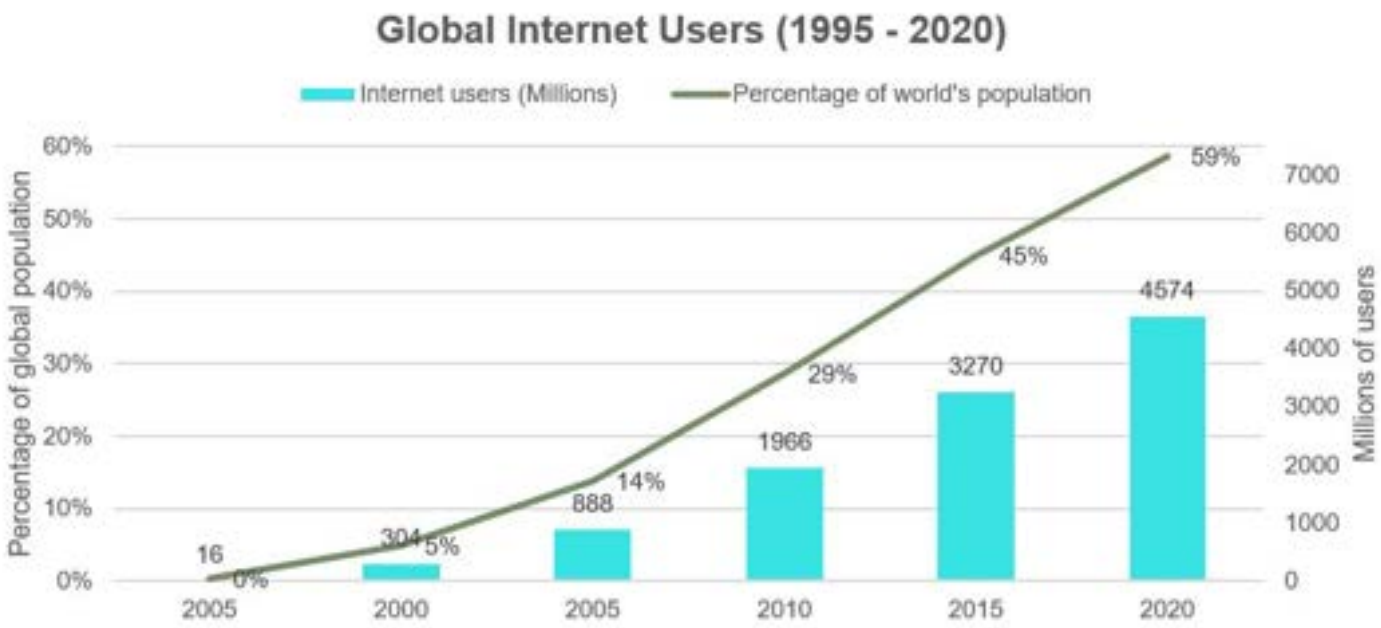
- Organisations such as universities and schools may have a number of geographical locations
  - Each one will have its own network. When these networks are joined together they will create a WAN
- Other organisations may use WANs including:
  - Large companies with many offices such as banks
  - Governments which have employees in many locations
  - Research organisations which are located in more than one place

Continental Internet backbone cables



Internet growth

- In 1969, there were four Internet connected networks
  - By 1989, there were 100,000 connected networks
  - By 2019, 4.5 billion people were Internet users



The Internet of Things (IoT)

- In 2020 there were over 31 billion IoT items connected to the Internet
- These include:
  - Mobile devices
  - Parking meters
  - Smart meters
  - Cars
  - Kitchen appliances
  - Cat trackers

IPv4 Addressing

- Like every house in the world has an address, every computer connected to the Internet has an IP (Internet Protocol) address
- IPv4 addresses are four 8-bit numbers (0-255) separated by dots when written in denary
  - For example 72.129.214.16
- Newer IPv6 addresses are 128-bit numbers usually shown in hex
  - For example 1023:0ef1:0000:0000:34dd:00fe:0189:2233

Worksheet 1

- Complete **Task 1** on **Worksheet 1**

A circuit switching network

- When a traditional telephone call is made, a dedicated connection is set up between you and the person you are calling for the duration of the call
  - This is called ‘circuit switching’
  - It works fine for phone calls, but there could never be enough lines for all the billions of people sending data across the Internet
- Packet switching solves this problem

Packet switching

- Suppose you want to send a file of 3 MB to someone via the Internet
- The file is broken up into data ‘packets’ of 512 bytes
- Each packet is given a header containing
  - The IP address it is going to
  - The IP address it has come from
  - The sequence number of the packet
  - The number of packets in the whole communication
  - Error checking data

Data packets

- Each data packet will contain the same number of bytes
  - The packets may flow through the Internet via different routes
  - Each packet is marked with the packet number so that the packets can be put back together by the receiver

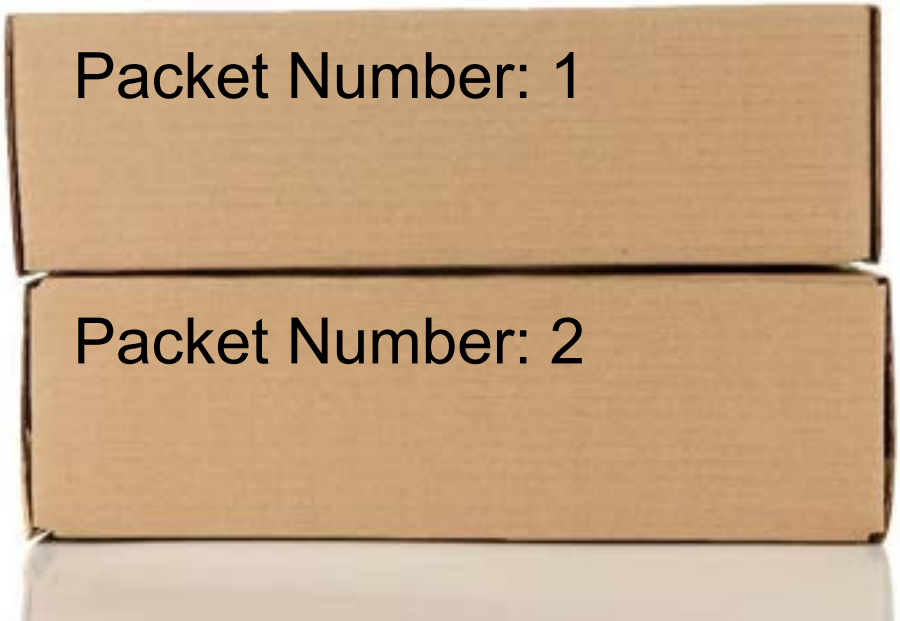
Internet packet routing

Worksheet 1

- Complete **Task 2** on **Worksheet 1**

Reassembling the data

- The recipient computer re-orders the packets into the correct order
- Each packet is checked for errors
  - Corrupt packets are requested to be resent



Moving packets on the net

- Points on a network diagram are called **nodes**
- Routers are any node that can route packets from one place to another
  - Your home router, routes packets to the different devices on your home network
  - Internet routers route packets around the world like a mail sorting office

PG ONLINE

Domain Name System (DNS)

- The **Domain Name System** is made up of domain names
  - A domain name is a name used to refer to an IP address
  - Google.co.uk is an example of a domain name
- A **DNS server** translates a domain name into an IP address
  - For example, google.co.uk translates to 142.250.179.227

DNS servers

- A computer will first check to see if it has cached the IP address for a domain name
  - If it hasn't then it will request the IP address from a DNS server
  - If the DNS server hasn't cached the domain name and IP address, then it will make a request to a more authoritative DNS server

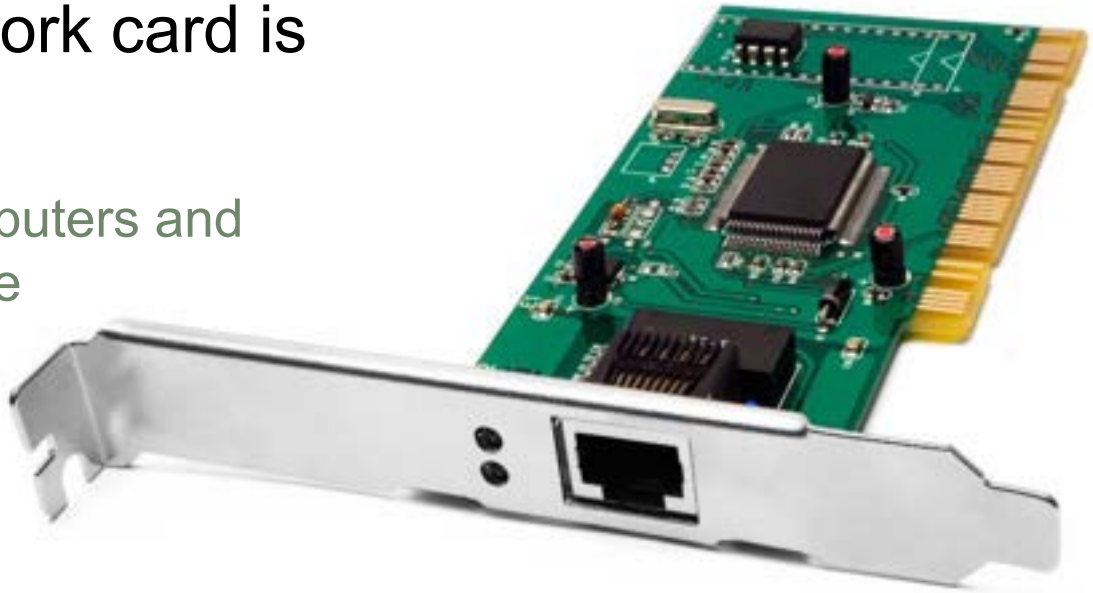
Your IP address can change

- If you use a laptop at home, its IP address is the one belonging to the router at home
- But if you take the laptop on holiday your laptop will be assigned a new IP address
  - Many home IP addresses will change when the router reconnects to the Internet. This is known as a dynamic IP address

PG ONLINE

Network Interface Cards

- A Network Interface Card (NIC) is built into every networked device
  - It may be wireless with an antenna or wired with a network cable socket
- A removable network card is shown here
  - Most modern computers and phones incorporate the NIC into the motherboard



MAC addressing

- A **Media Access Control** (**MAC**) address is assigned to each Network Interface Card (NIC) by the manufacturer
  - Your computer may have more than one MAC address if it has both an Ethernet and Wi-Fi connection
- A smart phone has two different MAC addresses
  - One will be for Wi-Fi and one for Bluetooth
- A MAC address is stored in binary, but is presented to users like this:

3B:14:E6:39:0A:2C

MAC address

- Every networked device in the world has a unique MAC address
- There are 2^{48} possible addresses – enough for 40,000 NICs per person with a world population of 7.125 billion
  - Why doesn't the Internet use a MAC address instead of an IP address to locate a web address?

The role of the MAC address

- The actual geographical location of a networked device can be determined by its IP address
  - If you take your laptop to another part of town, or to another country, its IP address will change
- When you request a web page, each router along the way uses the MAC address of the next router to send the data packet on the next leg of its journey
  - The MAC address of a NIC is unique and never changes

Worksheet 1

- Complete **Task 3** on **Worksheet 1**

Plenary

- Work with a partner asking alternate questions
- What do each of the following words or acronyms mean?
  - WAN
  - Packet
  - Routing
  - DNS
  - IP address
  - MAC address

Plenary

Answers

- WAN – Wide Area Network (large geographical area)
- Packet – one unit of data sent through an IP network
- Routing – directing IP packets to their destinations
- DNS – Domain Name System
- IP address – Internet Protocol address. Where packets of data are sent to or received from
- MAC address – Media Access Control address. The unique identifier of each device (or network interface card)

Copyright

© 2024 PG Online Limited

The contents of this unit are protected by copyright.

This unit and all the worksheets, PowerPoint presentations, teaching guides and other associated files distributed with it are supplied to you by PG Online Limited under licence and may be used and copied by you only in accordance with the terms of the licence. Except as expressly permitted by the licence, no part of the materials distributed with this unit may be used, reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic or otherwise, without the prior written permission of PG Online Limited.

Licence agreement

This is a legal agreement between you, the end user, and PG Online Limited. This unit and all the worksheets, PowerPoint presentations, teaching guides and other associated files distributed with it is licensed, not sold, to you by PG Online Limited for use under the terms of the licence.

The materials distributed with this unit may be freely copied and used by members of a single institution on a single site only. You are not permitted to share in any way any of the materials or part of the materials with any third party, including users on another site or individuals who are members of a separate institution. You acknowledge that the materials must remain with you, the licencing institution, and no part of the materials may be transferred to another institution. You also agree not to procure, authorise, encourage, facilitate or enable any third party to reproduce these materials in whole or in part without the prior permission of PG Online Limited.

OCR
Oxford Cambridge and RSA
An OCR endorsed teaching and learning tool

GCSE OCR

Computer Science J277

2

Local Area Networks

Unit 3 Networks, connections and protocols

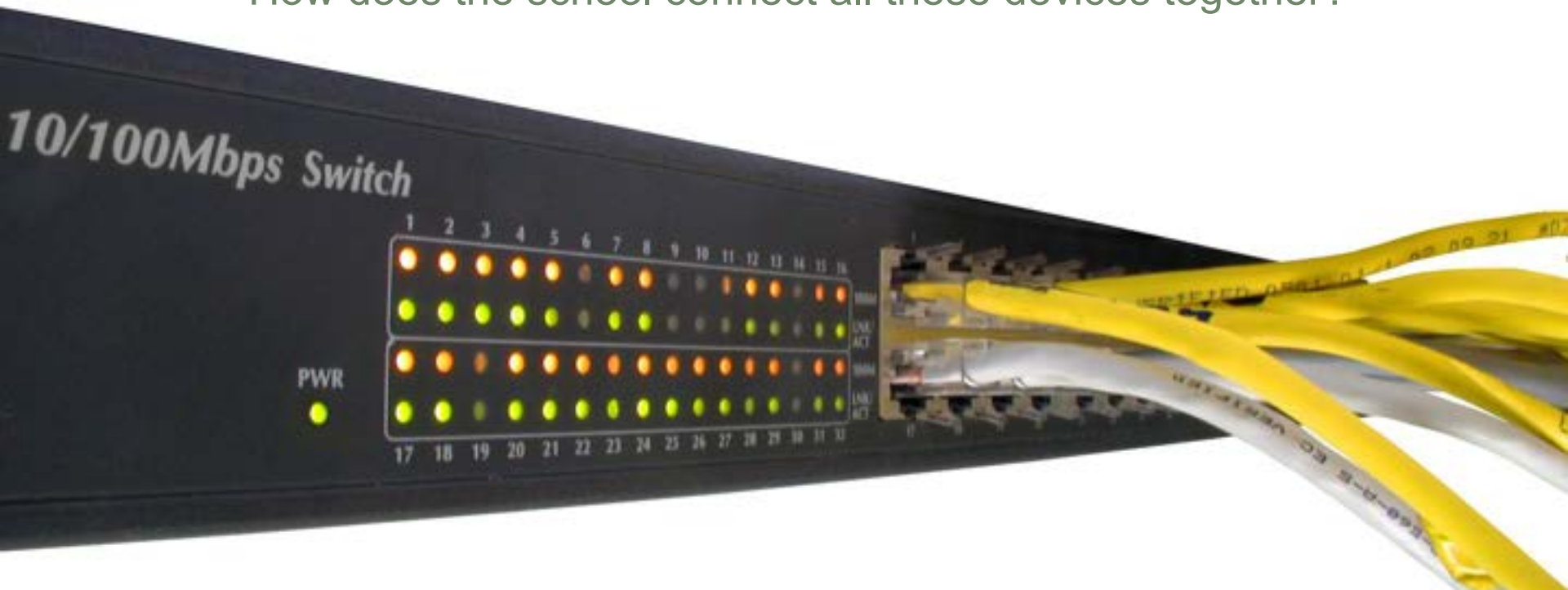
PG ONLINE

Objectives

- Describe the difference between a Local Area Network and a Wide Area Network
- Describe star and mesh network topologies
- Describe routers and switches needed to connect stand-alone computers into a Local Area Network
- Explain the use of Ethernet standards to transmit data over a wired network
- Explain the concept of virtual networks

Starter

- A typical school network contains many computers
- It may include desktop computers, laptops and other devices such as tablets and smartphones
  - How does the school connect all these devices together?



What is a LAN?

- **LAN** stands for **Local Area Network**
- A LAN operates on a single site such as a school, hotel or business using their own cabling systems

PG ONLINE

Networked computers

- What is the difference between a LAN and a WAN?
- What are the advantages and disadvantages of connecting computers in a LAN compared to leaving the computers as standalone machines?

Networked computers

Answers

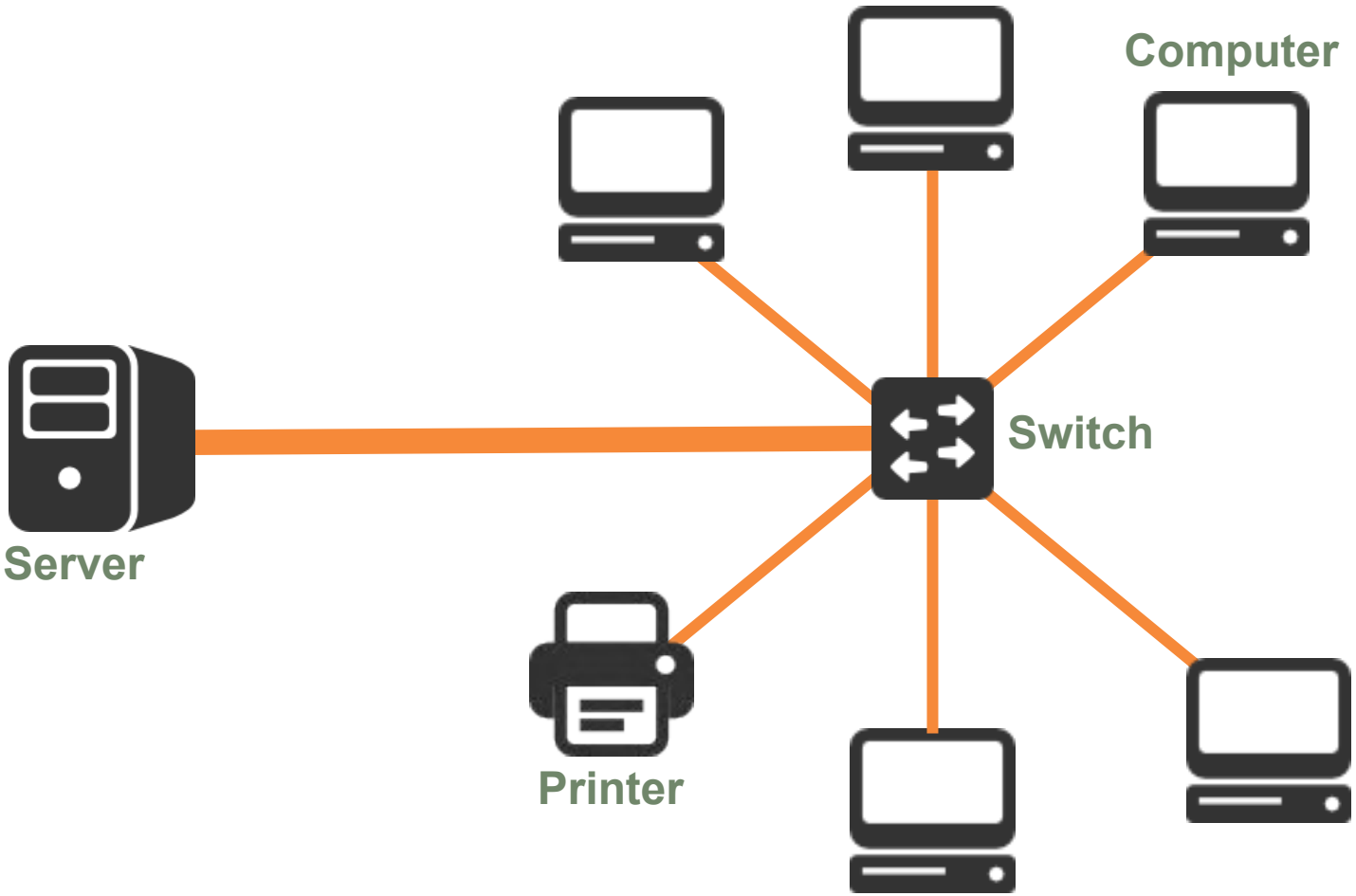
- LANs are local and located within one location such as a building
  - WANs connect networks at remote locations via leased lines

Advantages of networks	Disadvantages of networks
Computers can share resources such as printers	Purchasing the network hardware is expensive
Files can be accessed through any computer in the network	Managing a large network is complicated
Data is easy to back up as it is stored centrally on the server	Viruses may be able to infiltrate the network and infect every computer

Network topologies

- A topology is the way in which parts of a system are connected
  - For instance, the London Underground network has a topology that shows where connections, and lines are
- There are many topologies for setting up networks and LANs
- Two of these topologies are:
  - Star
  - Mesh

Star network

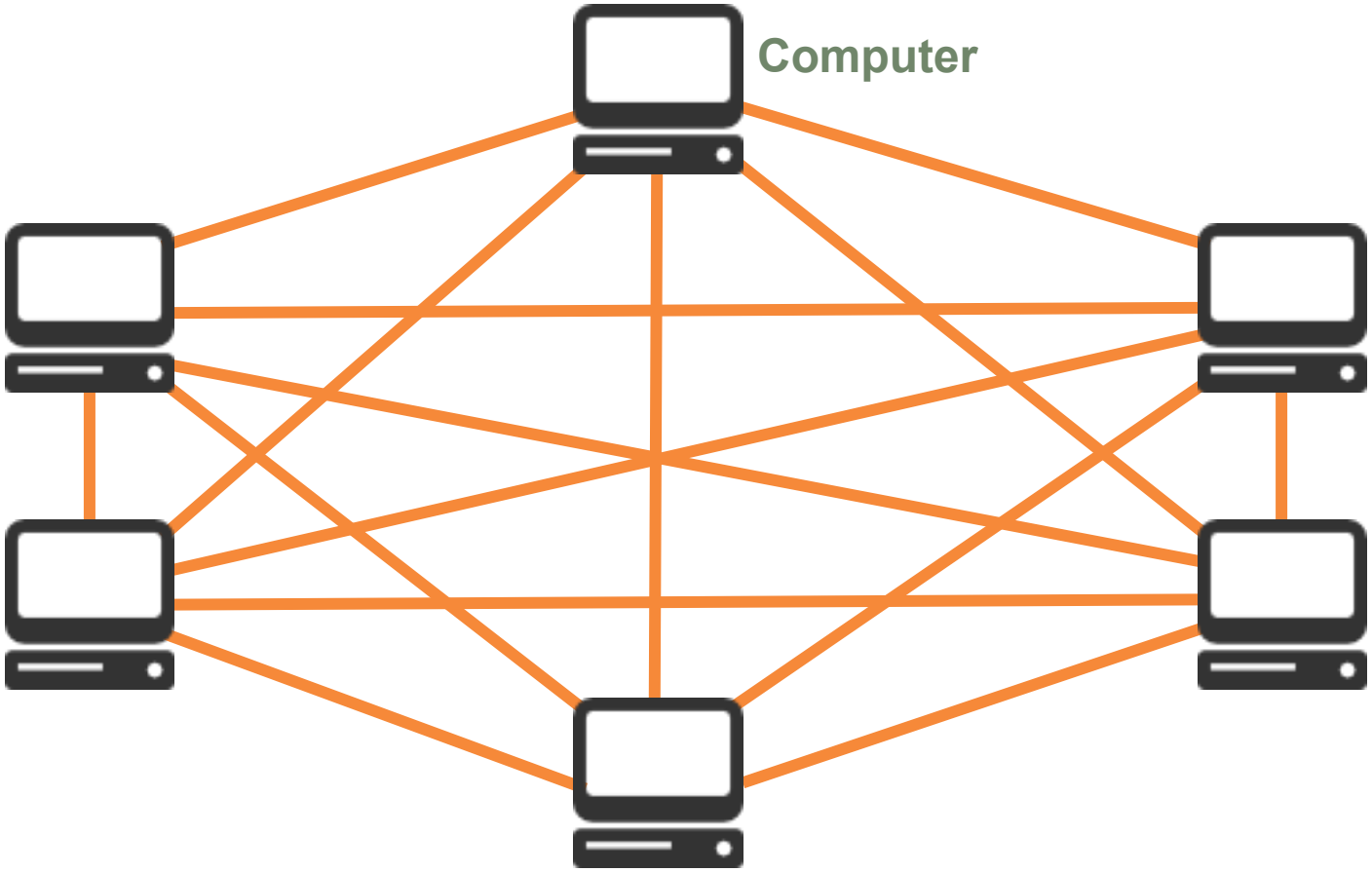


Star network

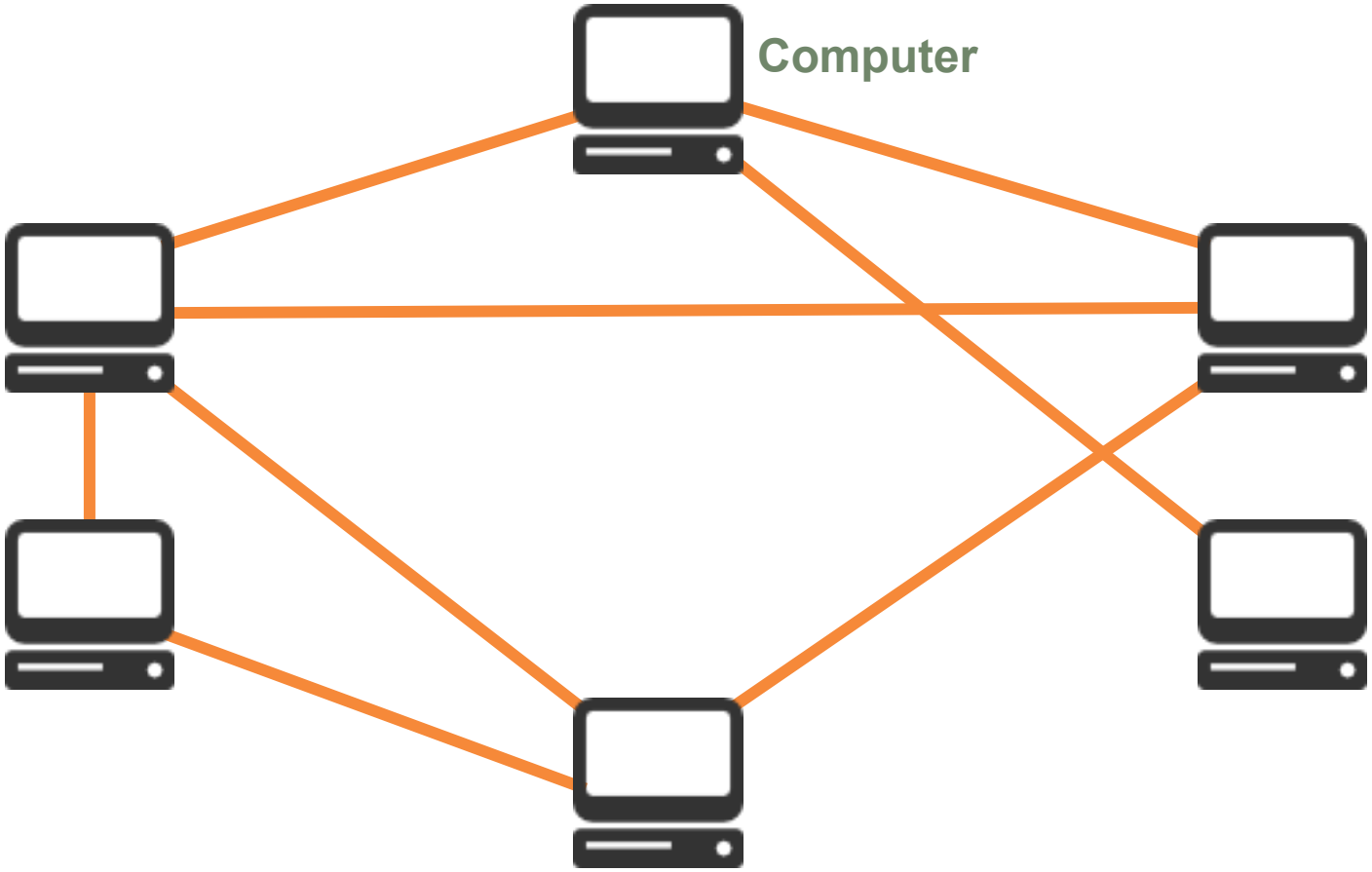
- In a star network, computers and other devices (known as nodes) are all connected to a central switch

Advantages	Disadvantages
Fast data transfer to the hub as each wire isn't shared with other computers	Requires additional hardware such as the central switch and network cables
If one cable fails the other computers are not affected	If the central switch fails the whole network goes down

Full Mesh network



Partial Mesh network



Mesh networking

- Nodes act as routers for data in order to relay and propagate data in the network
- A partial mesh network is usually used, often in conjunction with star topologies to create larger networks
  - What topology does the Internet use?
  - How might mesh principles affect wireless networks?

Internet topology

Answers

- The Internet is highly complex as it combines many networks together, but many parts operate as a **partial mesh**
  - For example, ISPs will connect to many other networks on the Internet. If one connection is lost, packets are directed through other connections
- Many LANs that connect to the Internet use a **star topology**, although mesh Wi-Fi is increasing in popularity

Wireless mesh networks

- Wireless mesh networks have enormous potential to provide Wi-Fi solutions for entire organisations, cities or even countries
  - Only one node needs a wired Internet connection – no other cabling or infrastructure is required
  - The more nodes that use the network, the more routes there are for data to travel through
  - A node within broadcast distance of three other nodes, will have triple the standard bandwidth
  - As the distance between one node and another is halved, signal strength becomes four times stronger

Mesh network

Advantages	Disadvantages
No single point of failure – it is resilient	Can involve redundant connections
Expansion and modification can be done without disrupting the network	Expensive to install cabling if using wired connections
Data can be transmitted from different devices simultaneously	Network maintenance and administration is difficult

Worksheet 2

- Complete **Task 1** of **Worksheet 2**

Network hardware

- Additional hardware is required to connect a stand-alone computer to a LAN
  - A **Network Interface Card/Controller** (NIC) in your computer or device
  - A **router** or **switch,** which provides access to a local area network
  - A **modem** is required to connect to the Internet – this is usually combined with the router inside a single device
  - A **Wireless Access Point** connects wireless devices to a network. Many home wireless access points are part of a router

Routers and switches

- **Router**
  - Looks at the destination of packets of data and sends them to the network that is closer towards their destination
  - A home router will route packets between the home LAN and the Internet

- **Switch**
  - Switches connect each node (computer) in a network
  - They know the MAC address of all connected computers and devices
  - When a packet of data arrives, they can send it to the correct computer

Ethernet protocols

- ‘Ethernet’ refers to a *family of standard local networking ‘protocols’* or rules
- It describes how devices should format data ready for transmission between computers on the same network
  - Similar to polite human conversation, nodes will wait until the connection is quiet before attempting to ‘speak’ or transmit
  - Two nodes attempting to transmit simultaneously will stop and each wait a random period before reattempting

Transmission media

- Transmission media is the way that communication is sent through the network
  - Ethernet is sent through twisted copper cables (typically with 100 meters range) or fibre optic cables
  - Internet connections to UK houses are typically sent through copper cables or coaxial cables. Fibre connections often only go as far as the local green cabinet

Worksheet 2

- Now complete **Task 2** of the worksheet

Plenary

- Define the following words:
  - LAN
  - Star network
  - Mesh network
  - Router
  - Switch
  - NIC
- Name **four** types of transmission media

Plenary

Answers

- LAN – local area network; connects nearby computers such as in a school, business, home
- Star network – each computer connects to central switch
- Mesh network – each computer/router can connect to other computers/routers
- Router – routes packets of data towards their destination
- Switch – sends data to the appropriate connected computer
- NIC – Network Interface Card/Controller

- Name **four** types of transmission media
  - Ethernet cable, fibre optic cable, Wi-Fi/wireless/microwaves, coaxial cable, copper cable

Copyright

© 2024 PG Online Limited

The contents of this unit are protected by copyright.

This unit and all the worksheets, PowerPoint presentations, teaching guides and other associated files distributed with it are supplied to you by PG Online Limited under licence and may be used and copied by you only in accordance with the terms of the licence. Except as expressly permitted by the licence, no part of the materials distributed with this unit may be used, reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic or otherwise, without the prior written permission of PG Online Limited.

Licence agreement

This is a legal agreement between you, the end user, and PG Online Limited. This unit and all the worksheets, PowerPoint presentations, teaching guides and other associated files distributed with it is licensed, not sold, to you by PG Online Limited for use under the terms of the licence.

The materials distributed with this unit may be freely copied and used by members of a single institution on a single site only. You are not permitted to share in any way any of the materials or part of the materials with any third party, including users on another site or individuals who are members of a separate institution. You acknowledge that the materials must remain with you, the licencing institution, and no part of the materials may be transferred to another institution. You also agree not to procure, authorise, encourage, facilitate or enable any third party to reproduce these materials in whole or in part without the prior permission of PG Online Limited.

OCR
Oxford Cambridge and RSA
An OCR endorsed teaching and learning tool

GCSE OCR

Computer Science J277

3

Wireless networking

Unit 3 Networks, connections and protocols

PG ONLINE

Objectives

- Understand wireless modes of connection, including:
  - Wi-Fi
  - Bluetooth
- Explain the need for Wireless Access Points to create wireless hotspots
- Understand how encryption is used to secure data across network connections

Starter

- How are wireless technologies used by people and electronic devices today?
  - Name at least **six** different products they are used in

Starter

Answers

- Wi-Fi
  - Laptops, some desktops, tablets, smartphones, many IoT devices such as smart speakers and smart lights
- Bluetooth
  - Wireless headphones, connecting phones to car entertainment systems, keyboards

Bluetooth

- **Bluetooth** is a wireless method of connecting devices together over a short distance
  - It uses radio waves at 2.4 GHz
  - Devices are typically within 10 metres of each other
- Devices need to be paired to connect them together
  - What does this mean?

PG ONLINE

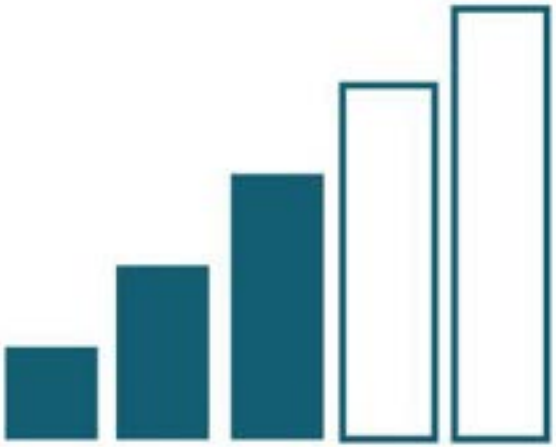
Pairing

Answers

- Pairing Bluetooth devices means the following:
  - A persistent link is made between the two devices
  - A device can ask for a passkey in order to pair to it – this gives a layer of security like a password
  - Passkeys only need to be entered once and will be remembered by devices that are paired
- When a paired device is close enough to communicate it will say that it is connected
  - This means that data can be communicated between the devices

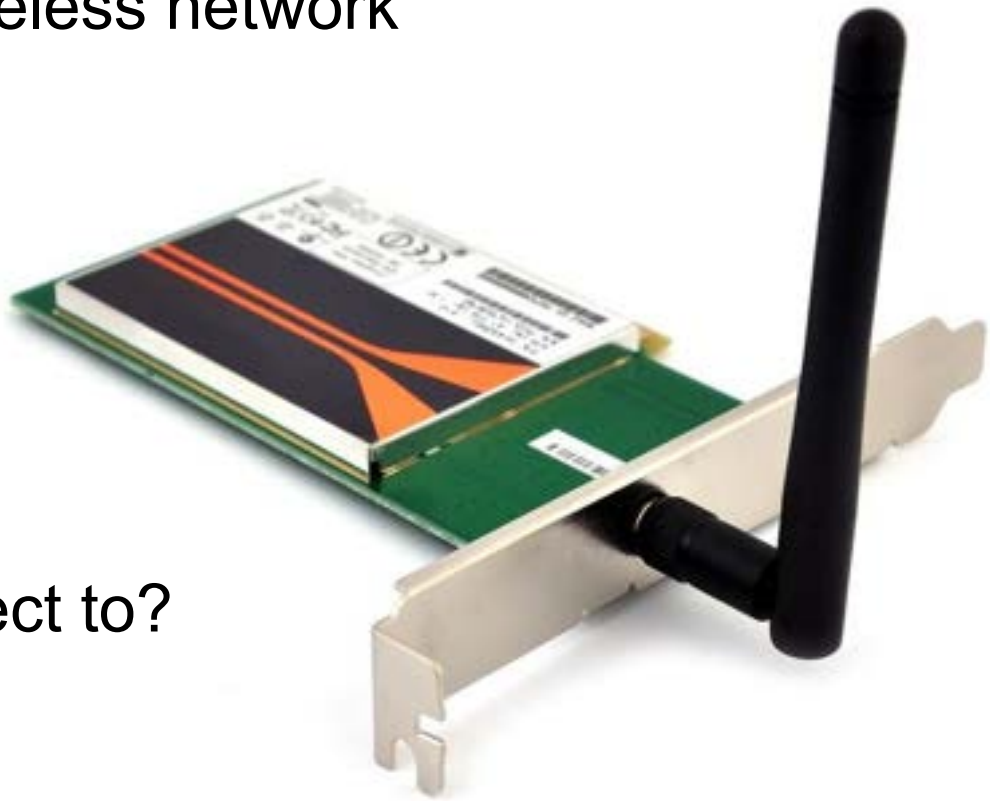
Wireless transmission

- Commonly uses radio waves for communication
- Susceptible to interference from objects and other nearby electronic or radio devices
  - How does your wireless signal strength vary throughout your own home or school?



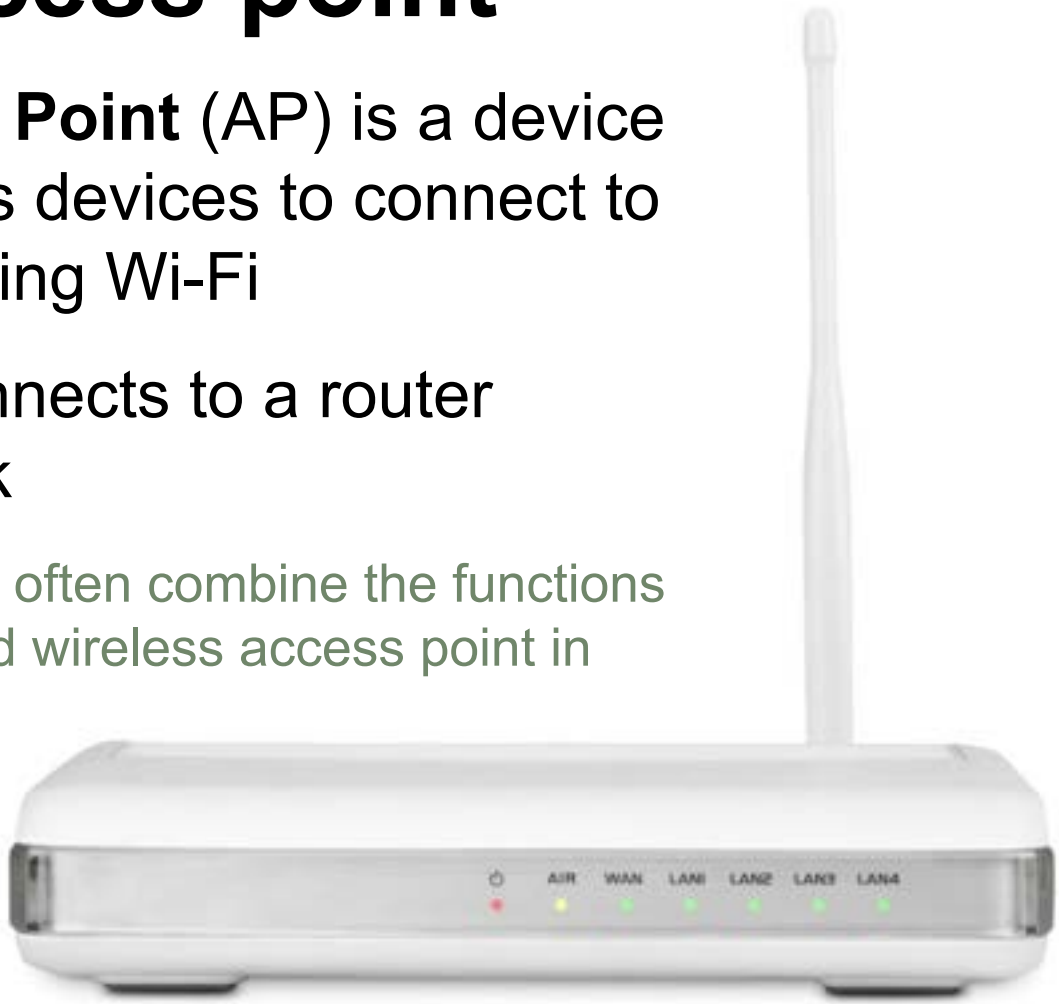
Wireless NICs

- Built into every networked device capable of connecting to a wireless network
- These include:
  - PCs
  - Smart phones
  - Wireless speakers
- What does a wireless NIC connect to?



Wireless access point

- A wireless **Access Point** (AP) is a device that allows wireless devices to connect to a wired network using Wi-Fi
- The AP usually connects to a router via a wired network
  - Home Wi-Fi routers often combine the functions of switch, router and wireless access point in one box



802.11 standards

- 802.11b/g/n uses the 2.4GHz frequency
- 802.11a/n/ac uses the 5GHz frequency

	Advantages	Disadvantages
2.4GHz	Greater range and coverage	More interference from other devices as this is a crowded frequency
5GHz	Less crowded space with 23 non-overlapping channels with higher data transmission rates	Less able to penetrate through walls

Intercepting data

- Data that is transmitted over a network can be intercepted
  - Any intercepted data can be read and understood unless measures are taken to prevent it from being interpreted
  - These measures are known as encryption

PG ONLINE

Worksheet 3

- Complete **Task 1** on **Worksheet 3**

Encryption

- Encryption is the encoding of data so that it can no longer be easily understood
- A simple shift cipher might encode "*Box345*" as follows:

Encryption algorithm

"Box345" → → **"Dqz567"**

Encryption terminology

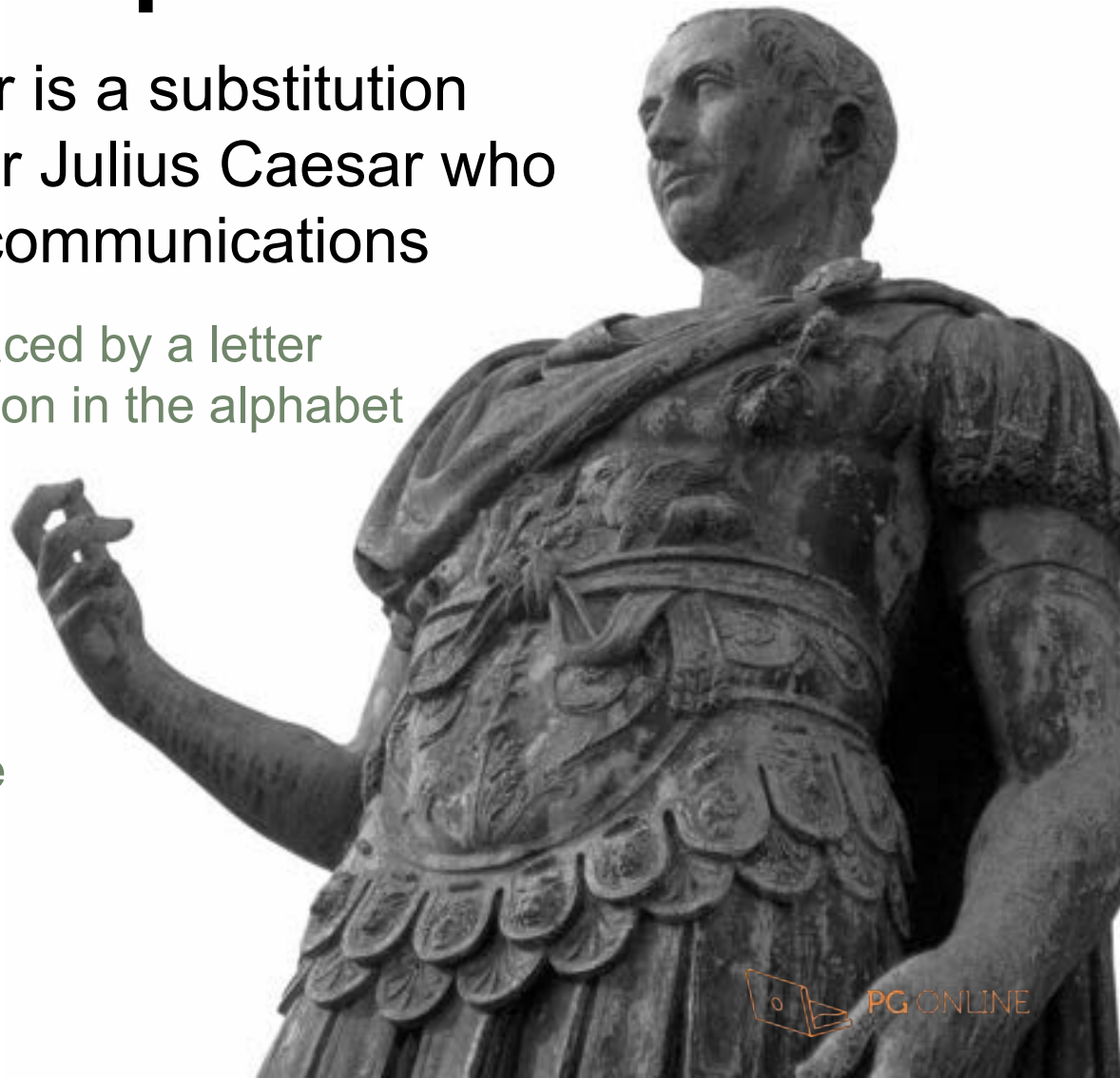
- Plaintext: the original message to be encrypted
- Ciphertext: the encrypted message
- Encryption: the process of converting plaintext into ciphertext
- Key: a sequence of numbers used to encrypt or decrypt, often data using a mathematical formula
- Encryption algorithm: the formula for encrypting the plaintext
  - Two inputs: **plaintext** and a **secret key**

Encryption techniques

- Private key (Symmetric encryption)
  - A single key is used to encrypt and decrypt a message and must be given to the recipient of your message to decrypt the data
- Public key (Asymmetric encryption)
  - Two keys are used - one (public key) to encrypt and the other (private key) to decrypt data
  - This is more secure as it means that you never have to send or reveal your decryption key

Caesar shift cipher

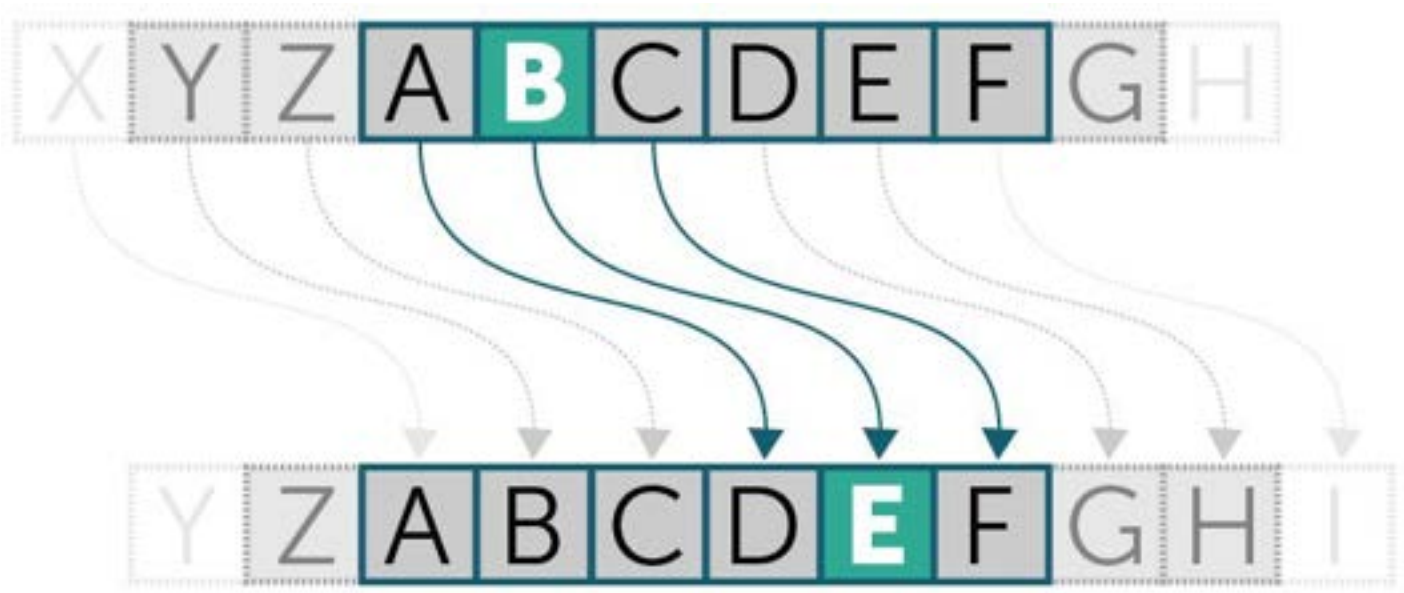
- The Caesar cipher is a substitution cipher named after Julius Caesar who used it in private communications
  - Each letter is replaced by a letter *n* positions further on in the alphabet
  - *n* is the key and is used to encrypt and decrypt the message
  - This is an example of **symmetric encryption**



PG ONLINE

Caesar cipher

- The Caesar cipher is most basic type of encryption and the most insecure
- Letters of the alphabet are shifted by a given number



Deciphering the code

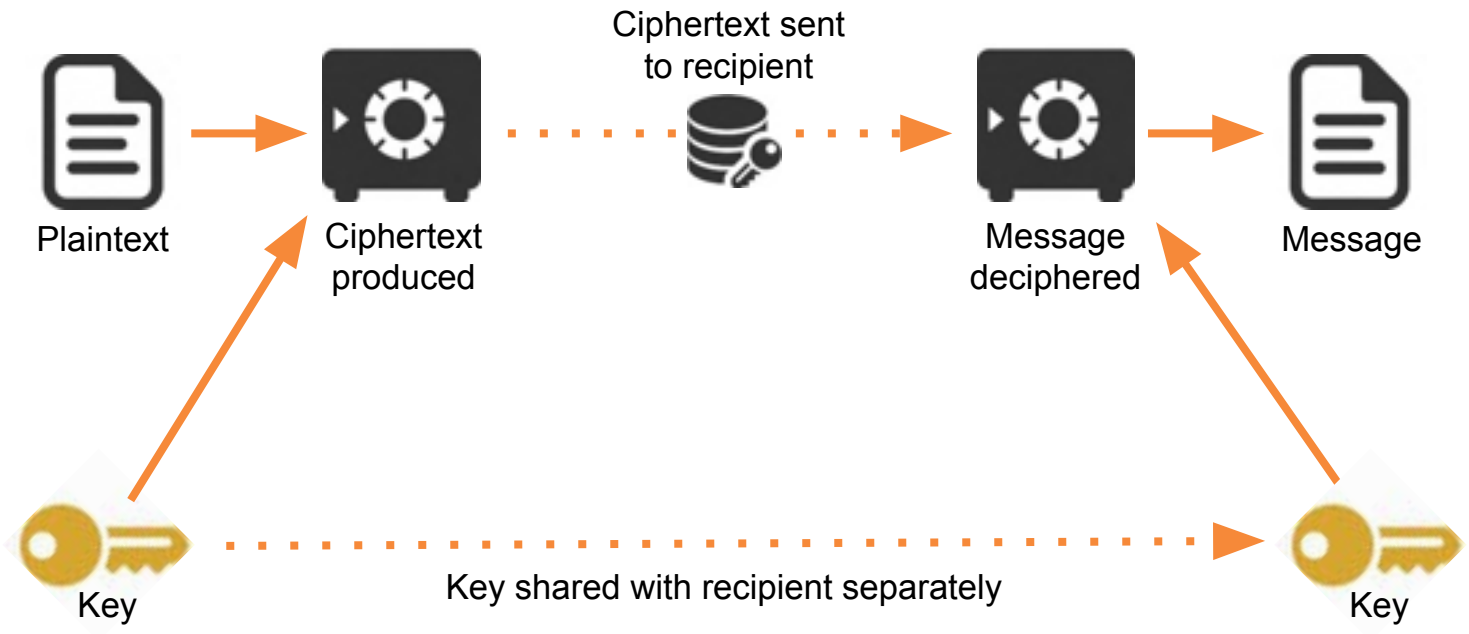
- **Key = Shift □ 3**
- Decode "**DWWDFN DW GDZQ**"

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
↓														
D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R

- Using a different key, crack the code for '**PCRPCYR**'

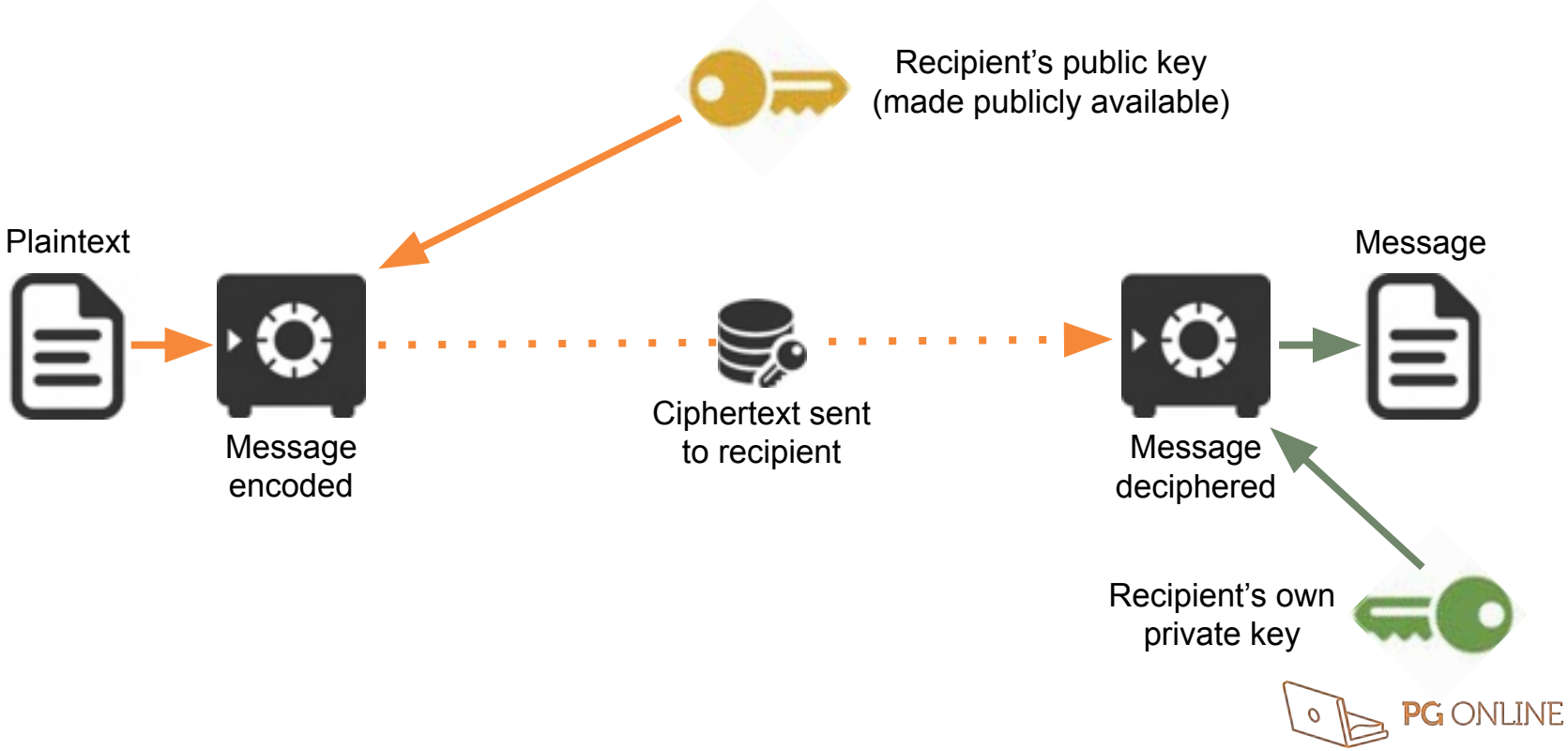
Symmetric encryption

- Same key used to encrypt and decrypt a message



Public key (Asymmetric) encryption

- Two keys are used: a public key known to everyone for encrypting and a private key for decrypting



Encryption in networks

- Wi-Fi is particularly vulnerable to eavesdroppers as the signal can be listened to from a distance
- Wired connections such as Ethernet and fibre optic are also vulnerable
  - What methods of encryption are used when using wireless networks?

Wireless encryption standards

- The most common wireless security standards are:
  - WEP (Wired Equivalent Privacy) and
  - WPA (Wi-Fi Protected Access)
- The WEP method of encryption is older and can be cracked in seconds
  - As such this standard should no longer be used to protect your home router
  - Instead, you should use WPA or preferably the newer WPA2 encryption methods
- HTTPS should be used for websites as it encrypts data sent to and received from the site

Strong and weak encryption

- These terms are relative, but:
  - Encryption can be considered to be ‘strong’ when the useful lifetime of the encrypted data is less than the time taken to break the code
  - With weak encryption, the code may be broken in time to use the information, but it wouldn’t be worth the effort trying

Worksheet 3

- Complete **Task 2** on **Worksheet 3**

Plenary

- Discuss in a pair or threes how you use, or could use wireless networks and network security in your home, school and life. Make sure to use the following words at least once:
  - Wireless
  - Wi-Fi
  - Bluetooth
  - Wireless Access Point
  - Encryption
  - WEP
  - WPA / WPA2

Plenary

Answers

- Possible answer:
  - We use a wireless router at home. This contains a switch, router and modem all in one box. All our mobile devices connect to the wireless access point with Wi-Fi. It is encrypted with WPA2. We don’t use WEP as it is insecure. Where possible we try to connect to websites using HTTPS as this makes all data transmitted secure. Our TV and soundbar are connected to the box using Ethernet cables. Bluetooth is used to connect our headphone to smartphones.

Copyright

© 2024 PG Online Limited

The contents of this unit are protected by copyright.

This unit and all the worksheets, PowerPoint presentations, teaching guides and other associated files distributed with it are supplied to you by PG Online Limited under licence and may be used and copied by you only in accordance with the terms of the licence. Except as expressly permitted by the licence, no part of the materials distributed with this unit may be used, reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic or otherwise, without the prior written permission of PG Online Limited.

Licence agreement

This is a legal agreement between you, the end user, and PG Online Limited. This unit and all the worksheets, PowerPoint presentations, teaching guides and other associated files distributed with it is licensed, not sold, to you by PG Online Limited for use under the terms of the licence.

The materials distributed with this unit may be freely copied and used by members of a single institution on a single site only. You are not permitted to share in any way any of the materials or part of the materials with any third party, including users on another site or individuals who are members of a separate institution. You acknowledge that the materials must remain with you, the licencing institution, and no part of the materials may be transferred to another institution. You also agree not to procure, authorise, encourage, facilitate or enable any third party to reproduce these materials in whole or in part without the prior permission of PG Online Limited.

OCR
Oxford Cambridge and RSA
An OCR endorsed teaching and learning tool

GCSE OCR

Computer Science J277

4

Client-server and peer-to-peer networks

Unit 3 Networks, connections and protocols

PG ONLINE

Objectives

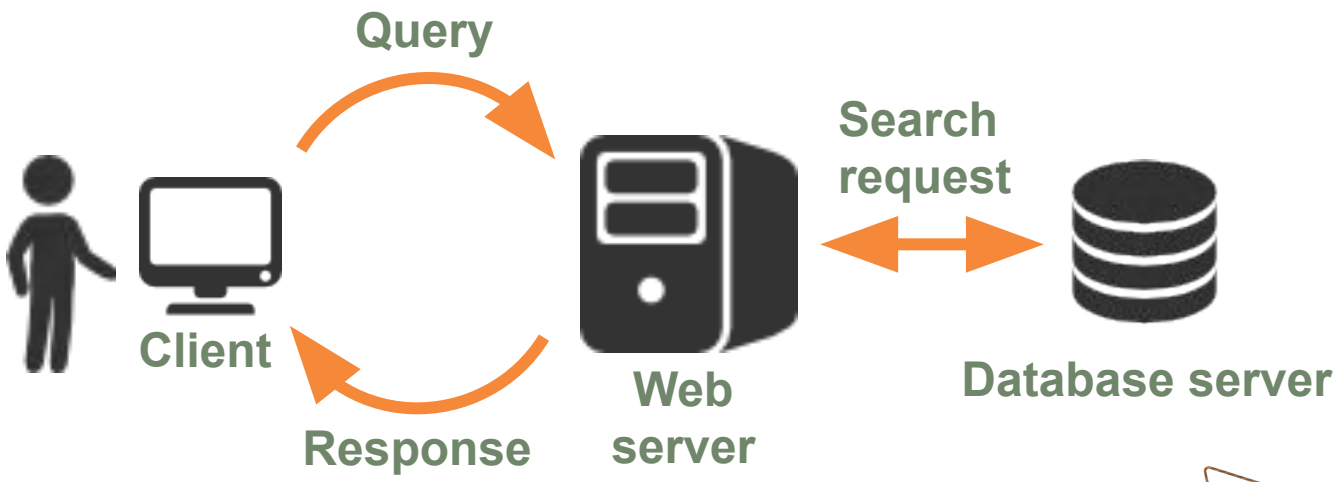
- Explain the role of computers in client-server and peer-to-peer networks
- Explain the advantages and disadvantages of client-server and peer-to-peer networks
- Describe what is meant by:
  - Hosting
  - The Cloud
- Explain the advantages and disadvantages of various transmission media
- Describe the factors that affect network performance

Starter

- A web address is entered to request a web page
  - What happens after enter is pressed?

The client-server model

- In the client-server model, there is a client computer and a server
  - The **server** is a powerful computer which provides services or resources required by any of the clients
  - A **client** is a computer or device which requests the services or resources provided by the server



Servers on LANs and WANs

- One or more servers may be part of a local area network
  - A server may also be part of a wide area network
- Companies such as Google and Amazon have huge data centres servers in several different countries
- Each data centre will contain thousands of servers

A Google server room in Council Bluffs, Iowa

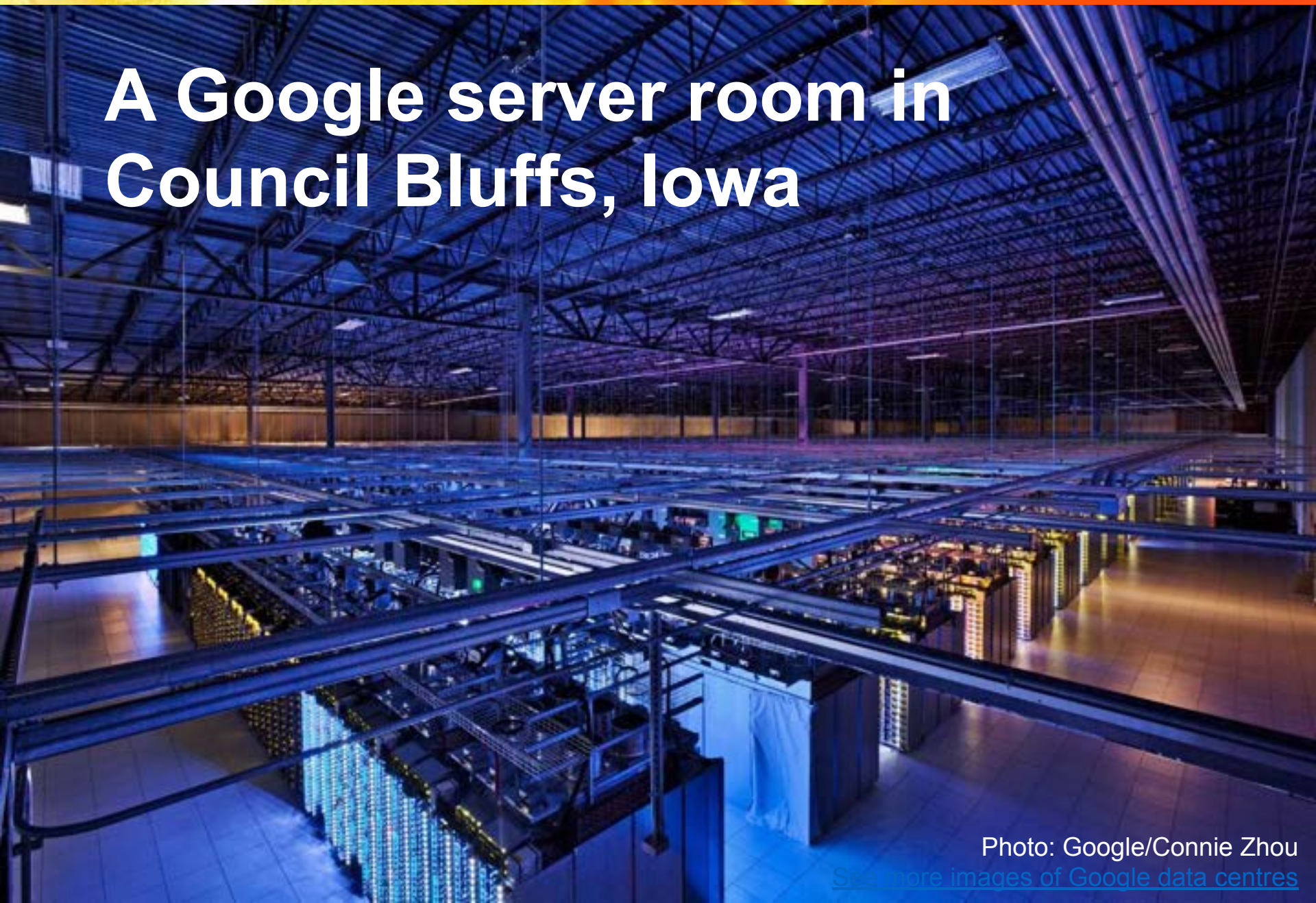
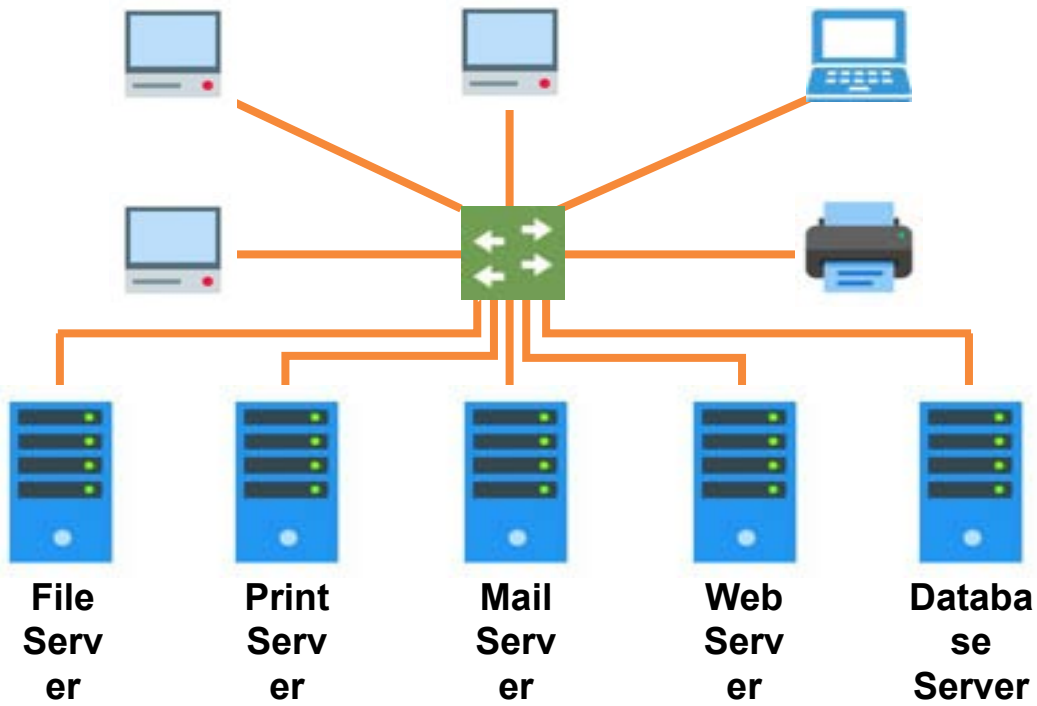


Photo: Google/Connie Zhou

See more images of Google data centres

A school client-server setup

- Five different types of server in a typical school are given below
  - What do each of them do?



PG ONLINE

Serving all your needs

Answers

- In a school network, there may be dozens or even hundreds of clients and several servers
  - A file server holds all the data files and databases and manages backups
  - A print server may organise printing on different printers
  - An email server may receive emails, detect and block spam, distribute emails to users
  - A web server may host the school's external website
  - A database server may hold student records

Role of the client

- The client sends requests to the server
- Waits for a reply
- Receives the reply

PG ONLINE

Role of the server

- The server waits for requests from a client
- Performs any processing required to fulfil the request
- The requested data is sent back to the client
  - What are the advantages of using a client-server model?

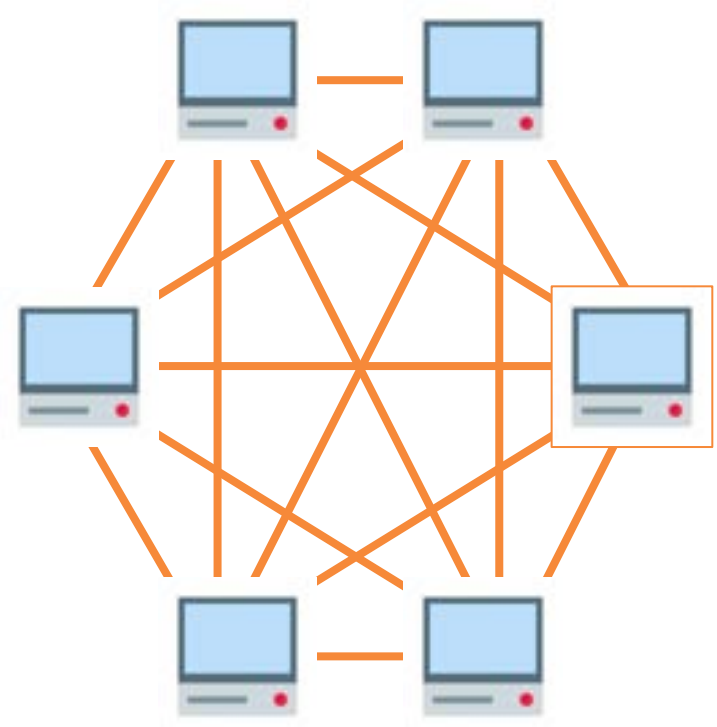
Client-server advantages

Answers

- Some specific examples of advantages include:
  - With file servers, just one file server needs to be backed up and files can be accessed from any computer
  - Email servers provide a central place for email that is then accessed from different devices
  - One print server can manage all files to be printed. This allows one printer to be shared by many computers

Peer-to-peer (P2P) structure

- A peer-to-peer network has no central server



Peer-to-peer networking

- Features of a peer-to-peer network include:
  - Suitable for a small companies with fewer computers
  - No central server controls files or security
  - All computers can easily see files on all other computers
  - All computers can communicate with each other without going through a server

Advantages of a P2P network

- There are a number of advantages of P2P networks
  - Easier to set up – computers can simply be cabled together
  - No need for dedicated server equipment
  - Individual computers can share a printer, router, modem and other hardware
  - Users can communicate directly with each other and share each others' files
- Peer to peer networks are often used in home setups or a small office with few computers
  - What are the disadvantages?

Disadvantages of P2P networks

- There are a number of disadvantages of peer to peer networks including:
  - Viruses and malware are more easily transferred over this type of network as there is no central firewall
  - Data recovery and backup is not done centrally, so each computer has to have its own backup system
  - Files are not centrally organised, but stored on individual computers and may be hard to locate if the computer's owner does not have a logical filing and naming system
  - If a computer is switched off, data on that machine cannot be retrieved from other machines

Client server vs Peer-to-Peer

Client-Server	Peer-to-Peer
A central backup server backs up all file servers	Each computer needs to be backed up
Security is controlled by a central server	Security is controlled by each individual machine
All users are reliant on the central server	There is no central server. If one computer goes down then only its services are lost
Using servers enables businesses to grow to hundreds or thousands of computers and users	Peer-to-peer networks are difficult to maintain beyond a few computers

Peer-to-peer on a WAN

- The P2P configuration can also be used for file-sharing
- Peer-to-peer networks are often used for the illegal distribution of copyright material such as music and films, as they are harder to close down
  - Some peer-to-peer networks have been more successful such as BitTorrent
  - Peer-to-peer file sharing allows each user to share parts of a file, rather than all downloads coming from one central server
  - This significantly reduces the amount of bandwidth required when trying to share a file

Worksheet 4

- Complete **Task 1** on **Worksheet 4**

Hosting

- Web hosting is a service offered by companies that will host web pages and files for websites
  - The web hosts will often provide database servers
  - Companies that provide more general hosting, such as backup servers, are known as hosting service providers
- Advantages of using a web host over sharing files from your own computer include:
  - Web hosts have far more bandwidth – so they can serve more users
  - Web hosts are able to monitor their equipment 24 hours a day
  - Web hosts will backup web sites remotely

The Cloud

- The ‘Cloud’ is a term used to refer to services that are delivered over the Internet
  - It enables users connected to any Internet-enabled computer to access these services
  - What types of services are available?

Cloud Computing applications

- Some examples include Google Drive, OneDrive and Dropbox
- The applications are potentially limitless, from word processing or database software, business applications and online exams, to apps that you run on your smartphone

Google Drive OneDrive Dropbox

Software as a Service (SaaS)

- Cloud-based servers enable access to software on demand
  - Typically the software is used on a subscription basis
  - The software doesn't need to be installed on a local machine
  - The software can be accessed from any Internet-connected computer, including smartphones
- Examples include:
  - Online accounting packages
  - Customer Relationship Management systems (CRMs)
  - Share trading applications

Online applications

PG ONLINE

Cloud computing advantages

- There are a number of advantages of cloud computer including:
  - You can access your data and applications from anywhere at any time
  - You don’t need a powerful computer with a huge hard drive – everything is stored and run on a remote computer
  - Backing up data is no longer crucial – it is done by the service provider

Disadvantages

- There are a number of disadvantages of using cloud computing
  - Personal data will be stored on another company's servers
  - If the Internet connection is lost, often the service becomes unusable
  - Slow Internet connections may result in a poor quality of service
  - Although most cloud services will backup data stored with them, it is not easy for people and organisations to see how resilient this is

Worksheet 4

- Complete **Task 2** on the worksheet

Transmission media

- Computers can be connected using various methods:
  - Copper cable
  - Fibre optics
  - Wi-Fi
- How is data transmitted using each method?

Comparison of media types

	Advantages	Disadvantages
Copper cable (uses electricity)	• tried and trusted technology • relatively inexpensive	• signal affected by electric and magnetic fields • low bandwidth • heavy cables
Fibre optics (uses light)	• very fast data transmission • low loss of signal over distance • not affected by magnetic or electric fields • require very little power • more difficult to 'tap into' than copper cables	• high investment cost • need for expensive optical transmitters and receivers
Wireless transmission (uses radio waves)	• no need for trailing wires (safer) • allows devices to be used anywhere provided there is a signal • easier to add devices to a network	• data transmission rate less than wired systems • signal can be blocked by objects or walls • increased security issues

PG ONLINE

Network performance

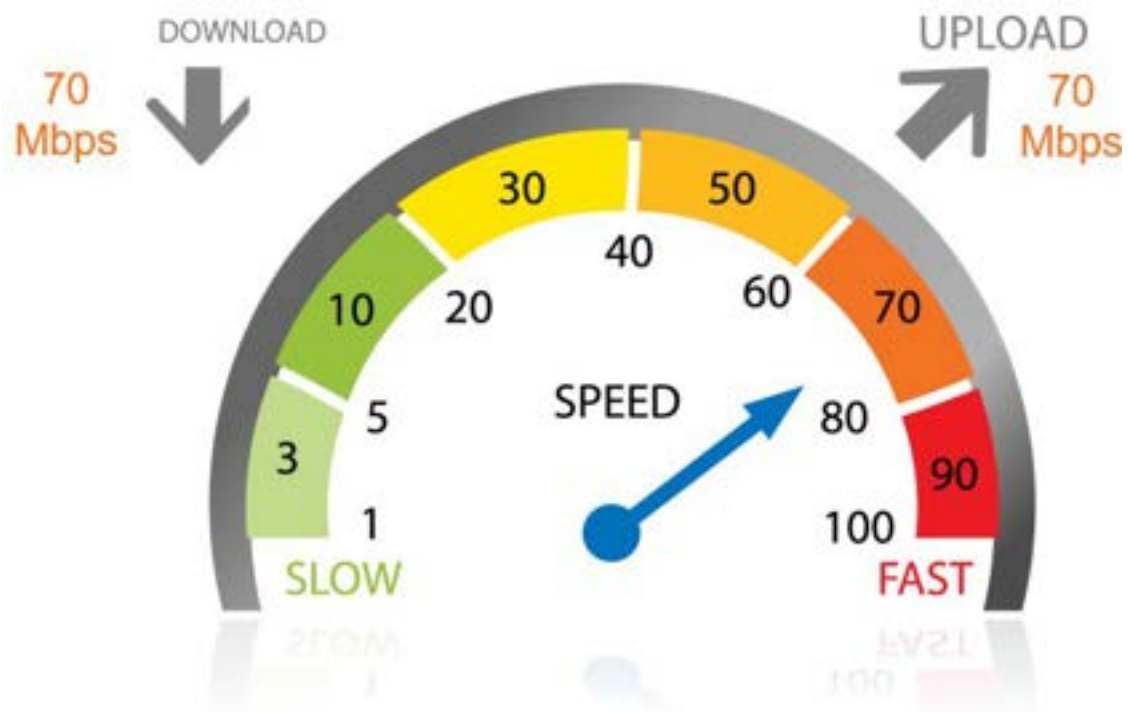
- There are several factors that affect the performance of a network including:
  - Bandwidth
  - Number of devices connected
  - Latency
- What do each of these terms mean?

Bandwidth

- Bandwidth is the amount of data that can be carried through a connection at a time
  - A good analogy is the amount of water that a water pipe can carry

Connection Speeds

- Connection speeds are measured in Mbps (Megabits per Second)
- Run a speed test: www.speedtest.net



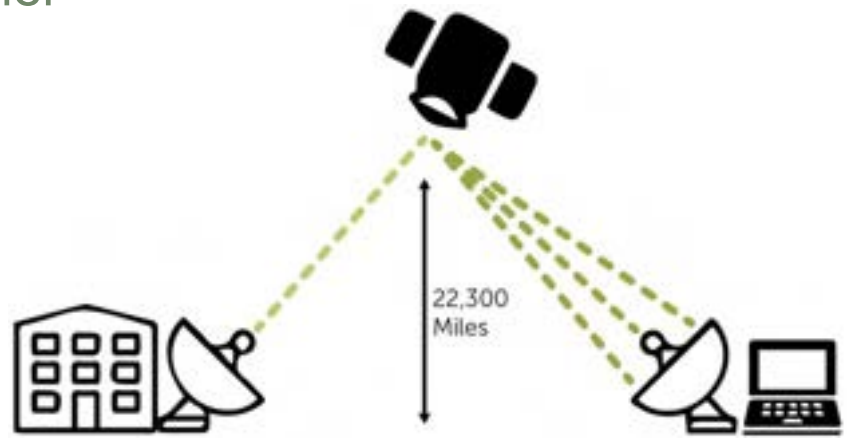
Latency

- When playing multiplayer games online, it is vital that the connection has very little delay between the client and server
  - The delay is known as latency
  - How does this differ to bandwidth?

PG ONLINE

Latency

- Latency is the length of time a packet of information takes to travel through the network
  - Bandwidth is how many packets of information can travel through per second
  - The latency of a phone call via a satellite phone will be longer than that via a landline as it has to travel further



Number of devices connected

- The bandwidth of a network connection may be shared by a number of devices
  - For example, in a home network, there is usually only one connection to the Internet. This is then shared by all the devices on the home network
  - As more devices are connected, the amount of data each one can transfer simultaneously will be reduced
  - A 10 Mbit/s connection may be the bandwidth required for watching streaming television with one TV. However, if 4 TVs were connected and needed to all stream at the same time, a 40 Mbit/s connection would be required

Worksheet 4

- Complete **Task 3** on the worksheet

Plenary

- Fill in the text below with the words at the bottom

A school makes use of a _____ that allows many users to access their _____ from different _____ machines. If all users try to access files at the same time, it may become _____ to receive the files. One reason for this is that there is not enough _____.

bandwidth **file server** **client**

files **slower**

Plenary

Answers

- Fill in the text below with the words at the bottom

A school makes use of a file server that allows many users to access their files from different client machines. If all users try to access files at the same time, it may become slower to receive the files. One reason for this is that there is not enough bandwidth.

bandwidth **file server** **client**

files **slower**

Copyright

© 2024 PG Online Limited

The contents of this unit are protected by copyright.

This unit and all the worksheets, PowerPoint presentations, teaching guides and other associated files distributed with it are supplied to you by PG Online Limited under licence and may be used and copied by you only in accordance with the terms of the licence. Except as expressly permitted by the licence, no part of the materials distributed with this unit may be used, reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic or otherwise, without the prior written permission of PG Online Limited.

Licence agreement

This is a legal agreement between you, the end user, and PG Online Limited. This unit and all the worksheets, PowerPoint presentations, teaching guides and other associated files distributed with it is licensed, not sold, to you by PG Online Limited for use under the terms of the licence.

The materials distributed with this unit may be freely copied and used by members of a single institution on a single site only. You are not permitted to share in any way any of the materials or part of the materials with any third party, including users on another site or individuals who are members of a separate institution. You acknowledge that the materials must remain with you, the licencing institution, and no part of the materials may be transferred to another institution. You also agree not to procure, authorise, encourage, facilitate or enable any third party to reproduce these materials in whole or in part without the prior permission of PG Online Limited.

OCR
Oxford Cambridge and RSA
An OCR endorsed teaching and learning tool

GCSE OCR

Computer Science J277

5

Protocols and layers

Unit 3 Networks, connections and protocols

PG ONLINE

Objectives

- Describe the uses of communications protocols including:
  - HTTP
  - HTTPS
  - FTP
  - POP
  - IMAP
  - SMTP
  - TCP/IP
- Explain the concept of layers in the TCP/IP protocol stack

Starter

- What are the rules around meeting people for the first time
  - What does each person do or say?
  - What can go wrong if people are using different rules?

PG ONLINE

Protocols

- The set of rules is called a **protocol**
- What is the ‘meet and greet’ protocol for meeting someone new?
  - Smile
  - Say hello
  - Does this work in all countries?
    - Eskimos rub noses
    - Japanese bow
    - Tibetans stick their tongues out!
  - Shaking hands with your left hand can be considered rude in many countries

HTTP and HTTPS protocol

- HTTP (hypertext transfer protocol) is used for accessing and receiving web pages via the Internet
  - Web pages are made in HTML (Hypertext mark-up language)
- The protocol requests a web page from the web server
  - The server then sends its response which contains the web page
- HTTPS (secure HTTP) encrypts the information so that it cannot be understood by an eavesdropper
  - Which websites might require the use of HTTPS?

Websites with HTTPS

Answers

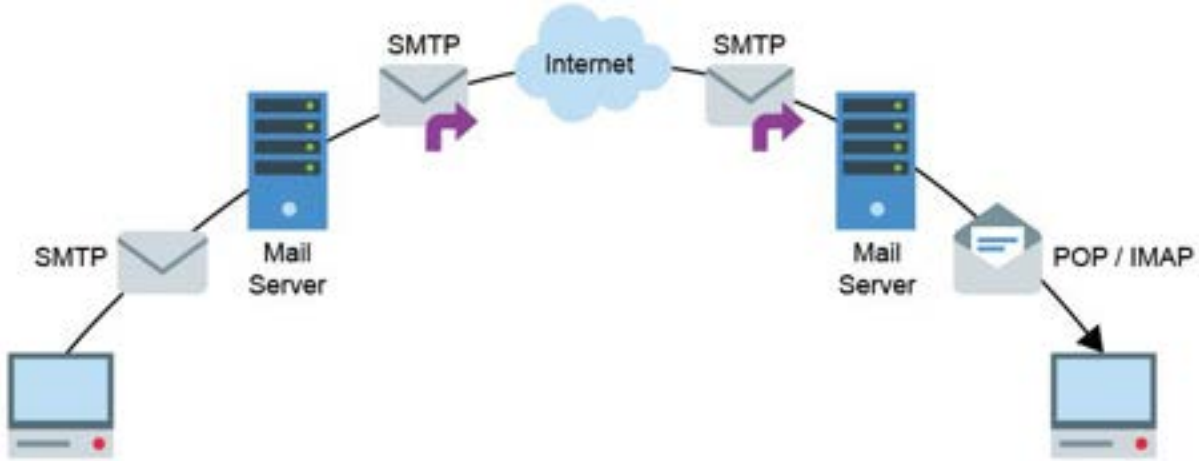
- These days, most websites use HTTPS as they deal with personal data
- Some specific types of site that use HTTPS are:
  - Banks – to prevent theft
  - Online shops – for bank and log in details
  - Social networks – for log in details and personal data

FTP

- File Transfer Protocol (FTP) is used for sending or retrieving files to or from a FTP server
  - A graphical interface will allow files to be dragged from your computer to the server

How email works

- When an email is sent from a computer it will first be sent to a mail server using the SMTP protocol
  - It is then forwarded on by other SMTP servers
  - When it reaches the destination mail server it is stored
  - The user's computer uses POP or IMAP to access the email



POP and IMAP

- These email retrieval protocols fetch message data and attachments from your remote mail server
  - **POP** (Post-Office Protocol) will download every new message to your local device and with them no longer being available on the server – This is similar to the protocol of sending a letter through a post office
  - **IMAP** (Internet Message Access Protocol) will leave the messages on a server. They can be accessed by multiple devices and they only are removed if the user deletes them

SMTP

- This is an email protocol used for sending email
  - Email software, such as Outlook, sends the email to the SMTP server used by the company or individual
  - The mail server then relays the message through various other servers known as mail relays
  - Finally, the email arrives at the destination mail server

Worksheet 5

- Now complete **Task 1** on **Worksheet 5**

TCP/IP protocol

- TCP (Transmission Control Protocol)
  - Breaks up messages sent over the Internet into small chunks called packets
  - Reassembles the packets at the other end
  - Detects errors
  - Resends lost messages
- IP (Internet Protocol)
  - Routes the individual packets from one IP address to another

The TCP/IP protocol stack

- The protocol stack defines four layers that enable communication on the Internet
  - This is a modular design with each layer being responsible for a small part of the communication process
- The four layers are divisions of network functionality, each carrying out different roles:
  - Application layer
  - Transport layer
  - Internet layer
  - Link layer

Sending data

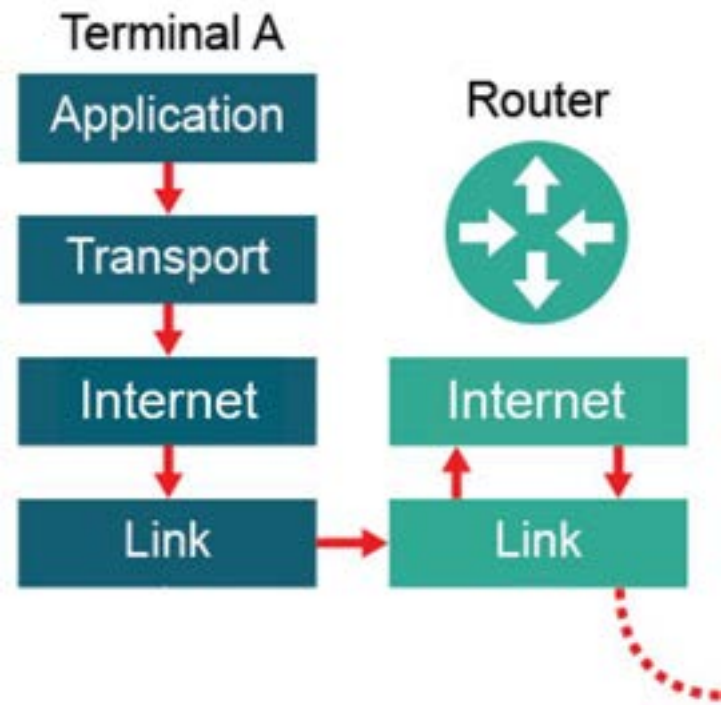
- The application you are using to send data will determine the correct protocol to use to communicate

Computer application produces data for sending

Splits data into packets

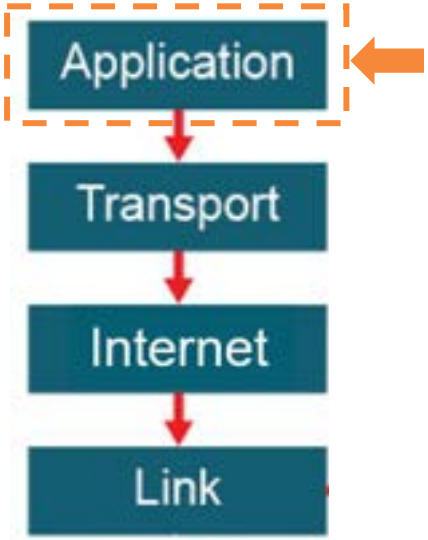
Sends and routes the packets

Physical hardware and cables



Application layer

- Applications such as email clients and web browsers create data to send in this layer
  - SMTP, FTP and HTTP operate in this layer
- Which application protocol would be selected by:
  - A browser connecting to a banking website?
  - An email program used to send a message?
  - File transfer software used to upload files to a website?



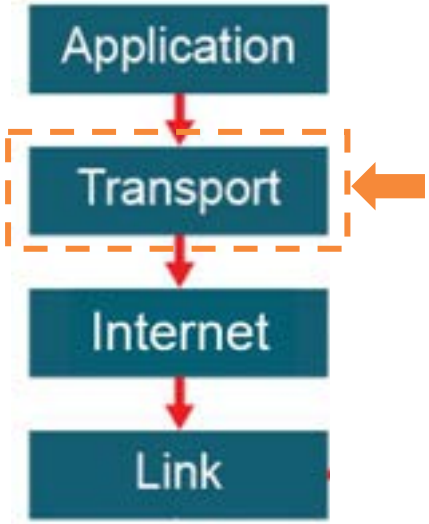
Application layer

Answers

- A browser connecting to a banking website
  - Uses HTTP
- An email program used to send a message
  - Uses SMTP
  - It also uses POP or IMAP to receive mail
- File transfer software used to upload files to a website
  - Uses FTP

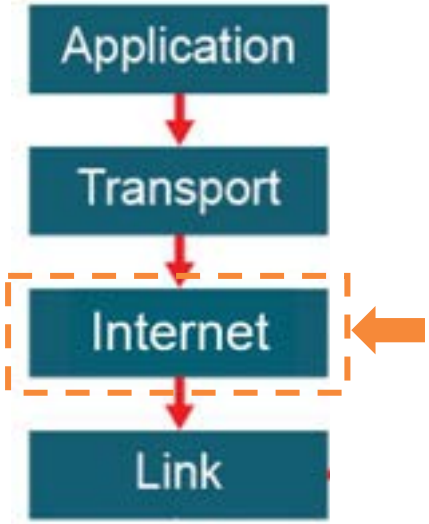
Transport layer

- The Transport layer creates the connection between two computers, or ‘hosts’
  - Data is then divided up into packets and given a packet number
  - Packets are reassembled by the recipient’s Transport layer
  - Lost packets are resent
  - This layer uses TCP (Transmission Control Protocol)



Internet layer

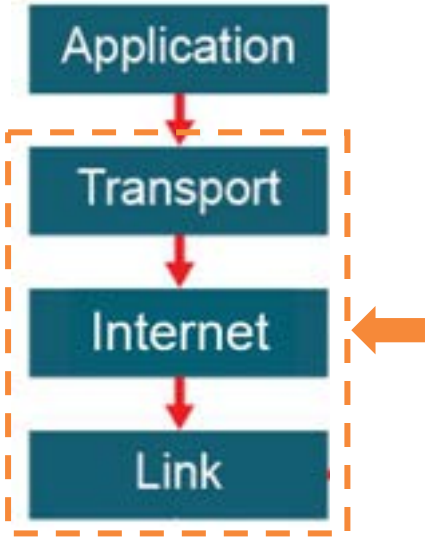
- The Internet layer is responsible for routing packets
  - Routers operate on this layer
- Source and destination addresses are written on to the packets ready for transmission
  - Which protocol operates on the Transport layer (in the last slide)?
  - Which protocol operates on the Internet layer?
  - Which protocols do you think operate in the Link layer?



Layer protocols

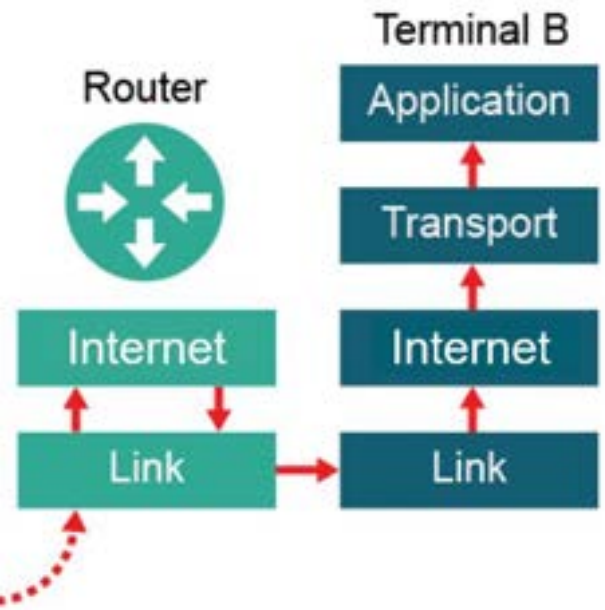
Answers

- Transport Layer
  - TCP (Transmission Control Protocol)
- Internet Layer
  - Internet Protocol (IP)
- Link Layer
  - MAC (Media Access Control)
  - Ethernet
  - Wi-Fi



Receiving data

- Data packets move back up the layers on arrival:



- The link layer removes the MAC address and passes packets up to the Internet layer
- The Internet layer removes the IP addresses and passes packets up to the Transport layer
- The Transport layer reassembles the packets and passes the data to the Application layer
- The Application layer uses the correct protocol to correctly display the data, web page or email for the user

The advantages of layers

- Layers are self-contained
  - The functionality of one layer can be changed without affecting the functionality of other layers
- Different hardware or software operates on a particular layer providing interoperability between providers and systems
  - This means that manufacturer's routers operating on the Internet layer, will operate with another manufacturer's Network Interface Cards (NICs) operating on the Link layer
  - Senders and receivers using different software and hardware can communicate using the same layer protocols

Worksheet 5

- Now complete **Task 1** on **Worksheet 5**

Plenary

- Work in a pair to decide what each of the following acronyms stand for
  - HTTP
  - HTTPS
  - FTP
  - POP
  - IMAP
  - SMTP
  - TCP/IP
- Which pair in the class got the most correct?

Plenary

Answers

- HTTP – Hypertext Transfer Protocol
- HTTPS – Hypertext Transfer Protocol Secure
- FTP – File Transfer Protocol
- POP – Post Office Protocol
- IMAP – Internet Message Access Protocol
- SMTP – Simple Mail Transfer Protocol
- TCP/IP – Transmission Control Protocol / Internet Protocol

Copyright

© 2024 PG Online Limited

The contents of this unit are protected by copyright.

This unit and all the worksheets, PowerPoint presentations, teaching guides and other associated files distributed with it are supplied to you by PG Online Limited under licence and may be used and copied by you only in accordance with the terms of the licence. Except as expressly permitted by the licence, no part of the materials distributed with this unit may be used, reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic or otherwise, without the prior written permission of PG Online Limited.

Licence agreement

This is a legal agreement between you, the end user, and PG Online Limited. This unit and all the worksheets, PowerPoint presentations, teaching guides and other associated files distributed with it is licensed, not sold, to you by PG Online Limited for use under the terms of the licence.

The materials distributed with this unit may be freely copied and used by members of a single institution on a single site only. You are not permitted to share in any way any of the materials or part of the materials with any third party, including users on another site or individuals who are members of a separate institution. You acknowledge that the materials must remain with you, the licencing institution, and no part of the materials may be transferred to another institution. You also agree not to procure, authorise, encourage, facilitate or enable any third party to reproduce these materials in whole or in part without the prior permission of PG Online Limited.

OCR
Oxford Cambridge and RSA
An OCR endorsed teaching and learning tool

GCSE OCR

Computer Science J277

1

Network threats

Unit 4 Network security and systems software

PG ONLINE

Objectives

- Understand forms of attack and threats posed to a network:
  - Malware
  - Social engineering including phishing
  - Brute force attacks
  - Denial of service attacks
  - Data interception and theft
  - SQL injection

Starter

- List at least **three** types of software that are made to harm a computer
  - How are these types of software prevented from carrying out harm?

Starter

Answers

- Harmful software includes:
  - Malware
    (viruses, worms, Trojans and ransomware)
- Prevention methods include:
  - Anti-malware, anti-virus, encryption, acceptable use policies, backup and recovery procedures

Malware

- Malware comes from two words:
  - **Mal**icious – to cause an act of harm
  - Soft**ware**
- Malware are executable programs that run on a computer
- One type of malware that exists is a computer virus
  - What are **two** other types of malware?

PG ONLINE

Malware - Viruses and worms

- Computer viruses infect computers
  - They replicate their code in other programs
  - They infect other computers
  - They harm the computer by deleting, corrupting or modifying files
- A worm replicates itself in order to spread to other computers
  - They might cause no damage to the attacked computers
  - They slow down networks and computers

PG ONLINE

Malware - Trojan horses

- During the Trojan War there is a story that the Greeks made a wooden horse and hid men inside
  - The Trojans brought the horse into the city, allowing the Greeks to open the city gates letting the army in to destroy Troy
- Computer Trojans are similar:
  - They have a program, game or cracked file which is something the user wants
  - They have negative program code which causes damage, takes control, or provides access to the computer

PG ONLINE

Malware - ransomware

- Ransomware is software which:
  - Holds a computer hostage by locking or encrypting data denying access to it
  - If the data is encrypted, not even a cyber security professional will be able to recover the data unless backups are available
  - Once a ransom is paid to the attacker, access is restored
- Should the ransom be paid?

PG ONLINE

Worksheet 4

- Now complete **Task 1** on **Worksheet 4**

Exploiting vulnerabilities

- There are a number of ways that hackers can exploit technical vulnerabilities. Two of these include:
  - **Unpatched software** – if software updates and security updates are not installed then the software will be vulnerable
  - **Out-of-date anti-malware** – if software, such as antivirus software, isn't regularly updated then it won't be able to detect the latest viruses
- What other technique can hackers use to gain access to passwords and personal data?

Social engineering

- Social engineering is the art of manipulating people into providing confidential data or unauthorised access to a computer system
  - What are two confidence tricks that would get someone to give you confidential or personal information?

PG ONLINE

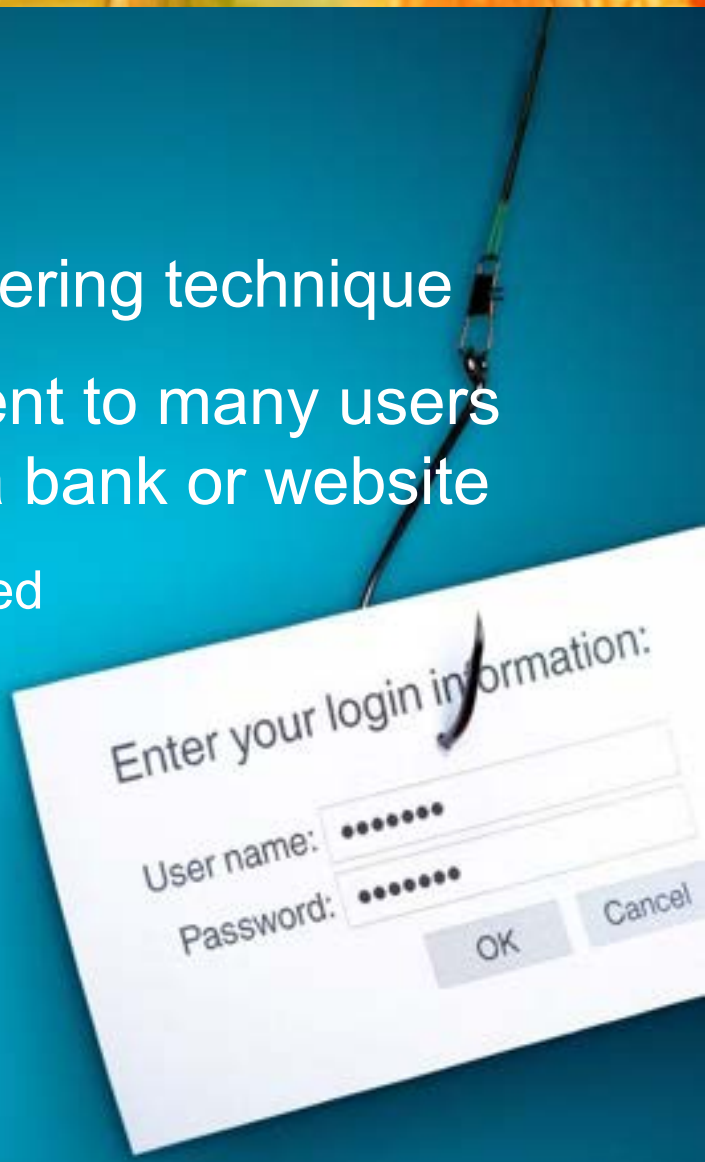
Shoulder surfing

- Shoulder surfing is the ability to get information or passwords by observing as someone types them in
- The following are two examples:
  - Looking over someone's shoulder
  - Using a CCTV camera
- What are **two** other ways that shoulder surfing could be carried out?

PG ONLINE

Phishing

- Phishing is a type of social engineering technique
- Emails, texts or phone calls are sent to many users commonly pretending to be from a bank or website
  - The ‘From’ email address may be forged
- These messages will try to get personal information such as:
  - Usernames
  - Passwords
  - Credit cards details
  - Other personal information



A typical phishing email

Halifax Bank Plc (no-reply@home.ne.jp) Add to contacts 25/06/2014

To: Recipients

Hi,

We're just checking this is the right email address for you.
Soon your email address will become your username to access Halifax Account - that makes it easier than remembring yet another username.

If this is the email address you want to use, all you have to do is click the link below
https://my.halifax.co.uk/your-account/verify-email-details?verificationCode=eee96442-51d6-4868-b0f3-a5484447eae8
We'll let you know when your username has been changed to your email address.
If you don't verify your email address you'll need to re-register if you want to view your bill online or make change to any of your accounts in the future.

Thanks
The Online Team
Halifax

What to look out for

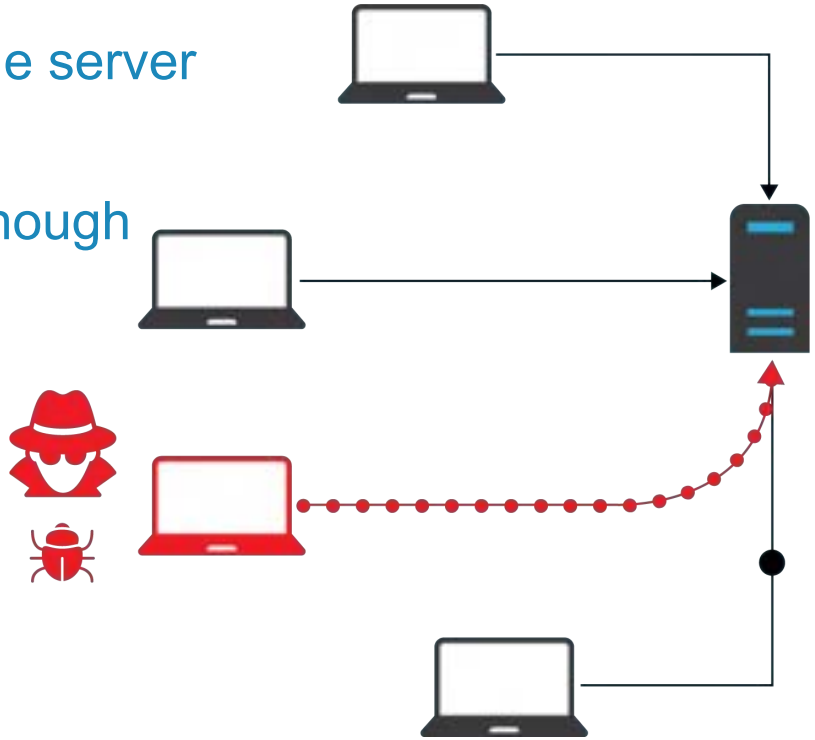
- **Greeting:** The phishers don't know your name – just your email address, so the greeting is not personalised
- **The sender's address** is often a variation on a genuine address
- **Forged link:** The link looks genuine, but it may not link to the website given. Roll your mouse over it to check
- **Request for personal information:** Genuine organisations never do this
- **Sense of urgency:** Criminals try to persuade you that something bad will happen if you don't act fast
- **Poor spelling and grammar**

Worksheet 4

- Now complete **Task 2** and **Task 3** on **Worksheet 4**

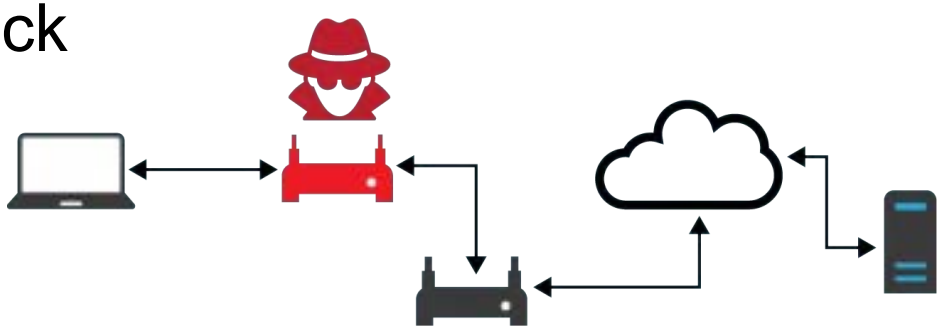
Denial of Service (DoS)

- In a denial of service attack, a hacker will use or infect a computer so that:
  - it sends as many requests to the server as it can (known as a flood)
  - the server can't respond fast enough so slows down or goes offline
- In a distributed denial of service attack (DDoS), many computers are used to send the requests



Man-in-the-middle attack

- A man-in-the-middle attack (MITM) allows the attacker to intercept communications between the user and server. The attacker can then:
  - eavesdrop to find passwords and personal information
  - add different information to a web page or other communication such as email
- Connecting to unencrypted Wi-Fi makes it easy to perform a MITM attack



PG ONLINE

Found – a free USB stick!

- Criminals sometimes leave a USB stick containing malware in a public place such as a company car park
  - An unsuspecting employee may pick it up and insert it into their computer
  - The malware can now install onto the computer so that a hacker can gain access to files, personal data and system resources
  - This technique is known as baiting

Data theft

- In 2014, the details of 125,000 students of Staffordshire University were stolen from a laptop left in a car
  - Data included the address, telephone number and email addresses for university applicants since 2006
  - Where could this information end up?
  - How could it be used?

Threats from digital devices

- Digital devices are often targeted by criminals
  - Loss of a mobile phone can lead to the loss of all the data stored on it, including passwords, account numbers and credit card details
  - Malware which targets digital devices may create ‘back doors’ to give malicious users access to your device
  - Many apparently legitimate apps are malicious and may lead to fraudulent charges on your phone bill or theft of personal information
- How can these risks be reduced on a mobile phone?

Protect your mobile phone

Answers

- Reducing threats on mobile devices
  - Use the password feature and choose a strong password
  - Make sure the data is encrypted
  - Do not follow links in suspicious emails
  - Think carefully before posting your mobile phone number on public websites
  - Don’t install apps without researching them first – if they require unnecessary extra permissions, don’t install them
  - Delete all information stored on your mobile before discarding

Brute force attacks

- In a brute force attack, a hacker will try every combination of password until the correct password is found
  - A computer program is usually used to do this as it can try millions of passwords per second

Password attempt	Outcome
0001	Incorrect password
0002	Incorrect password
0003	Incorrect password
0004	Incorrect password
0005	Password found

Worksheet 4

- Now complete **Task 3** and **Task 4** on **Worksheet 4**

SQL injection

- SQL (Structured Query Language) is a database query language
- SQL injection takes advantage of web input forms to access or destroy data
  - SQL commands can be input into web forms instead of the expected ‘real’ data
  - This can be interpreted by vulnerable web applications and end up causing damage or releasing personal information

Web form SQL

- The following line of SQL may be used in conjunction with a web form to display product data:

```
txtProductId = getRequestString(“ProductId");

SELECT *
FROM Product
WHERE ProductId = '' + txtProductId;
```

Product Id: [ ] Display data

Injecting SQL

- Entering '*00237*' as the Product ID would update the criteria in the SQL and display all of the record data for that product for example:

WHERE ProductId = 00237;

Product ID	Name	Description	Price	Manufacturer	Availability
00237	Small Tiger	Soft toy	11.99	StripeyToys	In stock

- Entering '*00237; DROP TABLE Customers*' adds an SQL command to delete all company customer data:

WHERE ProductId = 00237; DROP TABLE Customers

Worksheet 4

- Now complete **Task 5** on **Worksheet 4**

Plenary

- In a pair, come up with one accurate sentence to describe each of the following forms of attack:
  - Malware
  - Phishing
  - Social engineering
  - Brute force attacks
  - Denial of service attacks
  - Data Interception and theft
  - SQL injection

Plenary

Answers

- Malware – software that aims to harm computers and/or data
- Phishing – emails that pretend to be from legitimate companies but actually try to gain personal information
- Social engineering – deception to gain personal information
- Brute force attacks – trying all possible passwords until the correct one is found
- Denial of service attacks – flooding a server or network with pointless requests so that it fails or slows down
- Data Interception and theft – stealing data or intercepting it with a man-in-the-middle attack
- SQL injection – using web forms to add SQL instructions to a query that cause data loss or the revealing of personal information

Copyright

© 2024 PG Online Limited

The contents of this unit are protected by copyright.

This unit and all the worksheets, PowerPoint presentations, teaching guides and other associated files distributed with it are supplied to you by PG Online Limited under licence and may be used and copied by you only in accordance with the terms of the licence. Except as expressly permitted by the licence, no part of the materials distributed with this unit may be used, reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic or otherwise, without the prior written permission of PG Online Limited.

Licence agreement

This is a legal agreement between you, the end user, and PG Online Limited. This unit and all the worksheets, PowerPoint presentations, teaching guides and other associated files distributed with it is licensed, not sold, to you by PG Online Limited for use under the terms of the licence.

The materials distributed with this unit may be freely copied and used by members of a single institution on a single site only. You are not permitted to share in any way any of the materials or part of the materials with any third party, including users on another site or individuals who are members of a separate institution. You acknowledge that the materials must remain with you, the licencing institution, and no part of the materials may be transferred to another institution. You also agree not to procure, authorise, encourage, facilitate or enable any third party to reproduce these materials in whole or in part without the prior permission of PG Online Limited.

OCR
Oxford Cambridge and RSA
An OCR endorsed teaching and learning tool

GCSE OCR

Computer Science J277

2

Identifying and preventing vulnerabilities

Unit 4 Network security and systems software

PG ONLINE

Objectives

- Identify and understand the prevention of vulnerabilities including the use of:
  - penetration testing
  - anti-malware software
  - firewalls
  - user access levels
  - passwords
  - encryption
  - physical security

Starter

- Computers are vulnerable to many threats such as:
  - Malware including viruses
  - Social engineering
  - Denial of service attacks
  - Brute-force attacks
  - Data interception
- What methods do you use at school or home to reduce the risk of these threats?

PG ONLINE

Preventing vulnerabilities

Answers

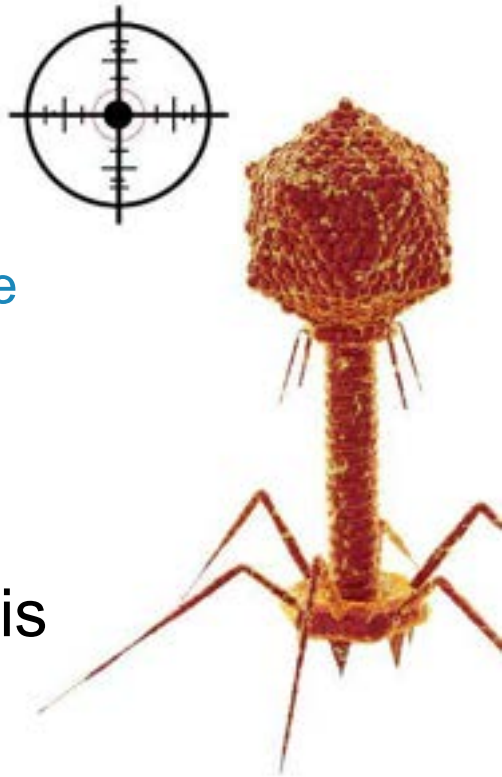
- There are many ways that vulnerabilities are protected against – in particular:
  - penetration testing
  - anti-malware software – including anti-virus software
  - firewalls
  - user access levels
  - passwords
  - encryption
  - physical security

Penetration testing

- ‘Pen’ testing is the practice of deliberately trying to find security holes in your own systems
- The goal of penetration testing
  - identify the targets of potential attacks
  - identify possible entry points
  - attempt to break in
  - report back the findings

Anti-malware software

- Anti-malware software will detect malware such as viruses, worms, trojans, and spyware
  - When a virus or new malware is detected it is sent to the anti-virus company
  - They verify it is malware then create a signature of the virus
  - They then add it to their virus database and tell computers to run an update
- Viruses can morph to avoid detection. This makes it harder to create a signature

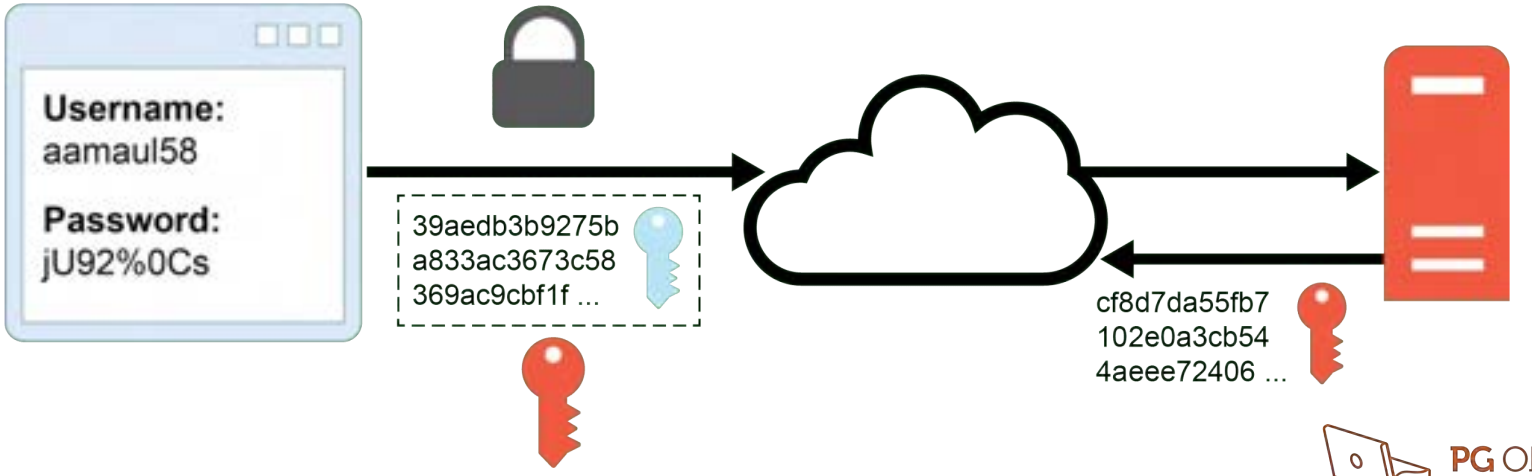


Encryption

- Encryption is a way of securing data so that it cannot be read without the encryption key
  - Passwords stored by websites are almost always stored in an encrypted form
  - If a hacker obtains the data in the password database, they won't be able to easily read the passwords
  - Devices and computers can also have their storage and hard disk data encrypted
- How do you know if a website you are using is encrypting the communication between your computer and the website?

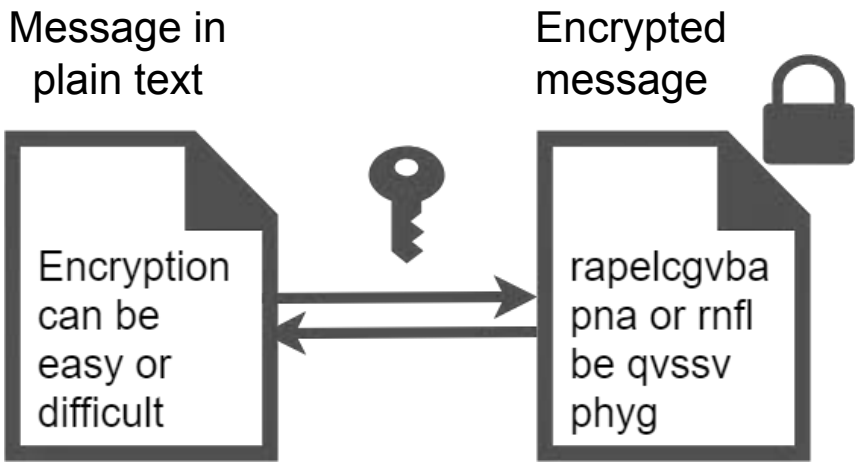
Encryption of transmitted data

- Websites using HTTPS (Secure HyperText Transfer Protocol) encrypt the connection to the server
  - Your web browser sends its key and form data, encrypting it with the server's key
  - The server encrypts the web page you request using your web browser's key



Encryption of individual files

- Files can be encrypted individually on a computer using a password
- They can then only be viewed by people who have the password
- Software such as zip files allow encryption to be applied
  - The message on the right uses a very simple encoding. How does it work?



Encryption of drives

- Ordinary computer drives can be encrypted so that a password is needed to access the information
  - This prevents a hacker from understanding the contents of the hard drive if it is removed and installed in a different computer
- For removable media, special hardware can be purchased which encrypts data on the disk

Worksheet 2

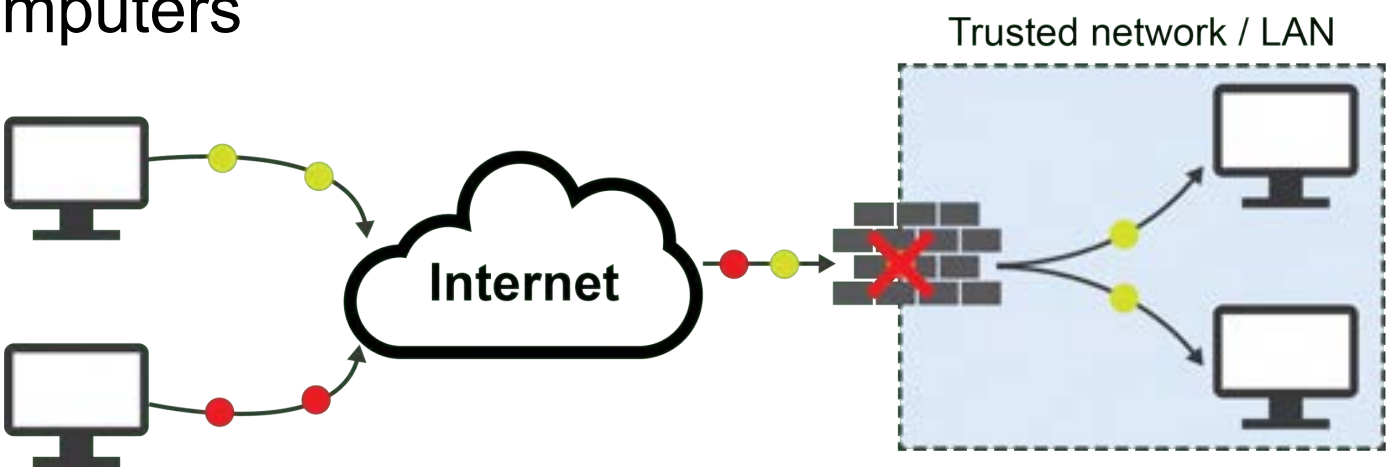
- Now complete **Task 1** on **Worksheet 2**

Firewalls

- Separate a trusted network from an untrusted network (normally the Internet)
- Data is sent around a network in small packets of information
- These packets are checked to see where they are coming from and going to
  - Packets that don’t match filtering rules are dropped
  - This is known as a packet filter
- Firewalls can be run on dedicated hardware or as software

Firewalls

- Firewalls may be built into your hardware
  - This may be a dedicated unit to the task of being a firewall
  - Alternatively, it may be built into other devices such as a home Wi-Fi router
- The firewall will detect packets from malicious computers

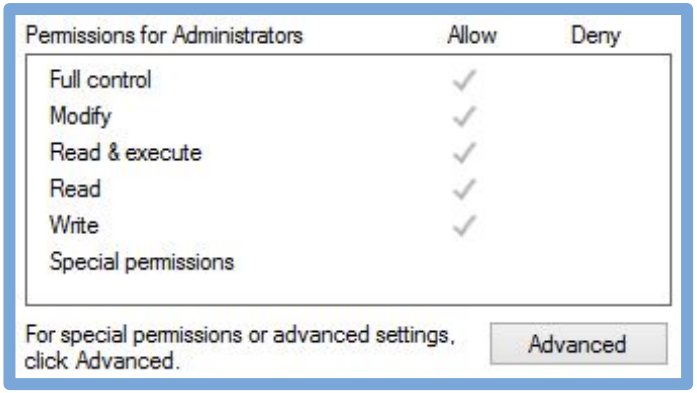


Firewall capabilities

- Firewall features:
  - Prevents attackers from gaining access to computers on a network
  - Can block certain malicious computers by filtering packets from a certain IP (Internet Protocol) address
  - Can prevent access to certain ports on the network. This is known as port blocking
  - Malicious or inappropriate websites can be blocked
  - Dedicated hardware firewalls are expensive
  - Software firewalls will slow down a computer

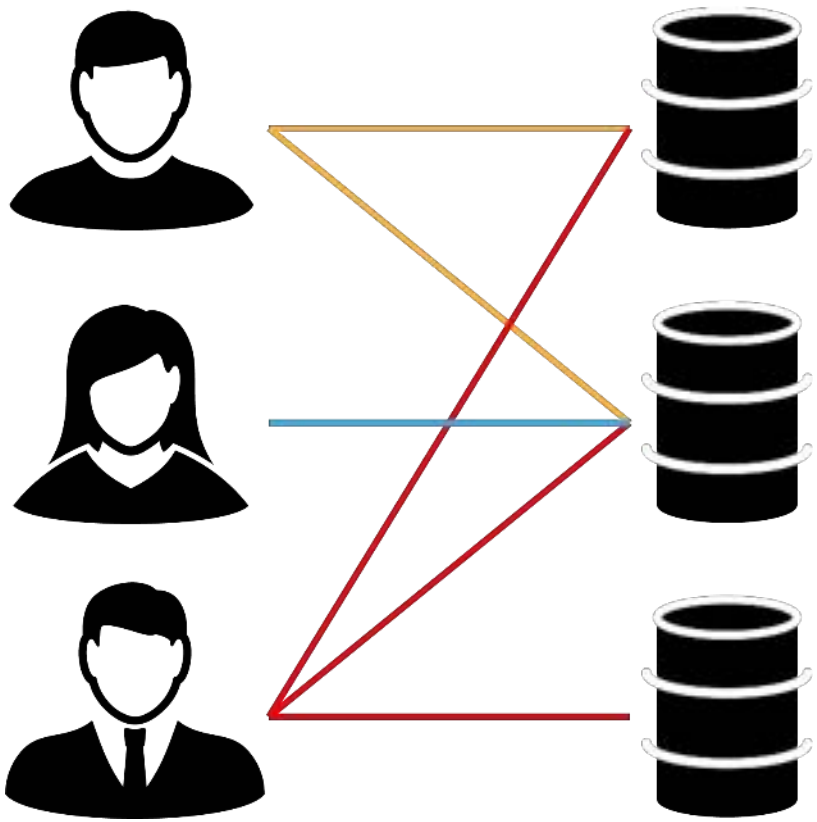
User access levels

- Access rights may be set on disks, folders and even individual files
  - How are access rights used in school?



Permissions for Administrators	Allow	Deny
Full control	✓	
Modify	✓	
Read & execute	✓	
Read	✓	
Write	✓	
Special permissions		

For special permissions or advanced settings, click Advanced. Advanced

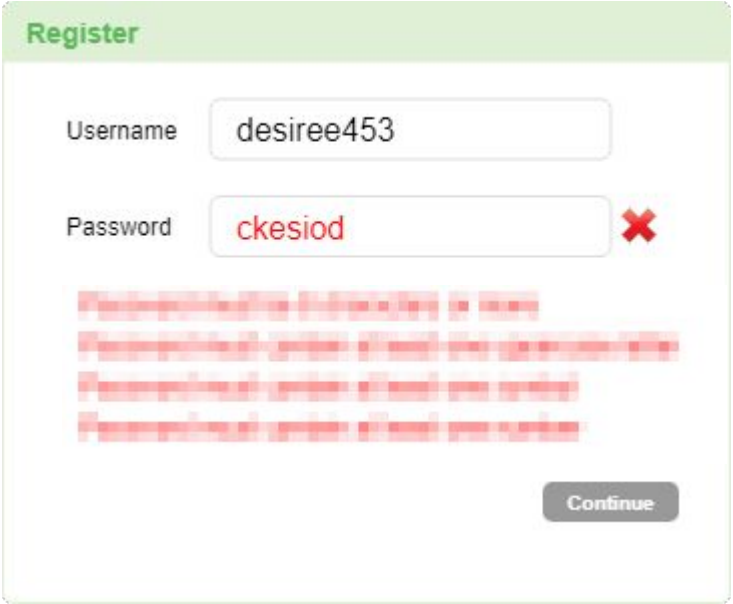


Worksheet 2

- Now complete **Task 2** on **Worksheet 2**

Password policy

- Passwords are often checked as they are created to make sure that the conform to the parameters given in a required policy
  - Four error messages have been given for this password. What are they likely to be?



Password policies

- Organisations and computer systems will often have password policies. These will make sure that your chosen password has features like:
  - Minimum length of characters
  - Include at least one lowercase letter
  - Include at least one uppercase letter
  - Include at least one symbol
  - Change password every month
- Pa55w*rd meets all the above criteria. Is it a good password?

Physical security

- Physical security is where hardware, software and networks are protected by physical methods
- An example would be security lighting
  - What are **five** other methods of physical security that are used?

Physical security

Answers

- Physical security methods include:
  - CCTV
  - Intruder alarms
  - Locks on doors or cabinets
  - Fences, walls, barbed wire
  - Security guards
  - ID cards and signs
  - Turnstiles and gates

Worksheet 4

- Now complete **Task 3** and **Task 4** on **Worksheet 4**

Plenary

- For each of the following, state whether it is:
  - A threat to a digital system
  - A way of protecting a digital system

Viruses	Encryption	Trojans	Penetration testing
Physical security	User access levels	Anti-malware	Firewalls
Ransomware	Worms	Social engineering	

Plenary

Answers

Threats

- Viruses
- Ransomware
- Worms
- Trojans
- Social engineering

Protections

- Firewalls
- Encryption
- Penetration testing
- Anti-malware
- User access levels
- Physical security

Copyright

© 2024 PG Online Limited

The contents of this unit are protected by copyright.

This unit and all the worksheets, PowerPoint presentations, teaching guides and other associated files distributed with it are supplied to you by PG Online Limited under licence and may be used and copied by you only in accordance with the terms of the licence. Except as expressly permitted by the licence, no part of the materials distributed with this unit may be used, reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic or otherwise, without the prior written permission of PG Online Limited.

Licence agreement

This is a legal agreement between you, the end user, and PG Online Limited. This unit and all the worksheets, PowerPoint presentations, teaching guides and other associated files distributed with it is licensed, not sold, to you by PG Online Limited for use under the terms of the licence.

The materials distributed with this unit may be freely copied and used by members of a single institution on a single site only. You are not permitted to share in any way any of the materials or part of the materials with any third party, including users on another site or individuals who are members of a separate institution. You acknowledge that the materials must remain with you, the licencing institution, and no part of the materials may be transferred to another institution. You also agree not to procure, authorise, encourage, facilitate or enable any third party to reproduce these materials in whole or in part without the prior permission of PG Online Limited.

OCR
Oxford Cambridge and RSA
An OCR endorsed teaching and learning tool

GCSE OCR

Computer Science
J277

3

Operating systems software

Unit 4 Network security and systems software

PG ONLINE

Objectives

- Explain the need for the following functions of an operating system:
  - User interface
  - Memory management and multitasking
  - Peripheral management and drivers
  - User management
  - File management

Starter

- Microsoft Windows is one example of an operating system
  - Name as many different operating systems as you can

PG ONLINE

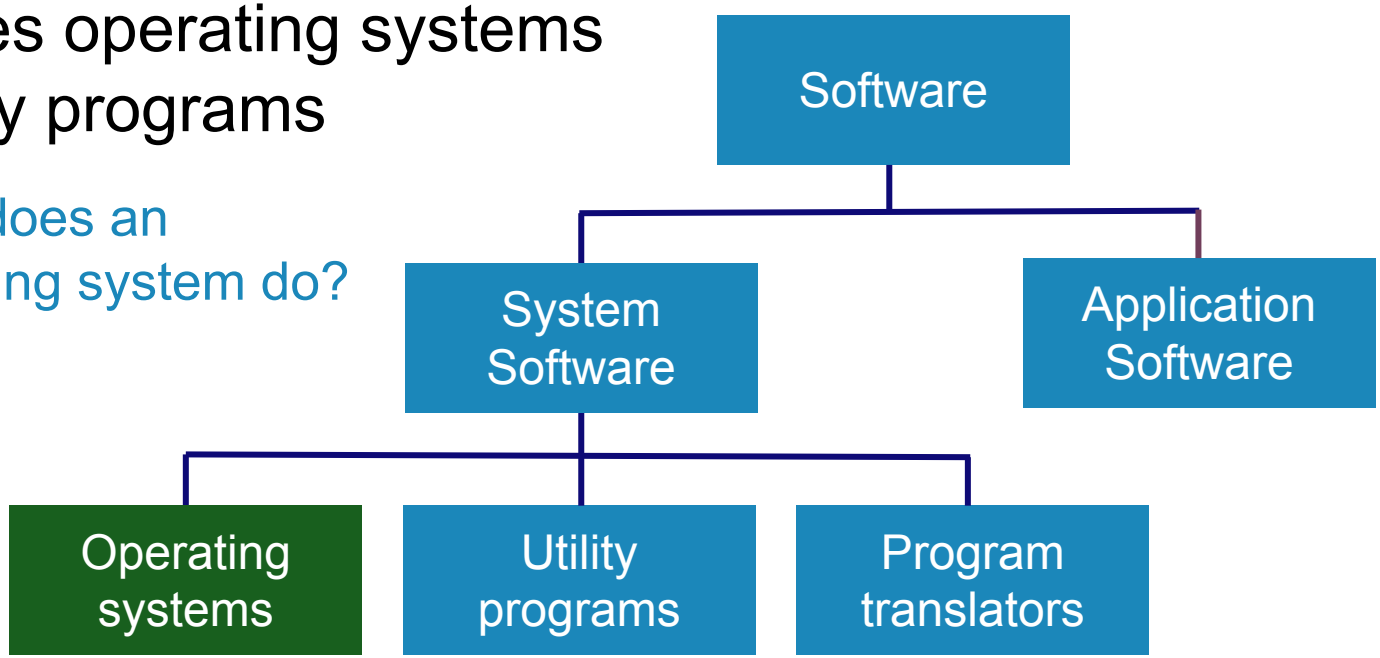
Some operating systems

Answers

- Microsoft
  - MS-DOS (1981-2000)
  - Windows (1985-present)
- Google
  - Android (2008-present)
  - Chrome OS (2011-present)
- Apple
  - macOS (2001-present)
- Linus Torvalds and other contributors
  - Linux (1991-present)

Types of system software

- System software is the software that provides a platform for other software to work
- It includes operating systems and utility programs
  - What does an operating system do?



Operating Systems

Answers

- Operating systems manage computer hardware, users and the resources used by software
- They are responsible for managing:
  - The user interface
  - Memory management
  - Multitasking
  - Peripheral management and drivers
  - User management
  - File management

User interfaces

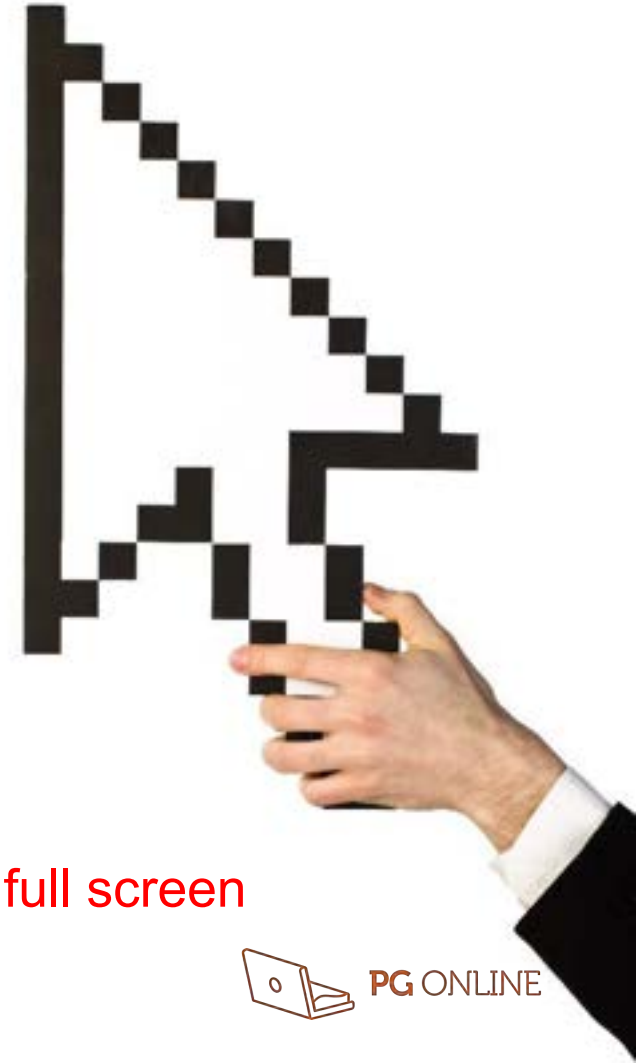
- User interfaces provide a method for users to interact with a computer
- One type of interface is the graphical user interface (GUI)
  - What features do GUIs have?
  - How do the features differ between desktop computers, tablets and smartphones?

PG ONLINE

Graphical user interfaces

Answers

- Desktop interfaces
  - Pointers
  - Windows
  - Menus
  - Icons
  - Drag and drop
- Smartphone and touch interfaces
  - No pointers in general as they cannot be seen under a finger
  - Windows (for apps) generally take up the full screen



Types of User Interface

- GUI: Graphical User Interface
  - WIMP is a commonly used acronym in traditional GUIs: **W**indows, **I**cons, **M**enus and **P**ointers
- Menu-driven interface
- CLI: Command Line Interface
  - Text commands are entered into the interface
- Voice activated
- Real-time
  - sensors detect inputs

Command Line Interfaces

- Text commands are entered into the computer
  - This requires very little processing and can save on the need for an expensive graphics card
- For expert users who know the commands it can be far faster to enter commands than using a mouse
  - A command line interface requires much less hard disk storage space
  - It also will require less RAM

PG ONLINE

Menu-driven interfaces

- Menu driven interfaces are often used by devices that have a dedicated function including
  - MP3 players
  - ATM machines
  - Self service checkouts

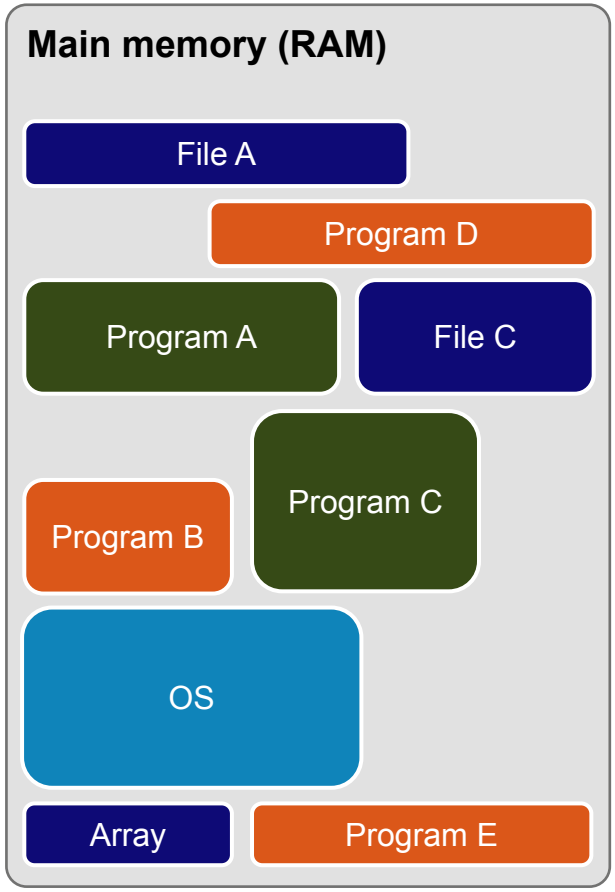
PG ONLINE

Worksheet 3

- Now complete **Task 1** and **Task 2** on **Worksheet 3**

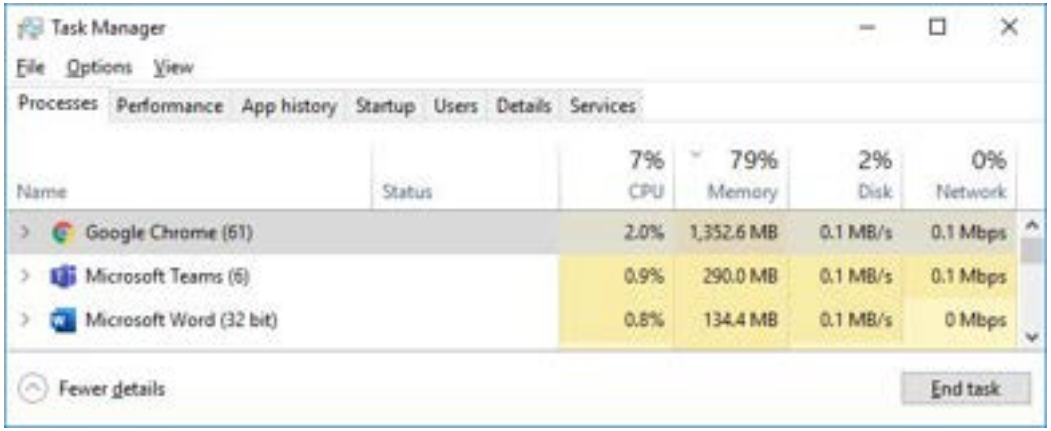
Memory management

- To run a program, the computer must copy the program from storage into main memory – **Why?**
  - Data used by the program is copied into main memory
  - The operating system keeps a record of where each program and its data are located
  - The operating system will make sure not to overwrite existing programs and data



Multi-tasking

- Multi-tasking is where an operating system manages many tasks happening at the same time
  - This may be having a web browser open whilst playing music and messaging friends
  - Many other background processes also are running
  - The operating system decides which process will next run on the CPU and for how long



PG ONLINE

Interrupts

- **Interrupts** are signals sent to the CPU by external devices to indicate an event that needs immediate attention
  - They tell the CPU to suspend its current activities and execute appropriate instructions
- **Hardware interrupts** are generated by hardware devices – for example, **printer out of paper**
- **Software interrupts** are generated by programs – for example, a **divide-by-zero error** will cause an error message to be displayed

Peripheral management

- The operating system must manage:
  - Getting inputs from and sending outputs to peripheral devices
- Peripherals include mice, keyboard, printers, displays, digital cameras and graphics tablets
  - What item of software needs to be installed for each peripheral to work?

Device drivers

- A device driver is a program that controls peripheral devices such as printers, mice and displays
  - Each device communicates with the OS via its own driver
  - Many device drivers come with an operating system but if you buy a brand new type of device, it will be supplied with a driver, which you will need to install

Sending data to a printer

- The computer can send data to a printer much faster than it can be printed
- The data will be sent to a print queue or buffer which is a special area of memory in a computer print server or the printer itself
  - Any data in the print queue is transmitted to the printer, typically a page at a time
  - The printer will send its status back to the operating system which reports any errors to the user. For example, the operating system may display an “out of paper” error message

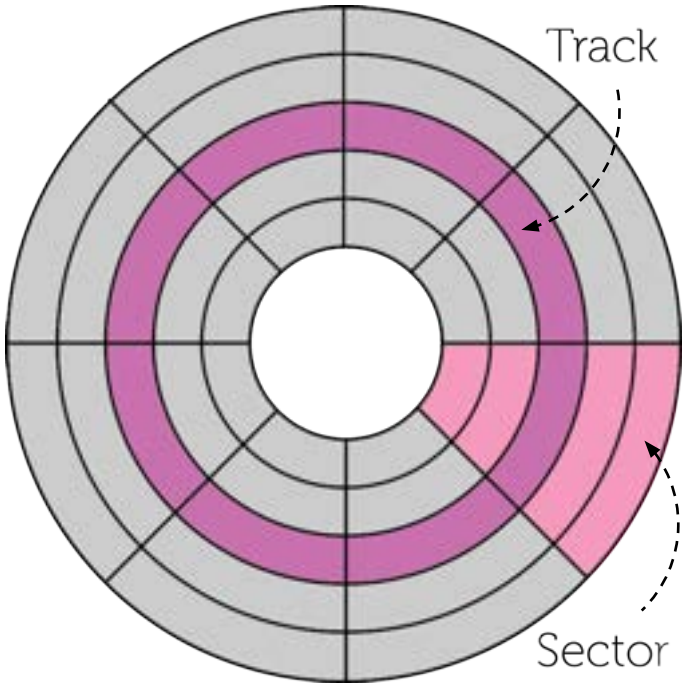
The print queue

- The screenshot shows a print queue in action
- It shows the status of each job in the queue, and whether it is printing or waiting its turn

Document Name	Status	Owner	Pages	Size	Submitted
Full page photo	Printing	Guest	N/A		13:27:41 30/06/2015
Full page photo	Spooling	Guest	1	5.25 MB	13:27:41 30/06/2015
Full page photo	Spooling	Guest	1	7.00 MB	13:27:37 30/06/2015

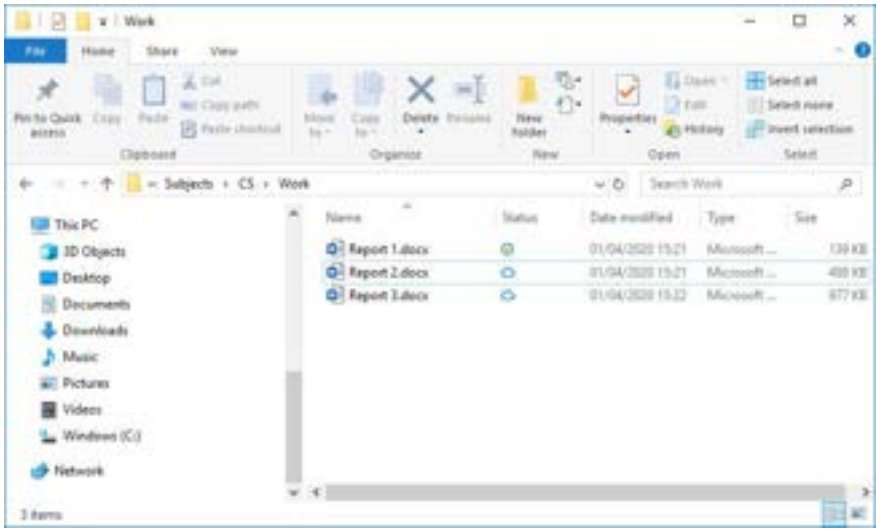
Disk and file management

- The hard disk in a computer is a storage peripheral
- The operating system:
  - manages where on the disk files are written
  - keeps track of where they are so they can be retrieved
  - makes sure no file overwrites another file



File management

- File management is carried out by the operating system
- The following features are available:
  - Naming files
  - Allocating files to folders
  - Moving files
  - Saving files
  - Copying files
  - Deleting files



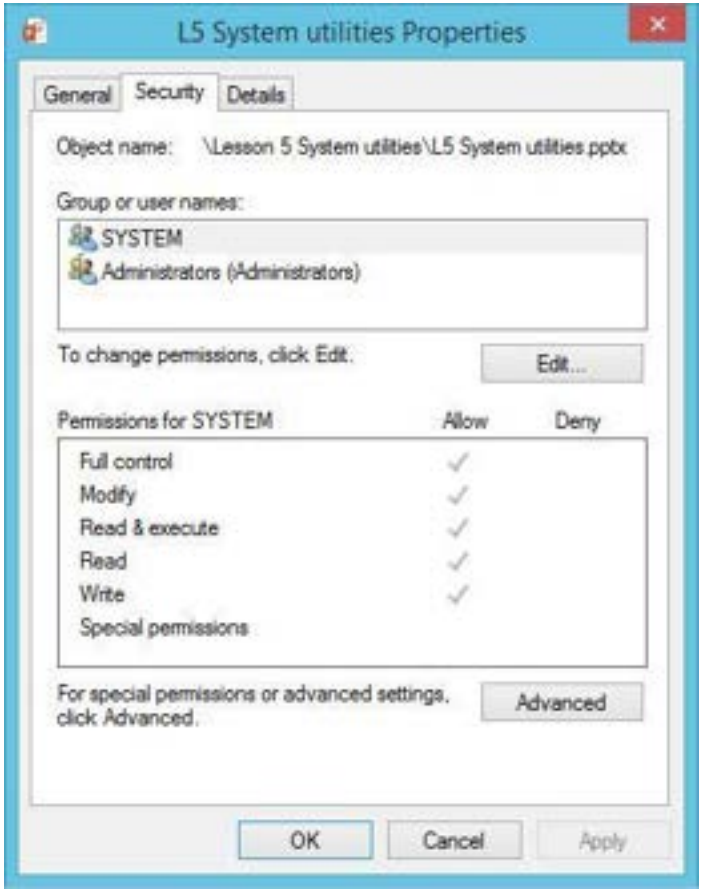
User management

- The operating system is responsible for user logins and passwords

- It will store all users and their passwords in a file or database

Access rights

- Access rights:
  - If a computer is used by more than one person, each user should be able to see only their own files
  - Users and system administrators have different levels of access rights
  - Some users may be allowed to read files but not edit them



Worksheet 3

- Now complete **Task 3** on **Worksheet 3**

Plenary

- Explain to a partner each of the following functions of an operating system
  - User interface
  - Memory management
  - Multitasking
  - Peripheral management and drivers
  - User management
  - File management

Plenary

Answers

- Operating system functions:
  - User interface – a method of the user interacting with the computer – windows, icons, menus, pointers, drag/drop – can be GUI, menu interface or command line interface
  - Memory management – manages programs and data stored in RAM, frees up memory when a program is closed
  - Multitasking – running two or more programs at once
  - Peripheral management and drivers – the management of devices such as mice and printers
  - User management – management of usernames and passwords along with what they have permission to access
  - File management – the organisation of files on storage drives

Copyright

© 2024 PG Online Limited

The contents of this unit are protected by copyright.

This unit and all the worksheets, PowerPoint presentations, teaching guides and other associated files distributed with it are supplied to you by PG Online Limited under licence and may be used and copied by you only in accordance with the terms of the licence. Except as expressly permitted by the licence, no part of the materials distributed with this unit may be used, reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic or otherwise, without the prior written permission of PG Online Limited.

Licence agreement

This is a legal agreement between you, the end user, and PG Online Limited. This unit and all the worksheets, PowerPoint presentations, teaching guides and other associated files distributed with it is licensed, not sold, to you by PG Online Limited for use under the terms of the licence.

The materials distributed with this unit may be freely copied and used by members of a single institution on a single site only. You are not permitted to share in any way any of the materials or part of the materials with any third party, including users on another site or individuals who are members of a separate institution. You acknowledge that the materials must remain with you, the licencing institution, and no part of the materials may be transferred to another institution. You also agree not to procure, authorise, encourage, facilitate or enable any third party to reproduce these materials in whole or in part without the prior permission of PG Online Limited.

OCR
Oxford Cambridge and RSA
An OCR endorsed teaching and learning tool

GCSE OCR

Computer Science J277

4

Utility software

Unit 4 Network security and systems software

PG ONLINE

Objectives

- Describe the purpose and functionality of common utility software including:
  - Encryption software
  - Defragmentation software
  - Data compression software

Utility software

- Operating systems provide some core functions such as memory management
  - What are **four** other core functions they provide?
- Computers also have utility software which helps to keep them working well
  - What are **four** types of utility software that operating systems often make use of?

PG ONLINE

Utility software

Answers

- Operating system functions (from the previous topic)
  - User interface
  - Multitasking
  - Peripheral management and drivers
  - User management, file management
- Utility software
  - Encryption software
  - Defragmentation software
  - Data compression
  - Disk clean-up tools, disk formatters, anti-virus software

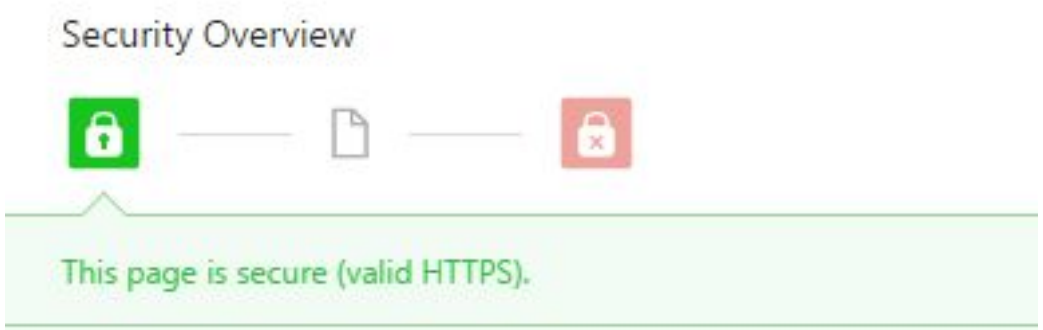
What is utility software?

- The operating system controls and manages the computer system
- Utility programs are system software (used to manage a computer system) that are useful but not essential
  - They are often packaged as part of the operating system
  - Some utility programs, such as anti-virus software, are bought as stand-alone software programs

PG ONLINE

Encryption software

- Encryption software is used to encode text so that it cannot be understood without knowing the key to decode it
  - How do you know if a connection to a website is encrypted?
  - What organisations encrypt their emails or file attachments?



Encryption software

Answers

- Encryption for connections to websites
  - Web browsers usually show a padlock for sites that are using HTTPS (Hypertext Transfer Protocol Secure)
- Organisations that use email encryption, or encryption of attachments include:
  - Councils when discussing sensitive personal information such as special needs or disabilities
  - Accountants discussing financial information of companies
  - Companies sharing confidential internal files

Encryption tools

- Encryption software utilities are commonly used to:
  - Encrypt entire hard disks so that the data cannot be understood if the computer is stolen
  - Encrypt files and folders on a portable disk such as USB removable storage
  - Encrypt communication with websites
  - Encrypt data in an organisation's database – especially sensitive data such as passwords

PG ONLINE

Worksheet 4

- Now complete **Task 1** on **Worksheet 4**

Disk defragmentation

- When you save a large file it may not fit on the disk in consecutive memory locations
- The file is referred to as ‘fragmented’
  - In order to read the file, the drive head will need to move to more locations
  - This makes it far slower to access files and programs
- Defragmenting the hard disk reorganises files so they are stored together
  - Read times are far faster
  - Free space is also in one place so new files do not need to be fragmented

Defragmentation

- Each colour below represents the fragments of one file

Before:
Files stored are fragmented

After:
Files and free space now defragmented

PG ONLINE

Data compression software

- Utilities such as WinZip enable users to compress and decompress files or folders
  - This reduces the amount of storage space needed on a disk
  - If sending the files, it reduces the amount of data that is sent
  - Compression may allow attachments to be sent via email that would otherwise have been larger than a file-size limit
  - More data can be stored on backup media

Worksheet 4

- Now complete **Task 2** and **Task 3** on **Worksheet 4**

Plenary

- With a partner, fill in the gaps below

_____ perform extra functionality beyond the core features of an operating system. They are used for _____ and to help keep the system secure. _____ compression reduces a file's file size.

_____ software will _____ a file so that it cannot be read without a decryption _____.

_____ software will put all _____ of a file together so that the file can be read faster.

Fragments Encode Encryption Utility software

Housekeeping Compression Defragmentation Key

Plenary

Answers

- With a partner, fill in the gaps below

Utility software perform extra functionality beyond the core features of an operating system. They are used for housekeeping and to help keep the system secure. Data compression reduces a file's file size. Encryption software will encode a file so that it cannot be read without a decryption key.
Defragmentation software will put all fragments of a file together so that the file can be read faster.

Fragments Encode Encryption Utility software

Housekeeping Compression Defragmentation Key

Copyright

© 2024 PG Online Limited

The contents of this unit are protected by copyright.

This unit and all the worksheets, PowerPoint presentations, teaching guides and other associated files distributed with it are supplied to you by PG Online Limited under licence and may be used and copied by you only in accordance with the terms of the licence. Except as expressly permitted by the licence, no part of the materials distributed with this unit may be used, reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic or otherwise, without the prior written permission of PG Online Limited.

Licence agreement

This is a legal agreement between you, the end user, and PG Online Limited. This unit and all the worksheets, PowerPoint presentations, teaching guides and other associated files distributed with it is licensed, not sold, to you by PG Online Limited for use under the terms of the licence.

The materials distributed with this unit may be freely copied and used by members of a single institution on a single site only. You are not permitted to share in any way any of the materials or part of the materials with any third party, including users on another site or individuals who are members of a separate institution. You acknowledge that the materials must remain with you, the licencing institution, and no part of the materials may be transferred to another institution. You also agree not to procure, authorise, encourage, facilitate or enable any third party to reproduce these materials in whole or in part without the prior permission of PG Online Limited.

OCR
Oxford Cambridge and RSA
An OCR endorsed teaching and learning tool

GCSE OCR

Computer Science
J277

Ethical and cultural issues

Unit 5
Impacts of digital technology

PG ONLINE

1

Objectives

- Discuss the impacts of digital technology on the wider society including:
  - Ethical issues
  - Cultural issues

Starter

- In January 2020, London's Metropolitan Police began to use live facial recognition
  - The system identifies people on a 'watchlist'
  - If it finds a match it can alert officers so that they can stop the person
- What are the advantages of such as system?
  - Are there any disadvantages or ethical concerns?

Advantages

- The system is used to find criminals wanted by the police or courts
  - They may be at risk of harm to themselves or others
  - They may be reported as missing
- The system is hoped to ‘deter and prevent crime’
  - It may also save the police time

PG ONLINE

Ethical Concerns

- Such systems use personal biometric data and some people may not consent to this
- The systems may alter public behaviour
  - For example, people may no longer wish to take part in peaceful public protests
- There is evidence that facial recognition technology may be more likely to misidentify children or people of colour

Ethics of CS technologies

- New digital technologies are developed very quickly, however, they often raise serious ethical concerns to many people
  - One issue is whether social networks should limit free speech
  - What other ethical issues do social networks create?

PG ONLINE

Internet addiction

- Many users of social networks have symptoms similar to an addiction
- Software is often designed with features that make people want to use it more
  - What limits, if any, should be placed on companies and the way they design their software?

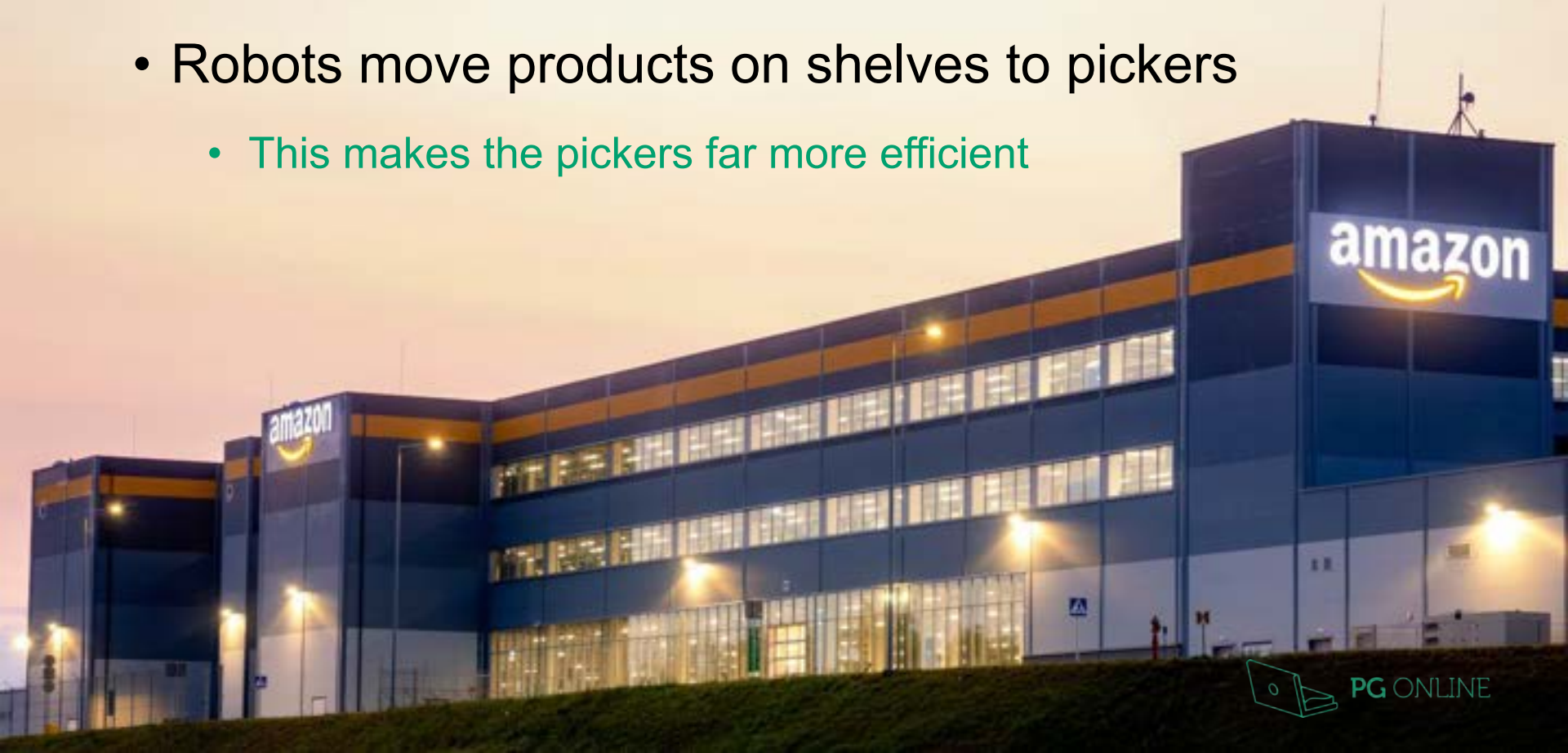
PG ONLINE

Worksheet 1

- Complete **Task 1** on **Worksheet 1**

Case study: Amazon fulfilment

- Amazon makes use of large warehouses called fulfilment centres
- Robots move products on shelves to pickers
  - This makes the pickers far more efficient



PG ONLINE

What jobs can robots do?

- Cars have been built by robots for many years
- Fast food restaurants now allow customers to order their own food
- Robots such as Maidbot are able to clean hotel rooms
  - In 2022, over 1 million lawn mowing units and almost 3 million robot vacuums were sold
- Are any groups of people particularly likely to be affected by these changes?

Computers in healthcare

- There are more than 300,000 healthcare apps for smartphones
- By 2020, more than 50% of smartphone users had downloaded a health app
- Wearable technology can offer continuous, long-term monitoring
  - This can lead to improvements in the way that diseases like epilepsy, asthma and diabetes are managed

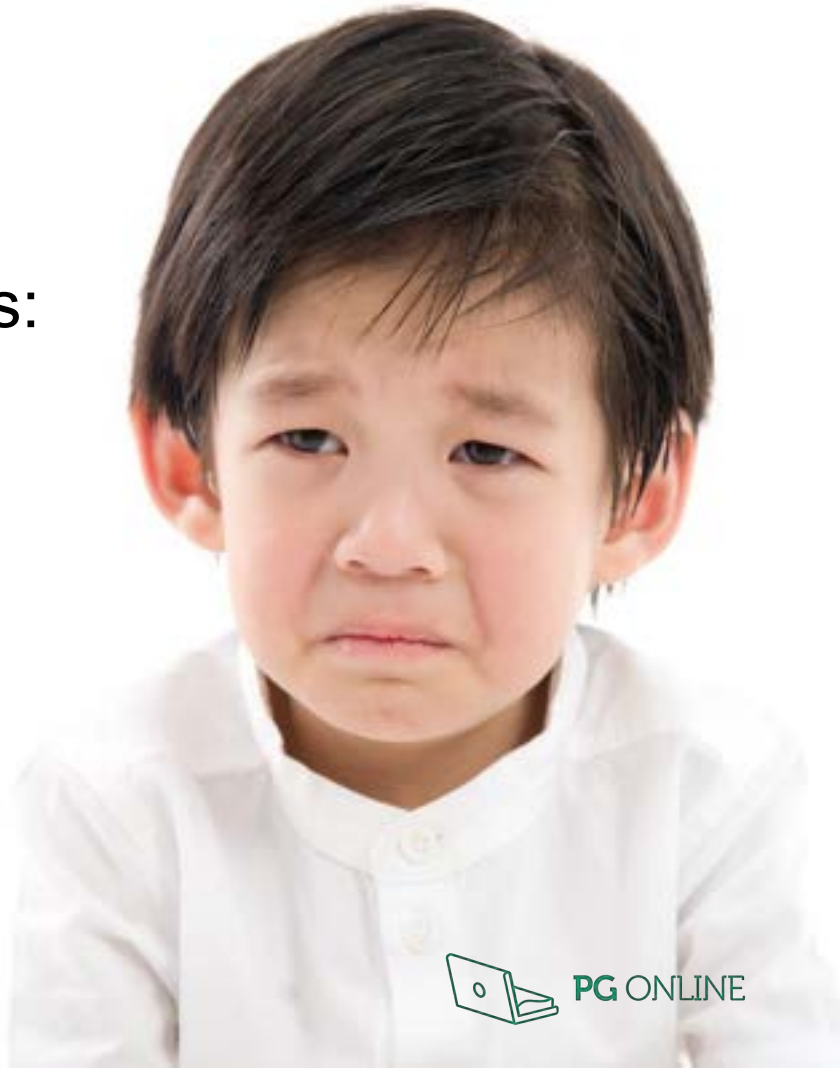
The healthcare robot

- Mabu, the healthcare robot, was launched in 2018
- Mabu sits on the owner's bedside table as a 'personal healthcare companion'
  - It is intended for patients managing chronic diseases
  - The robot talks to patients and reminds them to take their medication

PG ONLINE

Robots as carers

- In Japan and South Korea, childminding robots are being developed
- Scientists warn of the dangers:
  - *"We already see the overuse of robots looking after children – we expect severe attachment disorders that could wreak havoc in our society"* – Noel Sharkey, professor of AI



PG ONLINE

Worksheet 1

- Complete **Task 2** on **Worksheet 1**

Digital divide

- This is the difference between those who have and don’t have access to technology, and includes:
  - Internet connection speed
  - Mobile phone signal
  - 4G signal for mobile Internet
- There is also a divide between other groups of people, such as:
  - Elderly
  - People in poverty

Case study: Rural internet

- In December 2019, Kingsclere in Huntington (York) averaged 0.14 Mb/s
  - Darwin Street in Livingston averaged 183 Mb/s
- Often places in the countryside only have access to Internet slower than 1 Mb/s
  - What information and services will they struggle to access?
  - How would this affect someone starting a business?

PG ONLINE

Case study

Answers

- What information and services will they struggle to access?
  - Video services such as iPlayer, Netflix or YouTube
  - Communication tools such as Facetime or WhatsApp
  - Video games and cloud services
- How would this affect someone starting a business?
  - They may find it harder to set up a website
  - Sharing photos or videos of products will be harder
  - It may be slower to respond to customer queries
  - They may not have access to newer cloud services

Remote working

- In 2020, more than two-thirds of UK companies allow staff to work remotely
- 70% of 18-34 year olds already regularly work remotely
  - What are the benefits of remote working?
  - What are some disadvantages?

PG ONLINE

Advantages

Answers

- Some advantages of remote working include:
  - Time and money saved commuting
  - Less commuting can be better for the environment
  - Saves the company the cost of building more office space
  - Companies have a wider pool of people to choose from, and employees don’t need to relocate
  - Work-life balance can be improved
  - Allows virtual meetings to take place where face-to-face conversation is not possible

Disadvantages

Answers

- Some disadvantages of remote working include:
  - Remote workers can feel isolated
  - There is less face to face contact, which can make it harder to build relationships with colleagues or participate in meetings
  - Difficult for companies to monitor employee performance
  - It can be difficult to stop focusing on work, which disrupts work-life balance
  - It can be hard to avoid distractions when working at home from children, spouses, pets and doorbells

Worksheet 1

- Complete **Task 3** on **Worksheet 1**

Plenary

- Social networks are able to remove users who post or say things that go against their policies
- They can do this, even if the user hasn't broken any laws
  - Give **two** reasons why this can be considered ethical
  - Give **two** reasons why this can be considered unethical

PG ONLINE

Plenary

Answers

- Some suggestions in support of it being ethical:
  - The companies are private so should be able to decide who uses their service
  - Removing certain users can make the environment safer or more pleasant for other users
  - Anti-social people can be removed from public spaces in real life – e.g. with ASBOs
- Some suggestions that it is unethical:
  - A few people can control the views of billions of others
  - With very few social media channels, removal from just a few denies people any ability to participate in social media
  - It denies freedom of speech

PG ONLINE

Copyright

© 2024 PG Online Limited

The contents of this unit are protected by copyright.

This unit and all the worksheets, PowerPoint presentations, teaching guides and other associated files distributed with it are supplied to you by PG Online Limited under licence and may be used and copied by you only in accordance with the terms of the licence. Except as expressly permitted by the licence, no part of the materials distributed with this unit may be used, reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic or otherwise, without the prior written permission of PG Online Limited.

Licence agreement

This is a legal agreement between you, the end user, and PG Online Limited. This unit and all the worksheets, PowerPoint presentations, teaching guides and other associated files distributed with it is licensed, not sold, to you by PG Online Limited for use under the terms of the licence.

The materials distributed with this unit may be freely copied and used by members of a single institution on a single site only. You are not permitted to share in any way any of the materials or part of the materials with any third party, including users on another site or individuals who are members of a separate institution. You acknowledge that the materials must remain with you, the licencing institution, and no part of the materials may be transferred to another institution. You also agree not to procure, authorise, encourage, facilitate or enable any third party to reproduce these materials in whole or in part without the prior permission of PG Online Limited.

OCR
Oxford Cambridge and RSA
An OCR endorsed teaching and learning tool

GCSE OCR

Computer Science
J277

Environmental issues

Unit 5
Impacts of digital technology

PG ONLINE

2

Objectives

- Discuss the impacts of digital technology on the environment including:
  - The impact of manufacture and disposal
  - The impact of upgrading or replacing
  - The impact of e-waste

Starter

- Name each of the computer components below
  - What raw materials are they each made from?

Starter

Answers

Hard disk
- Aluminium
- Steel
- Fibreglass
- Plastic

Gaming controller
- Plastic
- Rubber
- Lithium

USB cable
- Copper
- Plastic

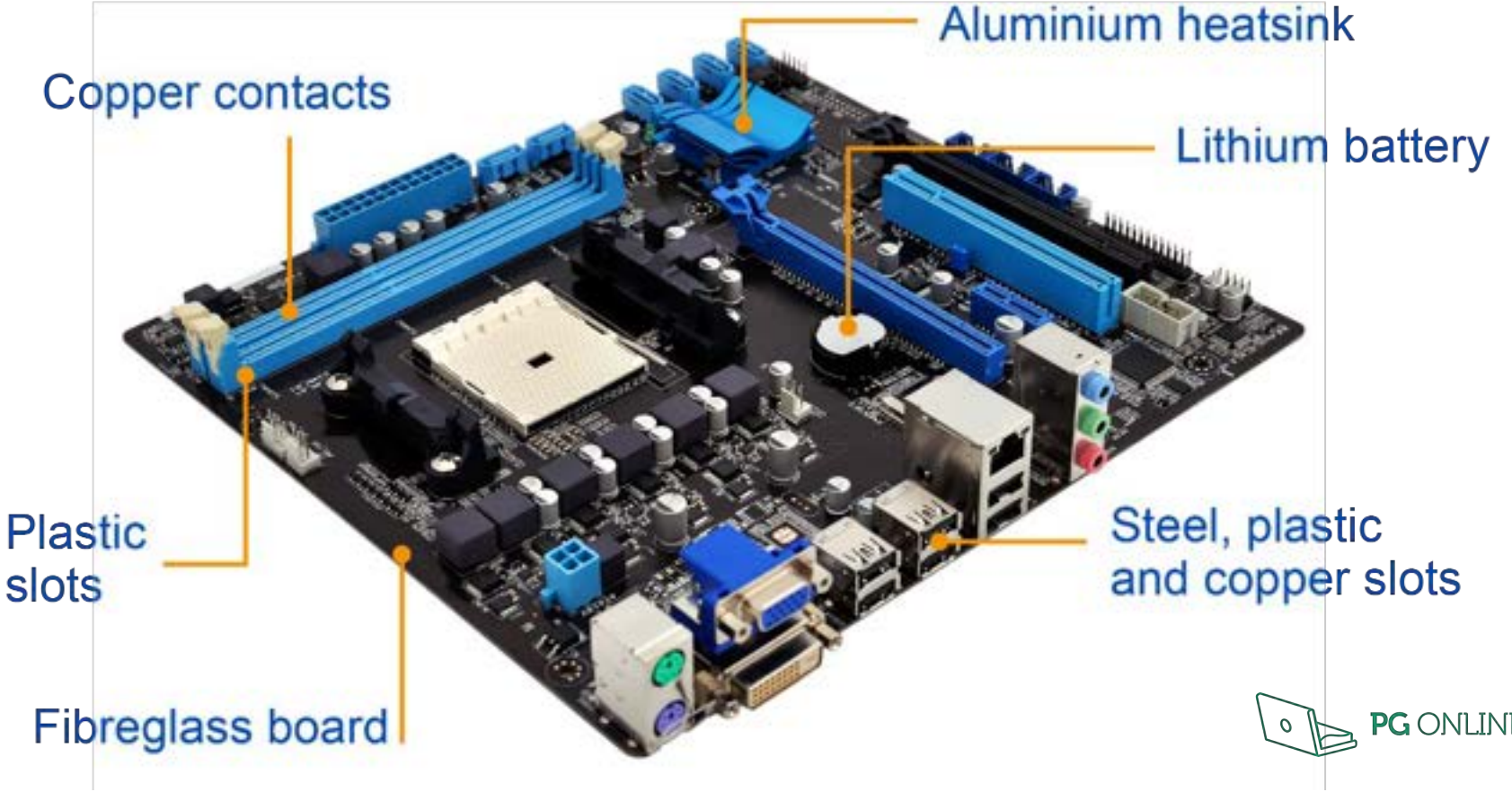
CPU
- Fibreglass
- Aluminium
- Gold

Heat sink
- Aluminium

RAM
- Fibreglass
- Gold

Materials in one component

- Which computer component is shown here?
  - What is each socket or slot used for?

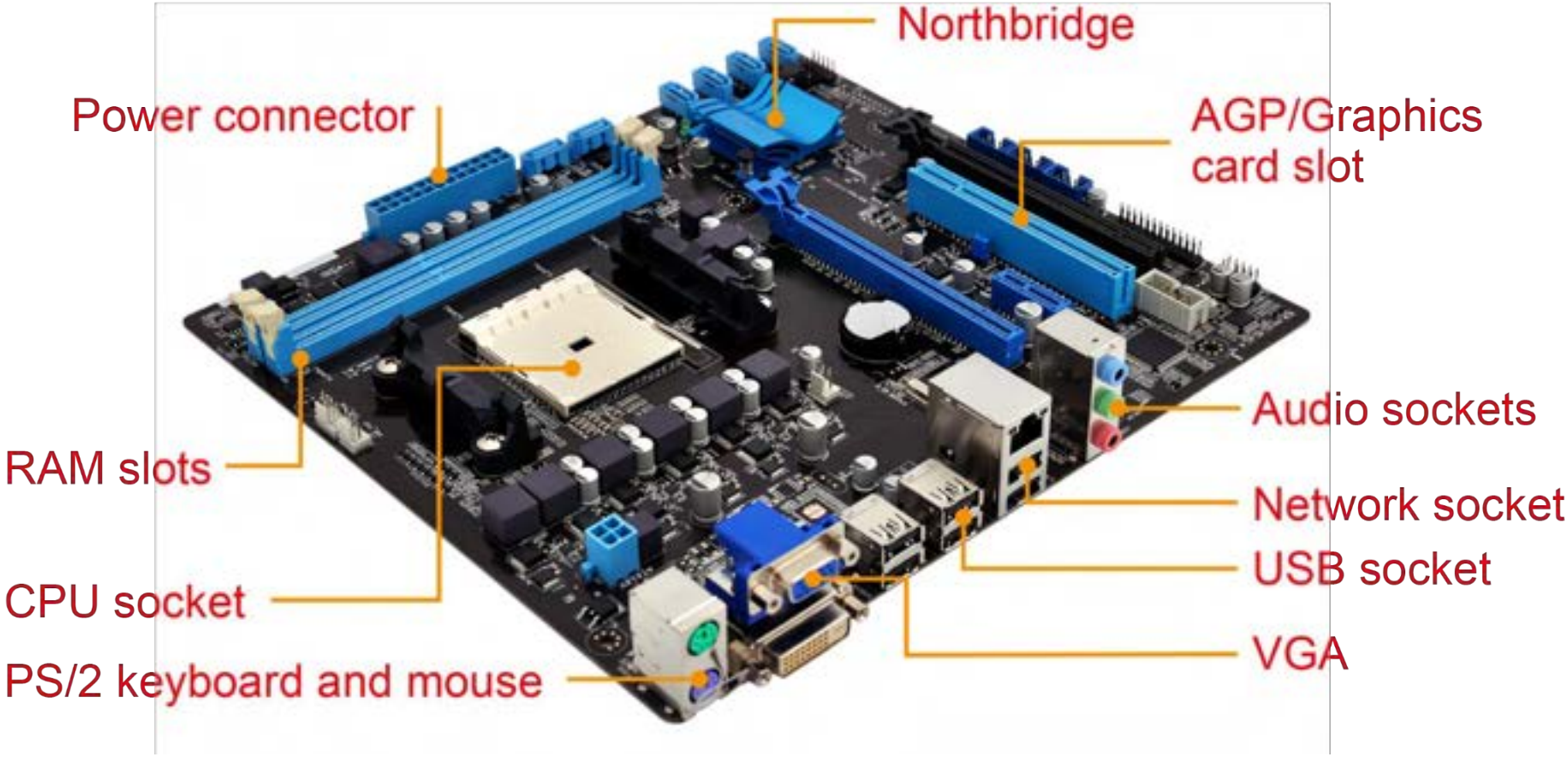


PG ONLINE

Materials in one component

Answers

- Computer socket uses



Worksheet 2

- Complete **Task 1** on **Worksheet 2**

Lifecycle of a smartphone

- Where do the computers or electronic devices come from before you buy them?
- What journey does a smartphone make when it reaches the end of its life?

Lifecycle of a smartphone

Answers

1. Mining for raw materials
2. Manufacture
3. Purchase and use
4. Recycling centre
5. Reprocessing plant

PG ONLINE

Environmental damage

- Mining raw materials leads to contamination and erosion
  - Plastics damage the environment if not recycled
- Gas and coal are needed to power factories
  - Diesel is needed to transport raw materials, parts and the final product

PG ONLINE

Renewable vs Non-renewable

- Are each of the following materials, used to make electronics, renewable or non-renewable?
  - Plastic
  - Steel
  - Gold
  - Copper
  - Aluminium
  - Fibreglass
  - Diesel
  - Water

PG ONLINE

Type of material

Answers

- Plastic Non-renewable
- Steel Non-renewable
- Gold Non-renewable
- Copper Non-renewable
- Aluminium Non-renewable
- Fiberglass Non-renewable
- Diesel Non-renewable
- Water Renewable

Difficulties in recycling

- Watch the video on the explosive problem with recycling old electronics [5m06s]
  - What problems can lithium batteries cause when recycling?
  - Why is sealing batteries into devices a problem for the environment?

E-waste

- The large scrap metal yard in Agbogbloshie, Ghana processes electronic waste
  - How are raw materials extracted?
  - How are people and the environment affected by e-waste?
  - How is e-waste recycling improving?

Worksheet 2

- Complete **Task 2** on **Worksheet 2**

What we throw away each year

- In the UK:
  - 1.6 million tonnes of e-waste is generated each year
  - 23% of electronics thrown away still work or need minor repair
  - Each mobile phone contains about 24mg of gold
- Worldwide we throw away:
  - 300 tons of gold (£9 billion)
  - 1000 tons of silver (£400 million)
  - 16 million tons of steel (£6.5 billion)

PG ONLINE

Case Study: Fairphone

- Fairphone was launched in 2013
- The modular phone is designed so that the user can replace the parts in it
- This means the phone lasts longer and creates less waste
  - What do you think the parts shown below do?

PG ONLINE

Case Study: Fairphone

Answers

- Changeable parts for the phone

Battery

Vibration mechanism

Rear camera

Fairphone

- Designed for easy use and to be long lasting
  - Offers repair tutorials to increase how long the phone can work for
- Responsibly sourced materials
- Good working conditions for people who make the phone
  - Why may it be hard to sell large numbers of this type of phone?

Worksheet 2

- Complete **Task 3** on **Worksheet 2**

Plenary

- What are **three** impacts of e-waste on the environment?
  - What are **two** ways that these impacts could be reduced?

Plenary

- Impacts of e-waste on the environment:
  - Very harmful to humans and the environment if not processed correctly
  - Lithium batteries can catch fire and may be hard to recover
  - If components cost too much to recover they go to landfill
  - Mining materials damages the environment
- Ways to reduce the impact:
  - Make devices that can be fixed with modular components
  - Use removable batteries
  - Use modern recycling facilities

Copyright

© 2024 PG Online Limited

The contents of this unit are protected by copyright.

This unit and all the worksheets, PowerPoint presentations, teaching guides and other associated files distributed with it are supplied to you by PG Online Limited under licence and may be used and copied by you only in accordance with the terms of the licence. Except as expressly permitted by the licence, no part of the materials distributed with this unit may be used, reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic or otherwise, without the prior written permission of PG Online Limited.

Licence agreement

This is a legal agreement between you, the end user, and PG Online Limited. This unit and all the worksheets, PowerPoint presentations, teaching guides and other associated files distributed with it is licensed, not sold, to you by PG Online Limited for use under the terms of the licence.

The materials distributed with this unit may be freely copied and used by members of a single institution on a single site only. You are not permitted to share in any way any of the materials or part of the materials with any third party, including users on another site or individuals who are members of a separate institution. You acknowledge that the materials must remain with you, the licencing institution, and no part of the materials may be transferred to another institution. You also agree not to procure, authorise, encourage, facilitate or enable any third party to reproduce these materials in whole or in part without the prior permission of PG Online Limited.

OCR
Oxford Cambridge and RSA
An OCR endorsed teaching and learning tool

GCSE OCR

Computer Science
J277

Legislation and privacy

Unit 5
Impacts of digital technology

PG ONLINE

3

Objectives

- Discuss the impacts of digital technology on wider society including:
  - Legal issues
  - Privacy issues
- Describe legislation relevant to computer science:
  - The Data Protection Act 2018
  - Computer Misuse Act 1990
  - Copyright Designs and Patents Act 1988
  - Software licences including open source and proprietary

Starter

- An employee of a web design company loses their job and is upset
  - Before they leave, they log in to a computer using a colleague's username and password
  - They then copy graphics from the company such as logos, photos and web layouts for their own use
  - They also copy a list of all the company's clients and their contact details
- Which laws has the employee broken?

Starter

Answers

- Laws broken:
  - Logging in without permission – Computer Misuse Act (1990)
  - Copying the graphics without permission – Copyright, Designs and Patents Act (1988)
  - Copying personal details without their permission – Data Protection Act (2018)

Data protection

- Data protection is about looking after the personal data of people
- The Data Protection Act (2018) is the law that covers this
  - In 2018, the EU General Data Protection Regulation (GDPR) was incorporated into a new UK law called the Data Protection Act (2018)
- Give **two** ways that personal data could be misused

PG ONLINE

Data collection

- Organisations that collect personal data must:
  - Only collect the data for a specific purpose
  - Make sure the data is accurate
  - Data that is not necessary for the specific purpose may not be collected
- Data can come from user sign-up details
  - What are **three** other ways that organisations can collect data?

Data collection sources

- There are many ways that companies can collect data, including:
  - 3rd parties – other companies can sell or share the data
  - Cookies track users as they browse websites
  - Paper registration forms
  - CCTV
  - Viewing habits with streaming services

Data processing

- There are six reasons for lawful processing
  - Consent
  - Legal obligation
  - Public task
  - Contract
  - Vital interests
  - Legitimate interests
- What do each of the above reasons mean?

PG ONLINE

Data processing

Answers

- Lawful processing reasons:

Lawful reason	Meaning	Example
Consent	A person has agreed to their data being used	A tick box to consent to receive a newsletter
Contract	Processing is needed for a contract	A contract for buying a house
Legal obligation	Processing the data is needed to meet the law	Keeping tax records
Vital interests	Processing is needed to protect someone's life	A teacher gives a collapsed student's name to a paramedic
Public task	For performing an official task	A criminal court
Legitimate interests	There is a clear benefit to the user or company	Processing customer data

Data storage

- Data must be stored so that:
  - The data is kept accurate and up-to-date
  - It is not kept any longer than necessary
  - It must not be transferred to other countries unless they can keep it protected
  - Customers must be told of a data breach within 72 hours of it happening
- Data must be stored so that it is protected from unauthorised access
  - Give **two** ways this can be achieved

PG ONLINE

Methods of securing data

Answers

- Personal data needs to be carefully secured using reasonable methods including:
  - Using passwords for any systems with access to the data
  - Encrypting the data
  - Only allowing access to those users that need it
- If the data is more sensitive then the following may be used
  - CCTV
  - Security guards
  - Two-factor authentication

Rights

- The Data Protection Act (2018) gives a number of rights:
  - The right to view data stored about you by organisations for free – previously it cost up to £10
  - You must consent to having marketing sent to you – this consent must be 'opt-in'
  - The right to withdraw consent – mailing lists have an unsubscribe link for this
  - The right to make changes to your data if it is inaccurate
  - The right to be forgotten – allows you to delete your personal data

PG ONLINE

Penalties

- Penalties from the Data Protection Act include:
  - Issuing warnings to the organisation
  - Order the organisation to comply
- For serious breaches, fines are up to:
  - 4% of company turnover
  - €20 million
- Shell is the largest UK company earning over £298 billion per year
  - How big could their fine be?

Privacy issues

- Every time you log on to a web site, data about your visit may be collected and stored
- Every time you use your phone to make a call, data about the number called, time and date called, and the duration of the call, is collected and stored
  - Why might people have concerns about this?

PG ONLINE

Privacy online

- Many online services such as search engines and social media are free to use
- These services are paid for by targeted advertising
  - How does a social media site know personal information about their users that they can target specific people with advertising messages?
  - If you visit a website that sells a holiday then many other websites will now show you adverts for holidays. What technology makes this possible?

Cookies

- Cookies are sent to a user's computer from web sites. The allow websites to:
  - Store data such as the contents of your shopping basket
  - Remember that you are logged into a website
  - Remember who you are
  - Track you
  - Target advertising to you
- Users must consent to receiving cookies
  - What are your views of this?

Worksheet 3

- Now complete **Task 1** on **Worksheet 3**

Computer Misuse Act (1990)

- The Computer Misuse Act (1990) created the following offences:
  - Unauthorised access to computer material
  - Unauthorised access with intent to commit or facilitate a crime
  - Unauthorised modification of software or data
  - Making, supplying or obtaining anything which can be used in computer misuse offences
- Committing any of these crimes can result in up to 10 years in prison and a fine
  - What does unauthorised access and unauthorised modification mean?

Unauthorised access

- Unauthorised access is where a person gains access to a computer system without permission
  - This could be a hacker who gains permission to a system they shouldn't be using
  - It may also be an employee who has permission to use some of a system, but gains access to another part
- Hackers try to find weaknesses in a computer system
  - They will often gain access with one user account and then try to alter the permissions to get root or admin access

Unauthorised modification

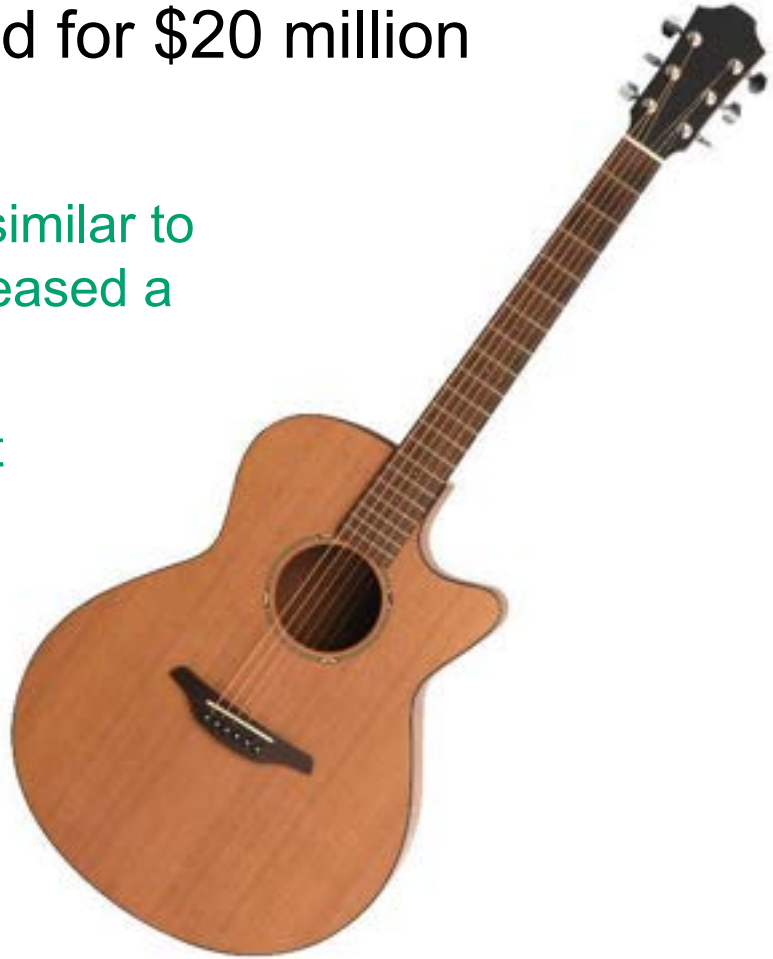
- Unauthorised modification includes:
  - Deleting another user's files
  - Changing the content of documents
  - Altering the content of web pages
  - Rewriting computer programs to remove activation keys
- What is **one** other unauthorised modification that can be made?

Copyright and patents

- Inventions are covered by patents
  - A patent prevents anyone else using the invention for 20 years
- Copyright protects books, video, music and software
  - Copyright typically lasts 70 years after publication or an author's death
  - Copyright lets an author or musician decide how their work should be used
- Copyrighted materials and patents are protected by the Copyright, Designs and Patents Act (1988)
  - Why do you think that copyright lasts longer than patents?

Case study: Ed Sheeran

- In 2017, Ed Sheeran was sued for $20 million for copyright infringement
  - His song ‘Photograph’ was very similar to Matt Cardle’s song ‘Amazing’ released a few years earlier
  - The case was settled out of court with Ed Sheeran paying £16 million



Penalties

- Copying books, webpages, music, video or software is known as copyright infringement
- Copyright or trademark infringement can result in:
  - Fines
  - Up to 10 years in prison
- What methods are used to protect software from being illegally copied?

Preventing copyright

Answers

- The following are some of the ways that software is made harder to copy without permission
  - Licence keys, activation keys and serial numbers
  - Holograms on physical products to show they are genuine
  - Online registration or activation will prevent the software from working if a licence has not been purchased
- Many companies are moving to software in the cloud
  - This is very hard to copy as the software is accessible only within the companies servers
  - The user views the output of the software through their web browser

Worksheet 3

- Now complete **Task 2** on **Worksheet 3**

Software licences

- Software is protected by copyright law
  - It may only be used if you created it yourself or you are licensed to use it
- For proprietary software a licence must be purchased before the software can be used
- For open source software the licence normally permits the software to be downloaded for free – the licence often has conditions such as requiring you not to sell it
  - What does proprietary and open source software mean?

Proprietary software

- Proprietary software is the most commonly used software by the general public today
  - The source code is usually kept by the developer (closed-source) and the user receives the compiled programs
  - The software cannot be adapted or modified by the user
  - The software can only be used if the user has a licence (which is usually paid for)
  - This type of software is typically off-the-shelf and not custom made

Open source software

- Open source software provides access to the source code that was used to create it
  - By having the source code, users have the freedom to develop or modify it
- Open source software is often developed collaboratively by many programmers and companies that give up their time to improve the software
  - Name **three** different examples of open source software

Worksheet 3

- Now complete **Task 3** on **Worksheet 3**

Plenary

- Jenny works in human resources for Bell & Smith
- She arrived at work this morning at 9 am
  - How is it possible for her to break three laws before 9:15 am using only her computer?

PG ONLINE

Plenary

Answers

- Three possible ways that Jenny could have broken the laws include:
  - Logging onto her computer with a colleague's username and password without their permission – Computer Misuse Act (1990)
  - Copying an image from the web for a company brochure without paying for the required licence – Copyright, Designs and Patents Act (1988)
  - Emailing another company the names and contact details of all employees at her company without their permission – Data Protection Act (2018)

Copyright

© 2024 PG Online Limited

The contents of this unit are protected by copyright.

This unit and all the worksheets, PowerPoint presentations, teaching guides and other associated files distributed with it are supplied to you by PG Online Limited under licence and may be used and copied by you only in accordance with the terms of the licence. Except as expressly permitted by the licence, no part of the materials distributed with this unit may be used, reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic or otherwise, without the prior written permission of PG Online Limited.

Licence agreement

This is a legal agreement between you, the end user, and PG Online Limited. This unit and all the worksheets, PowerPoint presentations, teaching guides and other associated files distributed with it is licensed, not sold, to you by PG Online Limited for use under the terms of the licence.

The materials distributed with this unit may be freely copied and used by members of a single institution on a single site only. You are not permitted to share in any way any of the materials or part of the materials with any third party, including users on another site or individuals who are members of a separate institution. You acknowledge that the materials must remain with you, the licencing institution, and no part of the materials may be transferred to another institution. You also agree not to procure, authorise, encourage, facilitate or enable any third party to reproduce these materials in whole or in part without the prior permission of PG Online Limited.

OCR
Oxford Cambridge and RSA
An OCR endorsed teaching and learning tool

GCSE OCR

Computer Science
J277

1

Computational thinking

Unit 6
Algorithms

PG ONLINE

Objectives

- Understand the principles of computational thinking including
  - Abstraction
  - Decomposition
  - Algorithmic thinking
- Be able to produce structure diagrams to show:
  - The structure of a problem
  - Subsections and their links to other subsections

Starter

- Computer science is about studying problems and how to solve them
- Discuss some solutions to the following:
  - How can you route pieces of information across a network to the other side of the world?
  - How can you make the images in a computer game look more realistic?
  - How can you program a computer to work out the school timetable?
  - How can you search 1,000,000 items quickly?

Starter

Answers

- The answers to these problems are too long to write here – some aspects that you may have considered in your discussions
  - Some problems such as ‘how can you search 1,000,000 items quickly?’ require a specific **algorithm** to be considered
  - There may be many different algorithms that solve the same problem
  - Problems like ‘How can you make the images in a computer game look more realistic?’ need **abstractions**, where we consider the important components of a realistic image
  - Routing items of data around the world requires many different protocols. This is a result of **decomposing** the problem into smaller tasks

What is an algorithm?

Algorithms

- An algorithm is a set of instructions for solving a problem or completing a task
- The task could be:
  - Making a chocolate cake
  - Summing the numbers 1 to 1000
  - Building a Lego model
  - Think of some more …

Strategies for problem-solving

- One strategy for solving a large problem is to first try and solve a similar but smaller problem
  - How do you set about doing a jigsaw puzzle?

Algorithmic thinking

- Solving these puzzles involves algorithmic thinking
- If you are using a computer to find the answer, you have to figure out how to solve the problem, and then write down the steps
  - Not all solutions are equally efficient

Divide and conquer

- Here is a problem
- Ask a friend to think of a number between 1 and 1000
- Guess the number by asking: "Is the number greater than ***n?***" (where ***n*** is your guess)
  - If ***n*** is the correct number, then stop the game
  - How many guesses will you need to find the number?

PG ONLINE

Worst case scenario

Answers

- Search for the number at the mid-point (500)
  - If it's the number we are searching for then stop
  - If it's lower than 500, then search at the mid point of the lower numbers (250)
  - Otherwise, if it's higher than 500, then search at the mid point of the higher numbers (750)
- The worst case scenario for this algorithm would be a search for numbers such as '1' or '367'
  - With this algorithm, any number can be guessed within 10 guesses
  - 1 = 500,250,125,63,32,16,8,4,2,1
  - 367 = 500,250,375,312,344,360,368,364,366,367

Worksheet 1

- Now complete **Task 1** on **Worksheet 1**

Abstraction

- Abstraction involves removing unnecessary detail from a problem so that you can focus on the essential components
  - The London Underground map is a good example of abstraction

Abstraction

- When you write a program to play a game involving dice with a computer, how does the computer ‘roll the dice’?

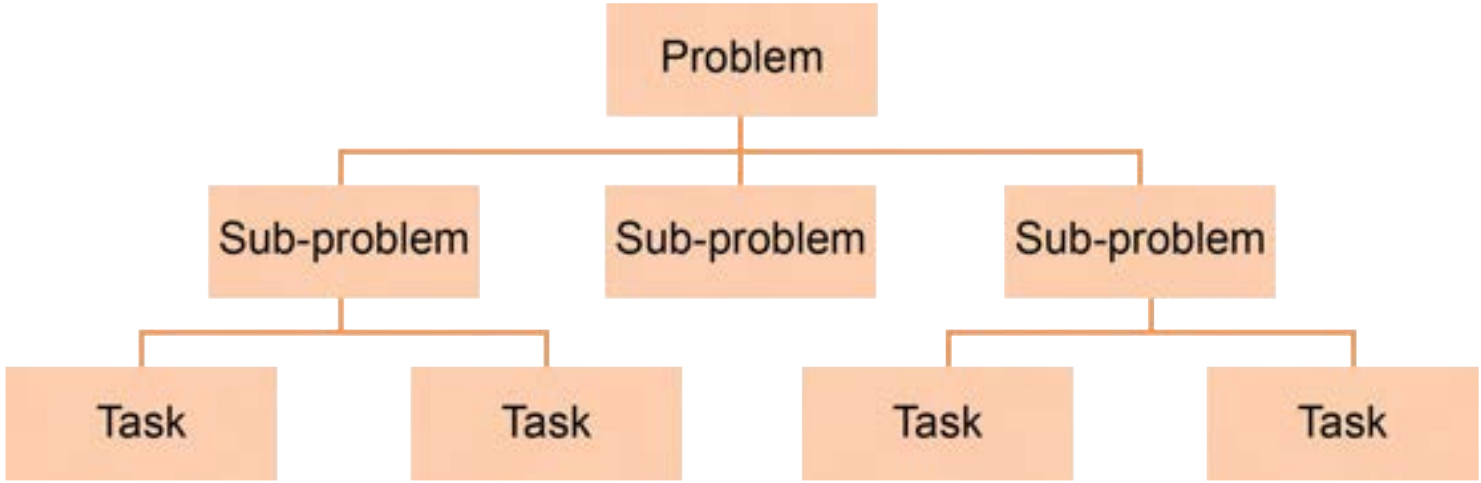
Rolling dice

Answers

- When a computer rolls a dice we can use abstraction to remove many unnecessary details
  - It will depend on the problem being solved as to what is and isn't important
  - A computer game may need to show a graphical representation of a dice – but they may be able to abstract away all the details about the surface it rolls onto and the physics of the bounce
  - Many programs just need a random number – in which case they don't need to worry about how the dice appears, its weight or how the spots are arranged – they can just find a random number with one line of programming code

Decomposition

- Decomposition involves breaking down a large problem into smaller sub-problems
- Then the sub-problems can be broken down further until each small task is manageable

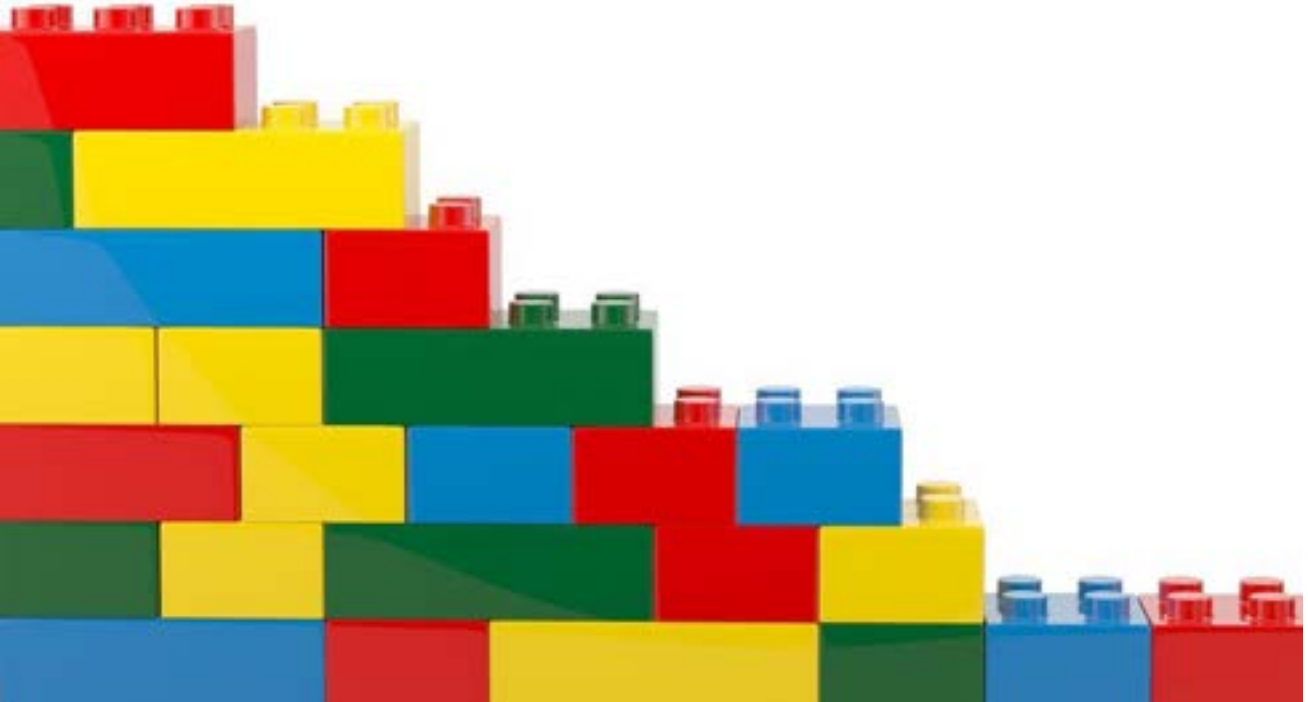


Decomposing a dice game

- Suppose you want to create a dice game to be played on the computer
- You need to think of the main tasks that need to be performed – for example:
  - Display the rules
  - Computer AI for a two player game
  - Display the board
  - Play the game
  - A results table

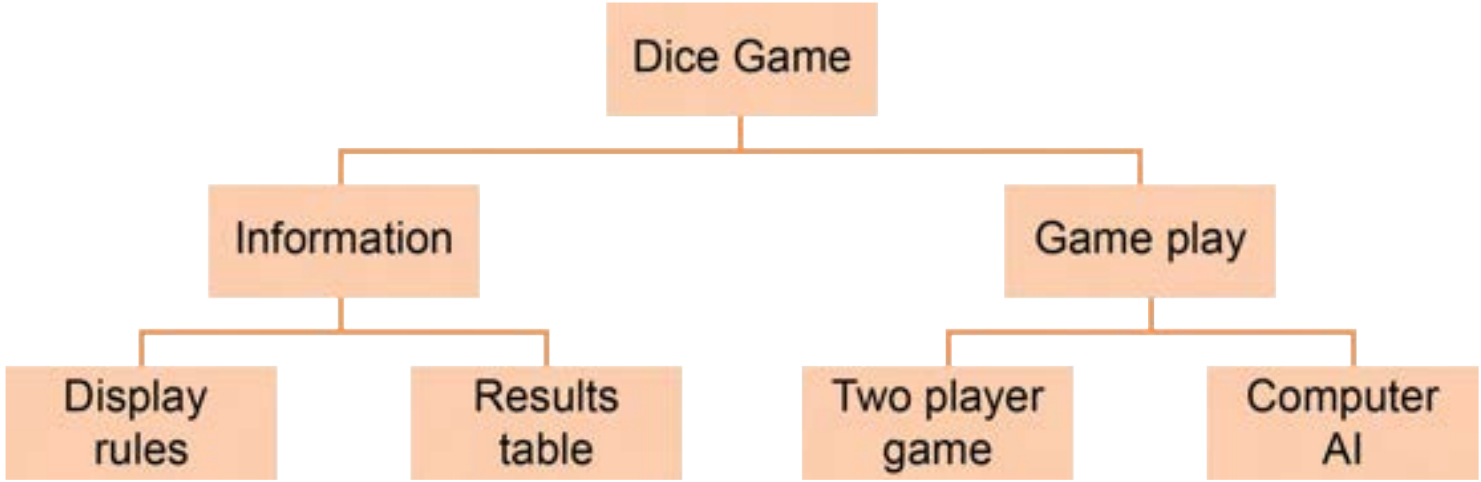
Decomposition - advantages

- The problem becomes easier to solve when it consists of a number of small subtasks or modules
- Some modules may be reusable in other programs, saving development time



Structure diagrams

- A structure diagram is used to show how a problem is broken down
  - It will show subsections and their links to other subsections



Worksheet 1

- Now complete **Task 2** on **Worksheet 1**

Plenary

- With a partner define the following terms:
  - Abstraction
  - Decomposition
  - Algorithmic thinking
  - Structure diagram

Plenary

Answers

- Definitions
  - Abstraction – removing unimportant parts of a problem in order to concentrate on those that are important
  - Decomposition – breaking down a problem into smaller more manageable ones
  - Algorithmic thinking – an approach to solving problems by the use of algorithms (sequences of steps that lead to a solution)
  - Structure diagram – a hierarchical diagram that shows how a problem is broken down into sub-sections/sub-tasks

Copyright

© 2024 PG Online Limited

The contents of this unit are protected by copyright.

This unit and all the worksheets, PowerPoint presentations, teaching guides and other associated files distributed with it are supplied to you by PG Online Limited under licence and may be used and copied by you only in accordance with the terms of the licence. Except as expressly permitted by the licence, no part of the materials distributed with this unit may be used, reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic or otherwise, without the prior written permission of PG Online Limited.

Licence agreement

This is a legal agreement between you, the end user, and PG Online Limited. This unit and all the worksheets, PowerPoint presentations, teaching guides and other associated files distributed with it is licensed, not sold, to you by PG Online Limited for use under the terms of the licence.

The materials distributed with this unit may be freely copied and used by members of a single institution on a single site only. You are not permitted to share in any way any of the materials or part of the materials with any third party, including users on another site or individuals who are members of a separate institution. You acknowledge that the materials must remain with you, the licencing institution, and no part of the materials may be transferred to another institution. You also agree not to procure, authorise, encourage, facilitate or enable any third party to reproduce these materials in whole or in part without the prior permission of PG Online Limited.

OCR
Oxford Cambridge and RSA
An OCR endorsed teaching and learning tool

GCSE OCR

Computer Science
J277

2

Searching algorithms

Unit 6
Algorithms

PG ONLINE

Objectives

- Understand and use different types of search
  - Binary search
  - Linear search

Starter

- We search for items many times a day
- Physical objects such as:
  - Clothes
  - Homework
- What are **five** items that we search for on computers or the Internet?

PG ONLINE

Searching a list

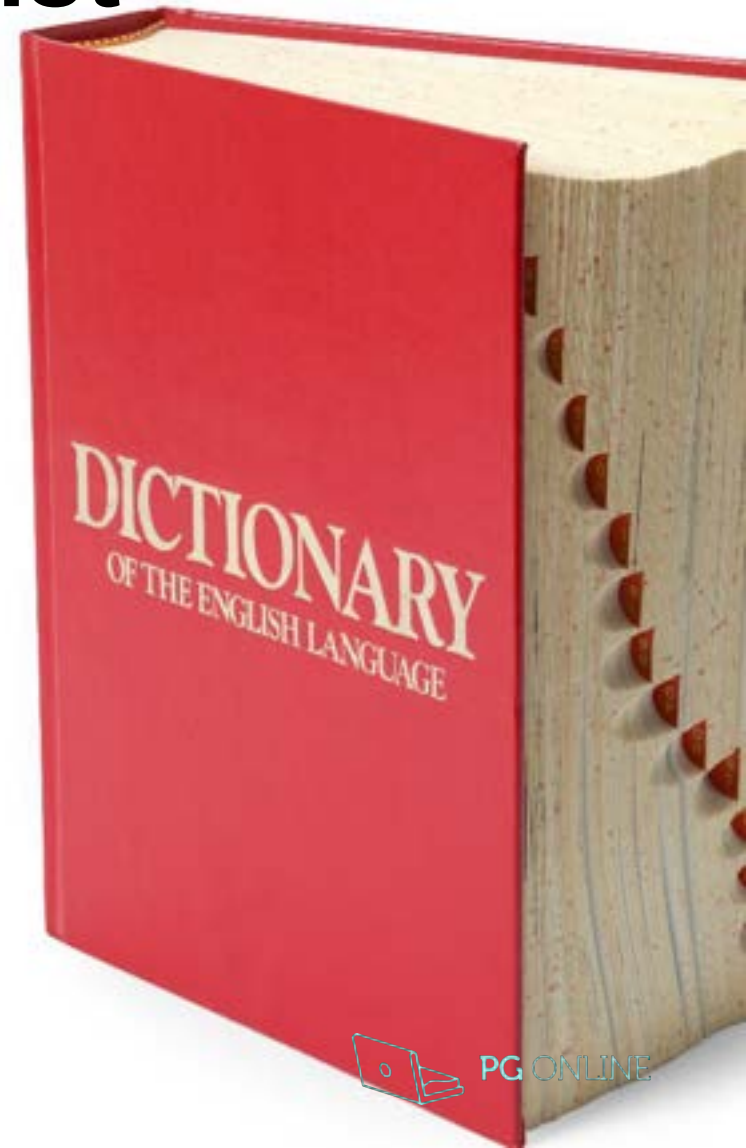
Answers

- Searching is a very common operation in computing
  - A doctor might search for a patient's notes
  - A police officer can search with a vehicle identification number (VIN) to find the owner of an abandoned car
  - A web browser can search for a word in a web page
  - There are numerous other examples of searches

PG ONLINE

Searching a sorted list

- Suppose you have a dictionary and you want to look up the word nebula
- What will be your strategy for finding the word?
  - One way is to start at the first word and go through every word in the dictionary one by one
  - What other methods could you use?



PG ONLINE

A linear search

- A linear search requires going through each item in the list, one by one

apple	**banana**	**dog**	**egg**

- To find ‘dog’ look at each word in the list:
  - Is "dog" == "apple"? **No**
  - Is "dog" = "banana"? **No**
  - Is "dog" = "dog"? **Yes – item found**
- A linear search will, however, be very slow for searching a complete dictionary

A binary search

- Here is a list of names:

Ali	Ben	Carl	Joe	Ken	Lara	Mo	Oli	Pam	Stan	Tara

A quicker way to find if a particular name is in a sorted list is to do a binary search

- Suppose we are searching for the name Mo
- The list has 11 items
- Examine the middle one first

A binary search

- The middle item in the list is Lara

Ali	Ben	Carl	Joe	Ken	Lara	Mo	Oli	Pam	Stan	Tara

- Lara comes before Mo alphabetically so we can discard all the names from Ali to Lara
- Now we only have five names to search

A binary search

- Here is a list of names:

Ali	Ben	Carl	Joe	Ken	Lara	Mo	Oli	Pam	Stan	Tara

- Examine the middle name of the remaining list
- The middle name is Pam
- Mo comes before Pam so we can discard all the names from Pam to Tara

A binary search

- Here is a list of names:

Ali	Ben	Carl	Joe	Ken	Lara	Mo	Oli	Pam	Stan	Tara

- Now we only have two names
- The ‘middle’ name is taken to be the left of the middle point
- Examine the name, Mo
  - Bingo! How many names did we look at for the binary search?

Divide and conquer

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16

- In a binary search, the size of the list is halved each time an item is examined
- How many items, at most, would have to be examined in a list of 16 items to find the one you are looking for?
- Try looking for the number 23 in this hidden list of numbers
  - Which box will you look at first?

Divide and conquer

							42								
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16

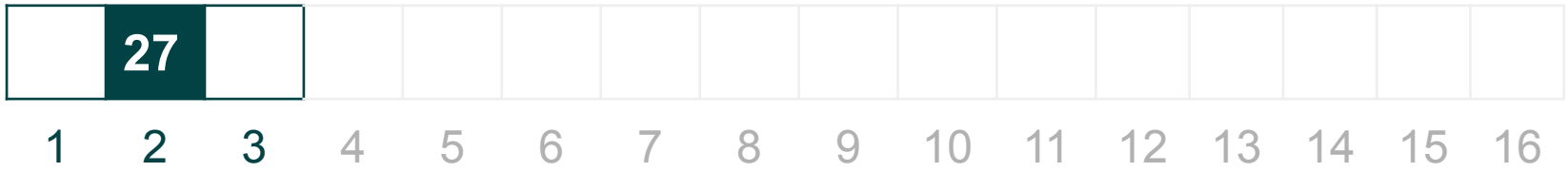
- You're looking for the number 23
- There are 16 boxes, so you first try: 16/2=box 8
- Box 8 has the number 42
- This is bigger than 23
  - Which box will you look at next?

Divide and conquer

			35												
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16

- You try box 4 which contains 35
- You're looking for the number 23
  - Which box will you look at next?

Divide and conquer



- You're looking for the number 23
- You've found the number 27
  - Which box will you look at next?

Divide and conquer

23															
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16

- You've found the number 23!
  - How many numbers did you look at?

Divide and conquer

23	27	32	35	37	38	41	42	45	50	52	53	54	58	61	67
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16

- You looked in boxes 8, 4, 2 and 1
- In a list of 2^n items, the maximum number of items you will need to look at will be $n + 1$
- How many items would be examined if you were searching for 67 instead of 23?
- Try searching for 61 in a list of 15 numbers (delete 67 from the list)
  - How many items need to be examined?

Worksheet 2

- Now complete **Task 1** on **Worksheet 2**

Linear search

- If the list to be searched is not sorted, it is not possible to do a binary search
- A linear search may be carried out instead

145	27	83	777	492	588	91	678	399	123

- Items are examined in the sequence

145, 27, 83, 777….

Comparison of searches

- In a sorted list of 1,000,000 items, how many items will have to be examined to establish that an item is NOT in the list?

PG ONLINE

Comparison of searches

Answers

- With a binary search, only 20 items have to be examined to discover that an item is not in the list
  - That's because 1,000,000 is less than 2^{20} but greater than 2^{19}
  - If the list were unsorted, 1,000,000 items would need to be checked to prove that the item is not in the list
- With a linear search, 1,000,000 items would need to be examined

Worksheet 2

- Now complete **Task 2** on **Worksheet 2**

A search algorithm

```
numbers = [5,1,9,8,7,6,4,10]
searchItem = int(input())
for i=0 to numbers.length - 1
    if numbers[i] == searchItem:
        print("searchItem found")
    endif
next i
```

- The above algorithm asks the user to input a number and then searches for it in the array numbers
  - What is the value of numbers.length – 1 ?
  - When will "searchItem found" be output?
  - Name the type of search algorithm

A search algorithm

Answers

```
numbers = [5,1,9,8,7,6,4,10]
searchItem = int(input())
for i=0 to numbers.length - 1
    if numbers[i] == searchItem:
        print("searchItem found")
    endif
endfor
```

- Answers:
  - What is the value of numbers.length – 1 ?
    7
  - When will "searchItem found" be output?
    Each time it finds a matching item in the array
  - Name the type of search algorithm
    Linear search

Worksheet 2

- Now complete **Task 3** on **Worksheet 2**

Plenary

- With a partner explain how the following searching algorithms work:
  - Binary search
  - Linear search
- Report back your answers to the class

Plenary

Answers

- Binary search
  - This only works on a sorted list
  - The middle item of the list is first checked
  - If the item searched for is less than this item the right of the list is discarded, and a binary search is carried out on the left of the list
- Linear search
  - Each item in the list is checked against the search item in order

Copyright

© 2024 PG Online Limited

The contents of this unit are protected by copyright.

This unit and all the worksheets, PowerPoint presentations, teaching guides and other associated files distributed with it are supplied to you by PG Online Limited under licence and may be used and copied by you only in accordance with the terms of the licence. Except as expressly permitted by the licence, no part of the materials distributed with this unit may be used, reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic or otherwise, without the prior written permission of PG Online Limited.

Licence agreement

This is a legal agreement between you, the end user, and PG Online Limited. This unit and all the worksheets, PowerPoint presentations, teaching guides and other associated files distributed with it is licensed, not sold, to you by PG Online Limited for use under the terms of the licence.

The materials distributed with this unit may be freely copied and used by members of a single institution on a single site only. You are not permitted to share in any way any of the materials or part of the materials with any third party, including users on another site or individuals who are members of a separate institution. You acknowledge that the materials must remain with you, the licencing institution, and no part of the materials may be transferred to another institution. You also agree not to procure, authorise, encourage, facilitate or enable any third party to reproduce these materials in whole or in part without the prior permission of PG Online Limited.

OCR
Oxford Cambridge and RSA
An OCR endorsed teaching and learning tool

GCSE OCR

Computer Science
J277

3

Sorting algorithms

Unit 6
Algorithms

PG ONLINE

Objectives

- Understand the standard sort algorithms:
  - Bubble sort
  - Insertion sort
  - Merge sort
- Be able to apply each algorithm to a data set
- Be able to Identify an algorithm if given the code for it

Starter

- Both people and computers often need data to be sorted
  - Give **five** types of data that people need to be sorted
  - Give **one** reason why a computer would work more efficiently if a list was sorted

PG ONLINE

Starter

Answers

- Types of data that people need sorted
  - Index cards/records of customer details
  - Directories and dictionaries
  - House numbers
  - Library books
  - Stock in a warehouse
  - and many more…
- Reasons why a computer will work more efficiently with sorted lists
  - Computers can use a binary search with sorted lists which is far more efficient than a linear search

Sorting

- Data sets frequently need to be sorted
  - Look at the two data sets of the top 10 boys and girls names in 2019
  - What other orders could they be sorted in?
  - Why might you need to sort the data sets?

Rank	Name	Count
1	Oliver	5 390
2	George	4 960
3	Harry	4 512
4	Noah	4 107
5	Jack	3 988
6	Leo	3 721
7	Arthur	3 644
8	Muhammad	3 507
9	Oscar	3 459
10	Charlie	3 365

Rank	Name	Count
1	Olivia	4 598
2	Amelia	3 941
3	Ava	3 110
4	Isla	3 046
5	Emily	2 676
6	Mia	2 490
7	Isabella	2 369
8	Sophia	2 344
9	Ella	2 326
10	Grace	2 301

Source: ONS (2019)

Sorting

Answers

- The data sets could be sorted as follows:
  - Count order (which will be the same as the rank order)
  - Alphabetical order
  - Reverse alphabetical or count order
- The girls and boys data sets could be combined to show the top 20 names in England and Wales
  - Again these could be sorted by count or name

The bubble sort

- Start with the leftmost position
  - Compare the item in this position with the one next to it
  - If the one next to it is less, swap the items
  - Repeat for all the other positions in the list
  - At the end of one pass through the list, the largest item is at the end of the list
- Repeat the process until the items are sorted
- Suppose you have a list of numbers to be sorted:

9	5	4	15	3	8	11	2

Bubble sort – First pass

9	5	4	15	3	8	11	2
5	9	4	15	3	8	11	2
5	4	9	15	3	8	11	2
5	4	9	15	3	8	11	2
5	4	9	3	15	8	11	2
5	4	9	3	8	15	11	2
5	4	9	3	8	11	15	2
5	4	9	3	8	11	2	15

- Each item is compared with the one on its right, and swapped if it is larger
- At the end of the first pass the largest item bubbles through to the end of the list
- (Mauve indicates sorted items)

First pass Second pass

9	5	4	15	3	8	11	2
5	9	4	15	3	8	11	2
5	4	9	15	3	8	11	2
5	4	9	15	3	8	11	2
5	4	9	3	15	8	11	2
5	4	9	3	8	15	11	2
5	4	9	3	8	11	15	2
5	4	9	3	8	11	2	15

5	4	9	3	8	11	2	15
4	5	9	3	8	11	2	15
4	5	9	3	8	11	2	15
4	5	3	9	8	11	2	15
4	5	3	8	9	11	2	15
4	5	3	8	9	11	2	15
4	5	3	8	9	2	11	15

Second pass

5	4	9	3	8	11	2	15
4	5	9	3	8	11	2	15
4	5	9	3	8	11	2	15
4	5	3	9	8	11	2	15
4	5	3	8	9	11	2	15
4	5	3	8	9	11	2	15
4	5	3	8	9	2	11	15

Third pass

4	5	3	8	9	2	11	15
4	5	3	8	9	2	11	15
4	3	5	8	9	2	11	15
4	3	5	8	9	2	11	15
4	3	5	8	9	2	11	15
4	3	5	8	2	9	11	15

Third pass Fourth pass

4	5	3	8	9	2	11	15
4	5	3	8	9	2	11	15
4	3	5	8	9	2	11	15
4	3	5	8	9	2	11	15
4	3	5	8	9	2	11	15
4	3	5	8	2	9	11	15

3	4	5	8	2	9	11	15
3	4	5	8	2	9	11	15
3	4	5	8	2	9	11	15
3	4	5	8	2	9	11	15
3	4	5	2	8	9	11	15

Fourth pass

3	4	5	8	2	9	11	15
3	4	5	8	2	9	11	15
3	4	5	8	2	9	11	15
3	4	5	8	2	9	11	15
3	4	5	2	8	9	11	15

Fifth pass

3	4	5	2	8	9	11	15
3	4	5	2	8	9	11	15
3	4	5	2	8	9	11	15
3	4	2	5	8	9	11	15

Fifth pass

3	4	5	2	8	9	11	15
3	4	5	2	8	9	11	15
3	4	5	2	8	9	11	15
3	4	2	5	8	9	11	15

Sixth pass

3	4	2	5	8	9	11	15
3	4	2	5	8	9	11	15
3	2	4	5	8	9	11	15

Sixth pass

3	4	2	5	8	9	11	15
3	4	2	5	8	9	11	15
3	2	4	5	8	9	11	15

Seventh pass

3	2	4	5	8	9	11	15
2	3	4	5	8	9	11	15

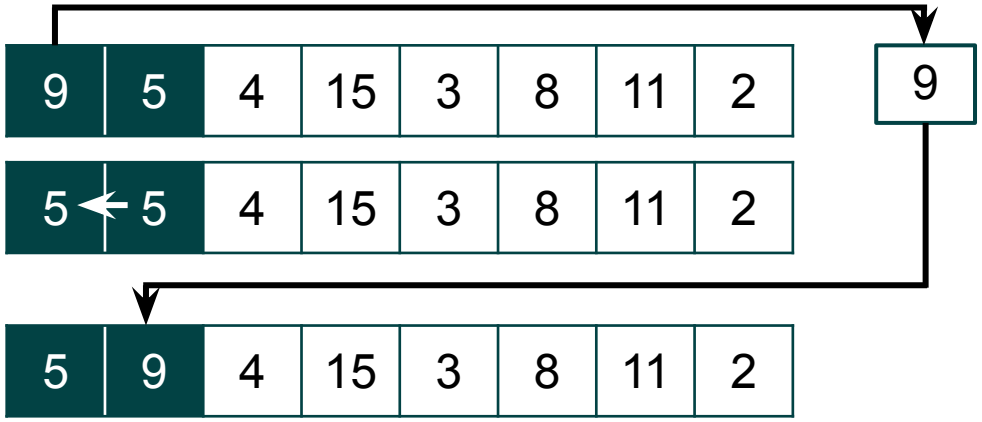
Seven passes through the list of eight numbers ensures that they are sorted

PG ONLINE

Swapping items in a list

9	5	4	15	3	8	11	2

- When performing a bubble sort, we need to swap items – in this case, 9 and 5
  - If we move ‘5’ to where ‘9’ is then we will lose the data
  - Instead we need to use a temporary variable



1. Put 9 in temp variable
2. Copy 5 to where 9 was
3. Copy from the temp variable

Worksheet 3

- Now complete **Task 1** on **Worksheet 3**

Insertion sort

- This algorithm sorts one data item at a time
- It is similar to how you might sort a deck of cards
  - One item is taken from the list, and placed in the correct position
  - This is repeated until there are no more unsorted items in the list

PG ONLINE

Insertion sort

List to be sorted:

9	5	4	15	3	8	11	2

Insertion sort

List to be sorted:

9	5	4	15	3	8	11	2

Leave the first item at the start:

9	5	4	15	3	8	11	2

Insertion sort

List to be sorted:

9	5	4	15	3	8	11	2

Leave the first item at the start:

9	5	4	15	3	8	11	2

5 is now inserted into the sorted list:

5	9	4	15	3	8	11	2

Insertion sort

List to be sorted:

9	5	4	15	3	8	11	2

Leave the first item at the start:

9	5	4	15	3	8	11	2

5 is now inserted into the sorted list:

5	9	4	15	3	8	11	2

4 is now inserted into the sorted list:

4	5	9	15	3	8	11	2

Insertion sort

List to be sorted:	9	5	4	15	3	8	11	2
Leave the first item at the start:	9	5	4	15	3	8	11	2
5 is now inserted into the sorted list:	5	9	4	15	3	8	11	2
4 is now inserted into the sorted list:	4	5	9	15	3	8	11	2
15 is now inserted into the sorted list:	4	5	9	15	3	8	11	2

Insertion sort

List to be sorted:	9	5	4	15	3	8	11	2
Leave the first item at the start:	9	5	4	15	3	8	11	2
5 is now inserted into the sorted list:	5	9	4	15	3	8	11	2
4 is now inserted into the sorted list:	4	5	9	15	3	8	11	2
15 is now inserted into the sorted list:	4	5	9	15	3	8	11	2
3 is now inserted into the sorted list:	3	4	5	9	15	8	11	2

Insertion sort

List to be sorted:	9	5	4	15	3	8	11	2
Leave the first item at the start:	9	5	4	15	3	8	11	2
5 is now inserted into the sorted list:	5	9	4	15	3	8	11	2
4 is now inserted into the sorted list:	4	5	9	15	3	8	11	2
15 is now inserted into the sorted list:	4	5	9	15	3	8	11	2
3 is now inserted into the sorted list:	3	4	5	9	15	8	11	2
8 is now inserted into the sorted list:	3	4	5	8	9	15	11	2

Insertion sort

List to be sorted:	9	5	4	15	3	8	11	2
Leave the first item at the start:	9	5	4	15	3	8	11	2
5 is now inserted into the sorted list:	5	9	4	15	3	8	11	2
4 is now inserted into the sorted list:	4	5	9	15	3	8	11	2
15 is now inserted into the sorted list:	4	5	9	15	3	8	11	2
3 is now inserted into the sorted list:	3	4	5	9	15	8	11	2
8 is now inserted into the sorted list:	3	4	5	8	9	15	11	2
11 is now inserted into the sorted list:	3	4	5	8	9	11	15	2

Insertion sort

List to be sorted:	9	5	4	15	3	8	11	2
Leave the first item at the start:	9	5	4	15	3	8	11	2
5 is now inserted into the sorted list:	5	9	4	15	3	8	11	2
4 is now inserted into the sorted list:	4	5	9	15	3	8	11	2
15 is now inserted into the sorted list:	4	5	9	15	3	8	11	2
3 is now inserted into the sorted list:	3	4	5	9	15	8	11	2
8 is now inserted into the sorted list:	3	4	5	8	9	15	11	2
11 is now inserted into the sorted list:	3	4	5	8	9	11	15	2
2 is now inserted into the sorted list:	2	3	4	5	8	9	11	15

Which item did not have to move?

Worksheet 3

- Now complete **Task 2** on **Worksheet 3**

Merging two lists

List A List B

6	8	13	20

2	7	11	16

- The following slides show how to merge two sorted lists together
  - Read item from list A; Read item from list B.
  - Write smaller to output list.
  - Read next item from the list that held the smaller value
  - Repeat until all items written to output list

Merging two lists

List A List B

6	8	13	20

2	7	11	16

- Read item from list A 6
- Read item from list B 2

Compare and write smaller item to output

Output list

2							

Merging two lists

List A List B

6	8	13	20

2	7	11	16

- Read item from list A 6
- Read item from list B 7

Compare and write smaller item to output

Output list

2	6						

Merging two lists

List A List B

6	8	13	20

2	7	11	16

- Read item from list A 8
- Read item from list B 7

Compare and write smaller item to output

Output list

2	6	7					

Merging two lists

List A List B

6	8	13	20

2	7	11	16

- Read item from list A 8
- Read item from list B 11

Compare and write smaller item to output

Output list

2	6	7	8				

Merging two lists

List A List B

6	8	13	20

2	7	11	16

- Continue with the rest of the comparisons

Output list

2	6	7	8	11	13	16	20

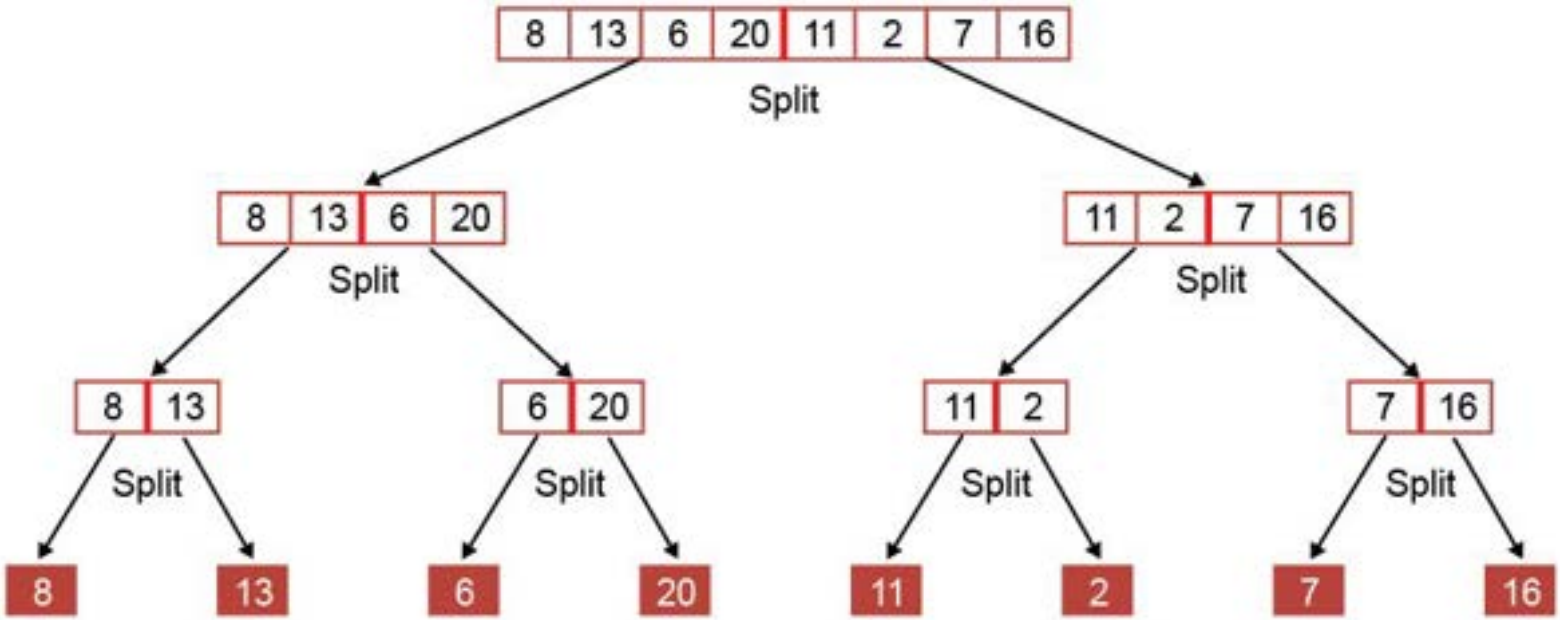
Merge sort

- This is much more efficient than the bubble sort
- The basic steps are:
  - Divide the unsorted list in two
  - Continue to divide these lists into two until there is just one item in each list
  - Now merge each list back until there is only one list remaining – which will be the fully sorted list
- Perform a merge sort on the following list:

8	13	6	20	11	2	7	16

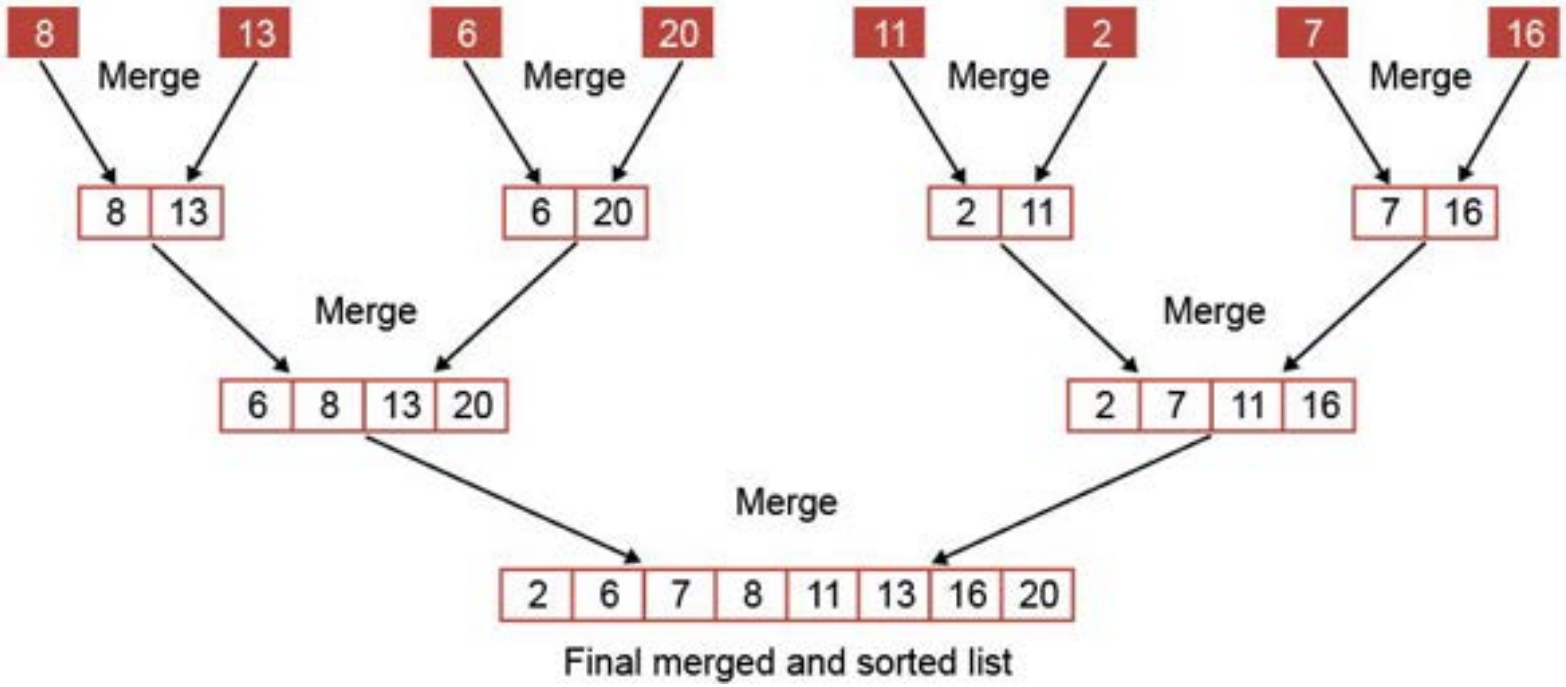
Merge sort part 1

- Divide the unsorted list into n sublists



Merge sort part 2

- Now merge the pairs of sub-lists into a single sorted list



Worksheet 3

- Now complete **Task 3**, **Task 4** and **Task 5** on **Worksheet 3**

Comparing sorts

- The insertion sort is much quicker than the bubble sort, but the merge sort is faster still
- Be sure you can explain how each sort works

Plenary

- Name three algorithms which can be used to sort lists of data
  - Explain briefly how each one works
- Which of these algorithms would you generally choose? Why?

PG ONLINE

Plenary

Answers

- Sorting algorithms
  - Bubble sort
  - Insertion sort
  - Merge sort
- Choice of algorithm
  - Merge sort is generally faster to sort lists, so would usually be the recommended algorithm

PG ONLINE

Copyright

© 2024 PG Online Limited

The contents of this unit are protected by copyright.

This unit and all the worksheets, PowerPoint presentations, teaching guides and other associated files distributed with it are supplied to you by PG Online Limited under licence and may be used and copied by you only in accordance with the terms of the licence. Except as expressly permitted by the licence, no part of the materials distributed with this unit may be used, reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic or otherwise, without the prior written permission of PG Online Limited.

Licence agreement

This is a legal agreement between you, the end user, and PG Online Limited. This unit and all the worksheets, PowerPoint presentations, teaching guides and other associated files distributed with it is licensed, not sold, to you by PG Online Limited for use under the terms of the licence.

The materials distributed with this unit may be freely copied and used by members of a single institution on a single site only. You are not permitted to share in any way any of the materials or part of the materials with any third party, including users on another site or individuals who are members of a separate institution. You acknowledge that the materials must remain with you, the licencing institution, and no part of the materials may be transferred to another institution. You also agree not to procure, authorise, encourage, facilitate or enable any third party to reproduce these materials in whole or in part without the prior permission of PG Online Limited.

OCR
Oxford Cambridge and RSA
An OCR endorsed teaching and learning tool

GCSE OCR

Computer Science
J277

4

Developing algorithms using flowcharts

Unit 6
Algorithms

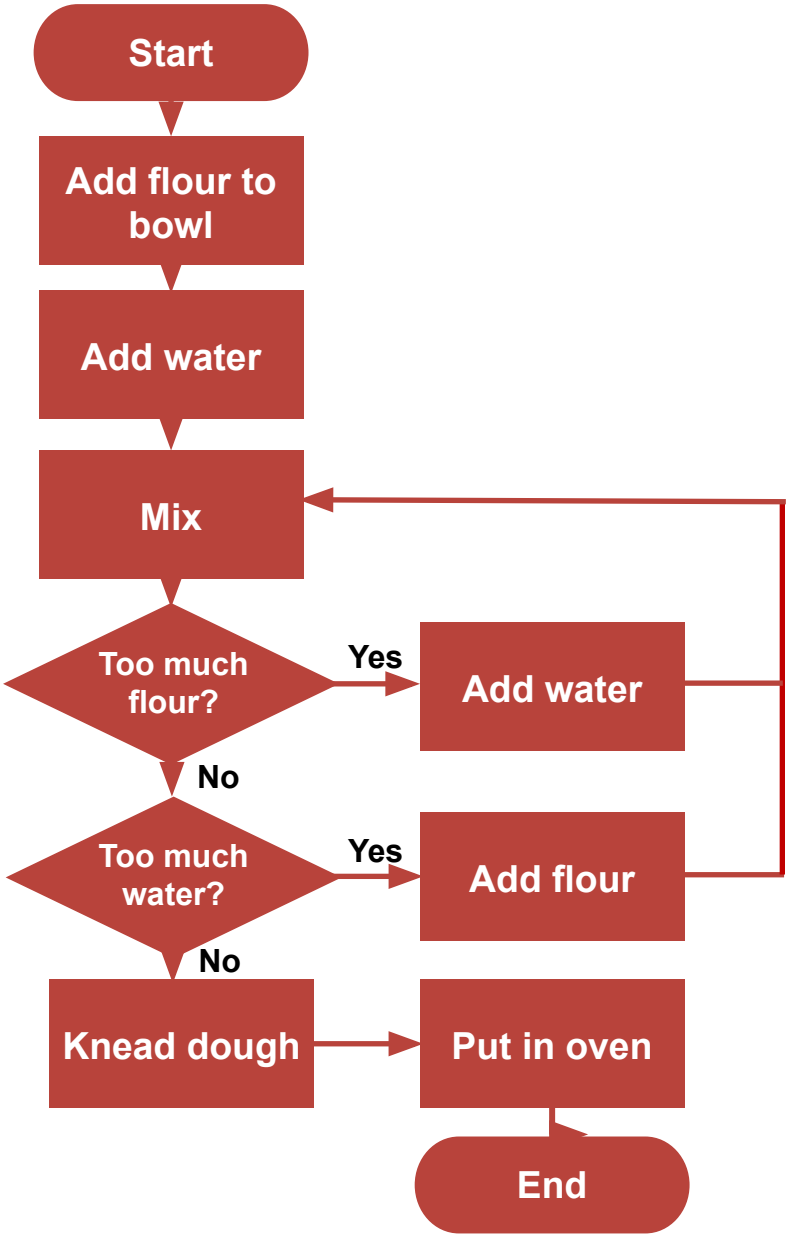
PG ONLINE

Objectives

- Understand flowchart symbols
- Create, interpret, correct, complete and refine algorithms using flowcharts
- Understand arithmetic operators and variables

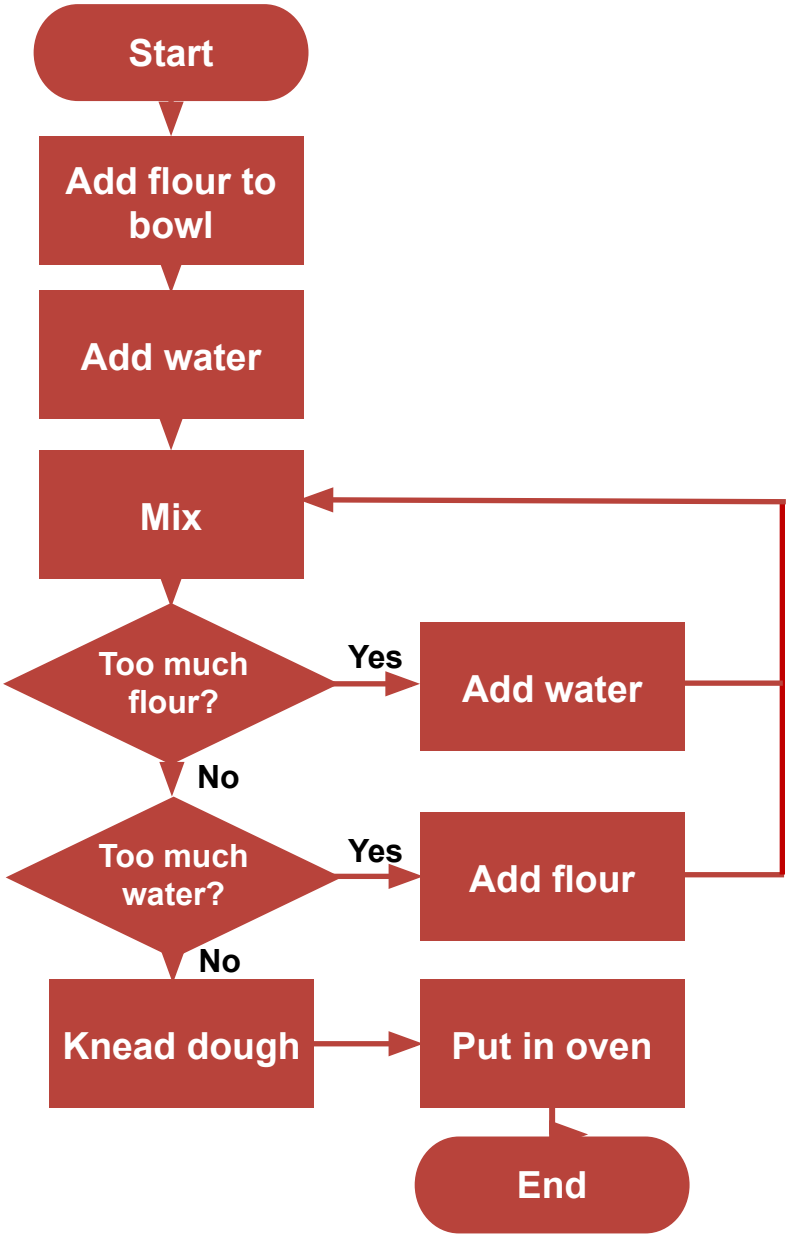
Starter

- Look at the flowchart
  - What is it a flowchart for?
  - Explain how the algorithm works



Starter

- It is a flowchart for making bread
  - The flour and water are mixed together
  - If there is too much flour, more water is added
  - If there is too much water, more flour is added
  - Once correct, the dough is kneaded
  - The dough is put in the oven



Arithmetic operators

Symbol	Description	Example
+	Add	5+7 = 12
-	Subtract	5-7 = -2
/	Divide	15/10 = 1.5
*	Multiply	5*7 = 35
^	Exponent	5^2 = 25
MOD	Modulo (Remainder)	17 MOD 3 = 2
DIV	Integer division	17 DIV 3 = 5

Using arithmetic operators

- What is assigned to x after each of the following operations?

x = (4 + 3) * 6

x = (10^2) / 4

x = 20 – 8 * 2

x = ((5 * 7) + 3) / 4

x = 23 MOD 5

x = 23 DIV 5

x = ((3^2) * 4 + 7) DIV 4

Using arithmetic operators

Answers

- What is assigned to x after each of the following operations?

x = (4 + 3) * 6 42

x = (10^2) / 4 25

x = 20 − 8 * 2 4 ***(remember BIDMAS)***

x = ((5 * 7) + 3) / 4 9.5

x = 23 MOD 5 3 ***(4 r 3 discard the integer)***

x = 23 DIV 5 4 ***(4 r 3 discard the remainder)***

x = ((3^2) * 4 + 7) DIV 4 10

What is a variable?

- A variable is a **location in memory** in which you can **temporarily store a value such as a string or number**
  - It is used like an empty box
- You can choose a name for the box – this is called a ‘variable name’
  - What you store in the box can be changed while the program is running

PG ONLINE

Variables

- When you write a statement such as

```
total = mark1 + mark2
```

- `total`, `mark1` and `mark2` are variables and you are assigning a value to `total`
- When you write a statement such as

```
total = total + mark3
```

  you are saying:

- "Add the value in the location called `mark3`, to the value in the location called `total` and store the result in the location named `total`"

PG ONLINE

What is an algorithm?

- We use ‘algorithms’ every day without even being aware of it
- Have you ever
  - made a cup of tea?
  - followed directions to a destination?
  - taken part in a school play?

PG ONLINE

An algorithm for a computer

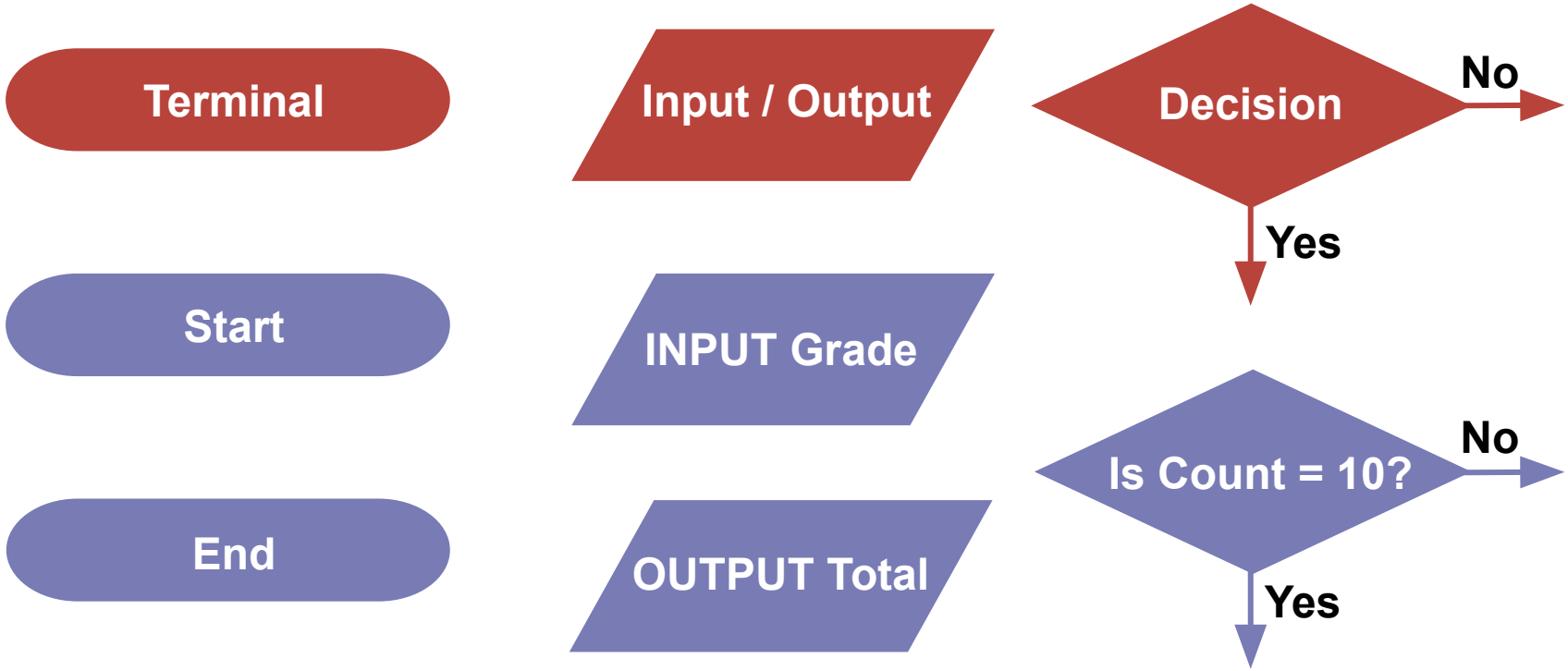
- Imagine programming a self-driving car…
  - What could possibly go wrong?

Writing algorithms

- An algorithm is a series of steps to solve a problem or carry out a task
- There are two common ‘tools’ to help plan and write down the steps needed:
  - Flowcharts
  - Pseudocode
- How many different flowchart symbols do you know?

Start

Flowchart symbols



Flowchart symbols

Line
Shows direction of flow

Process

count = count + 1

total = total + mark

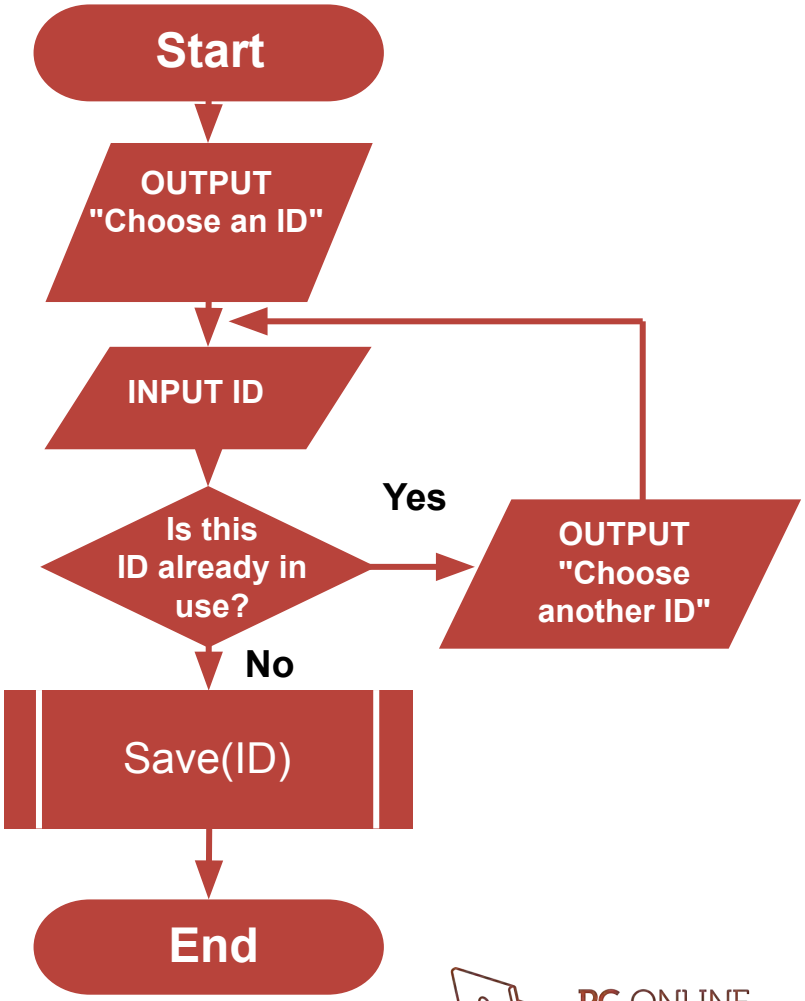
Sub program

showMenu()

search("Jack")

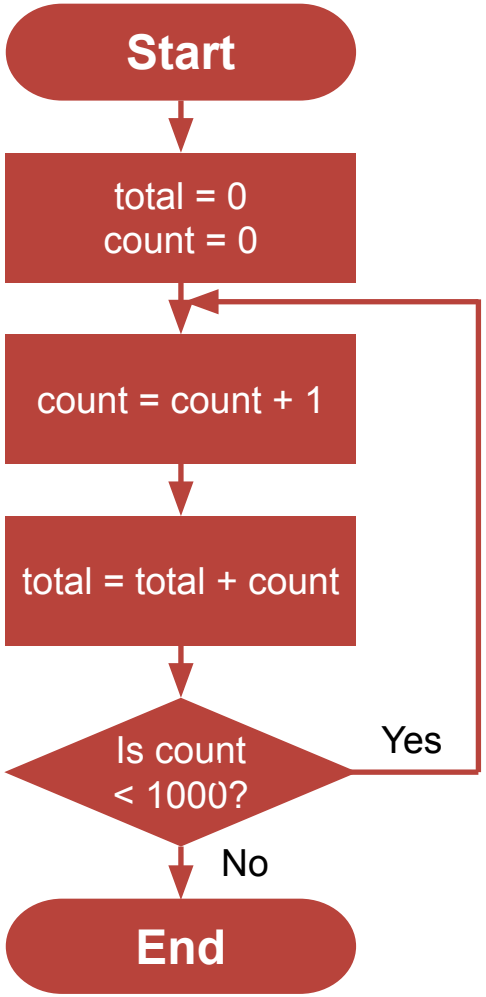
Drawing a flowchart

- An algorithm is a series of steps to solve a problem or carry out a task
  - What algorithm does this flowchart represent?



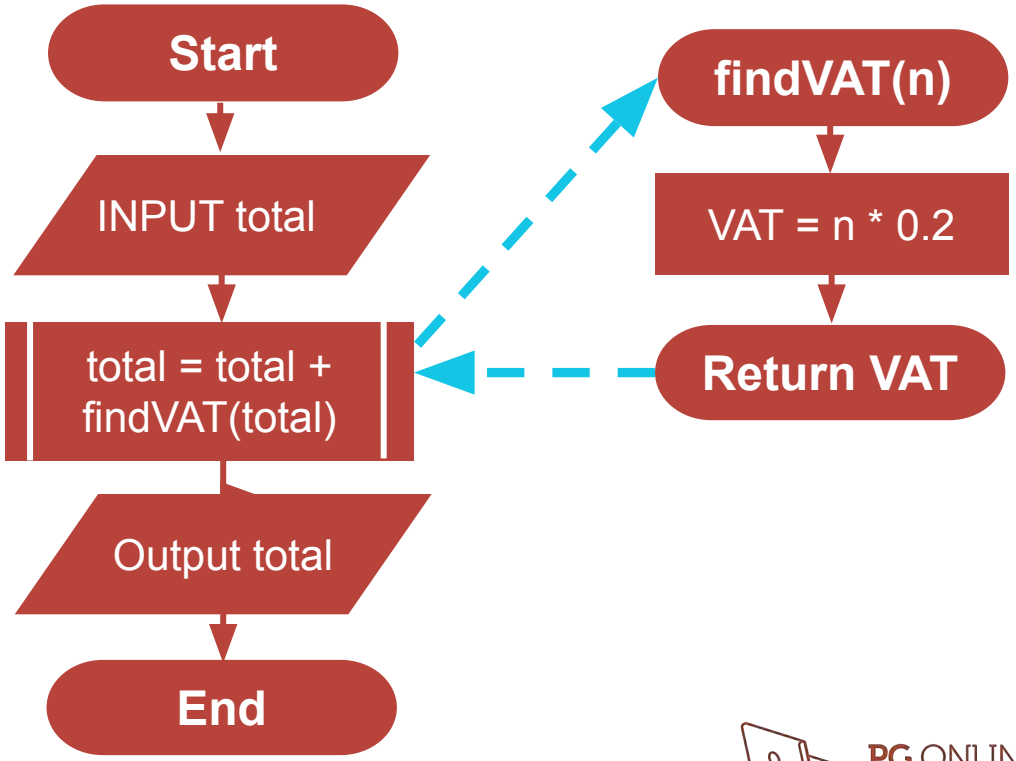
Counting and totalling

- The statement count = count + 1 means “Add 1 to the variable called count”



Sub programs

- Sub programs are used when you wish to call another procedure or function
  - They use a box with lines either side
  - In this example, findVAT is called from the sub program box
  - The findVAT sub program then runs and returns the answer to where it was called



Worksheet 4

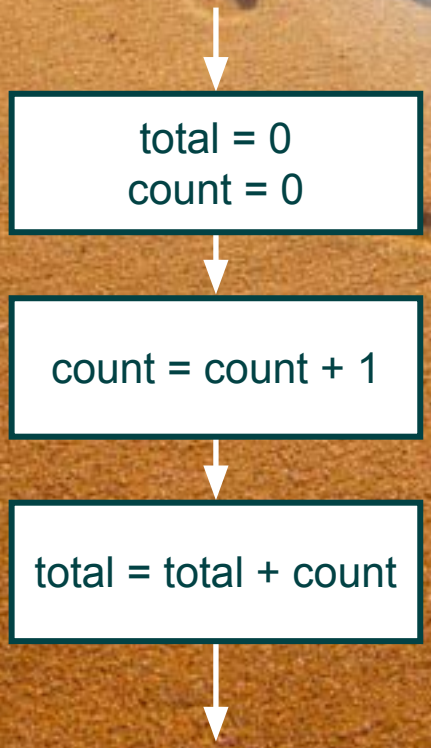
- Now complete **Task 1** and **Task 2** on **Worksheet 4**

Program structures

- There are three basic programming constructs used to control the flow of a program:
  - Sequence
  - Selection
  - Iteration (repetition)

Sequence

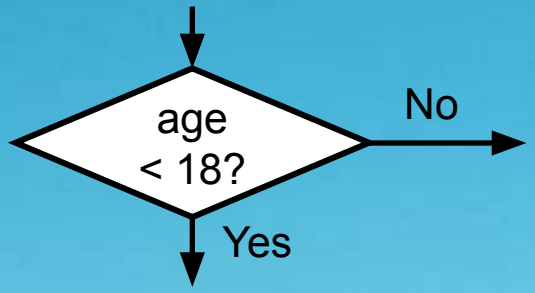
- A sequence is a series of steps which are completed one after the other
  - In a flow diagram they are shown as process, input or output symbols shown one after the other



PG ONLINE

Selection

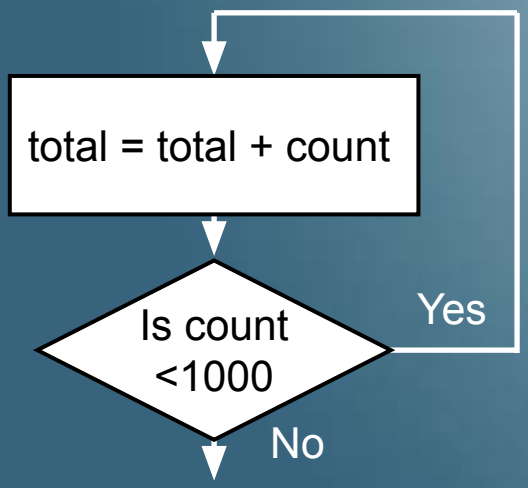
- Selection is the ability to choose different paths through a program
  - In flowcharts, decision symbols are used for selection



PG ONLINE

Iteration

- Iteration means repeating a part of a program
  - It is also referred to as repeating or looping
  - In flowcharts, iteration is shown by using arrows that repeat a section of the flowchart
  - A decision will control whether more iterations are required

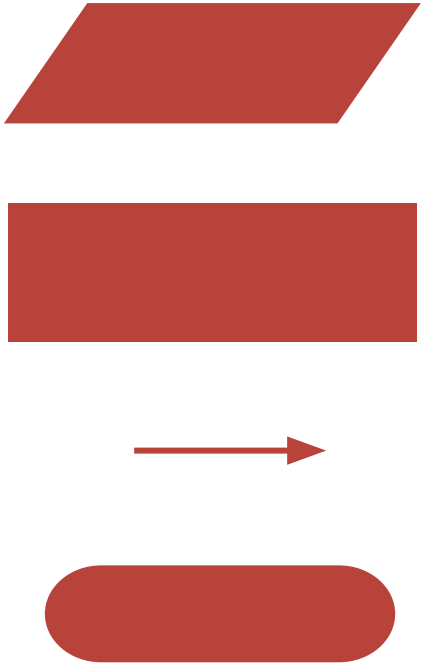


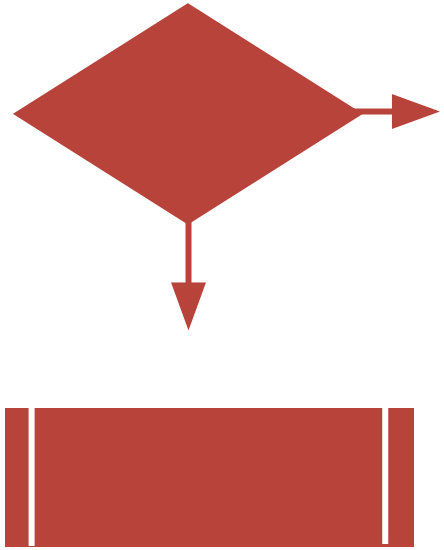
Worksheet 4

- Now complete **Task 3**, **Task 4** and **Task 5** on **Worksheet 4**

Plenary

- What are each of the following symbols?
  - Explain to a partner what each one does





Plenary

Answers

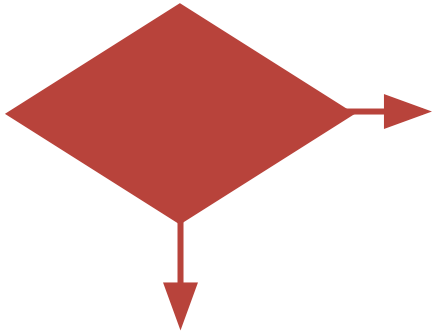
- What are each of the following symbols?
  - Explain to a partner what each one does

Input / Output

Process – Maths operations and assignment of variables

Line – shows direction of flow

Terminal – for start and stop



Decision – change flow based on a decision

Sub program – call a different function or procedure

Copyright

© 2024 PG Online Limited

The contents of this unit are protected by copyright.

This unit and all the worksheets, PowerPoint presentations, teaching guides and other associated files distributed with it are supplied to you by PG Online Limited under licence and may be used and copied by you only in accordance with the terms of the licence. Except as expressly permitted by the licence, no part of the materials distributed with this unit may be used, reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic or otherwise, without the prior written permission of PG Online Limited.

Licence agreement

This is a legal agreement between you, the end user, and PG Online Limited. This unit and all the worksheets, PowerPoint presentations, teaching guides and other associated files distributed with it is licensed, not sold, to you by PG Online Limited for use under the terms of the licence.

The materials distributed with this unit may be freely copied and used by members of a single institution on a single site only. You are not permitted to share in any way any of the materials or part of the materials with any third party, including users on another site or individuals who are members of a separate institution. You acknowledge that the materials must remain with you, the licencing institution, and no part of the materials may be transferred to another institution. You also agree not to procure, authorise, encourage, facilitate or enable any third party to reproduce these materials in whole or in part without the prior permission of PG Online Limited.

OCR
Oxford Cambridge and RSA
An OCR endorsed teaching and learning tool

GCSE OCR

Computer Science
J277

5

Developing algorithms using pseudocode

Unit 6
Algorithms

PG ONLINE

Objectives

- Define the data types integer, real, Boolean, character, string
- Be able to use Boolean operators
- Write algorithms in pseudocode involving sequence, selection and iteration

Starter

- Look at the symbols and keywords below
  - What do you think each one means?

Symbol / keyword	Meaning	Symbol / keyword	Meaning
`<`		`+`	
`<=`		`if elseif else`	
`>`		`switch case default`	
`>=`		`input()`	
`==`		`print()`	
`=`		`for`	
`!=`		`while`	
`*`		`do until`	
`^`		`str()`	
`+`		`int()`	

Starter

Answers

- Look at the symbols and keywords below
  - What do you think each one means?

Symbol / keyword	Meaning	Symbol / keyword	Meaning
`<`	Less than	`+`	Concatenation
`<=`	Less than or equal to	`if elseif else`	Branch depending on condition
`>`	Greater than	`switch case default`	Branch depending on case
`>=`	Greater than or equal to	`input()`	Get user input
`==`	Equal to	`print()`	Output to the user
`=`	Assignment	`for`	Repeat a set number of times
`!=`	Not equal to	`while`	Repeat while a condition is true
`*`	Multiply	`do until`	Do a loop until a condition is true
`^`	Exponent	`str()`	Convert to a string
`+`	Addition	`int()`	Convert to an integer

Data types

- You will use the following data types in your algorithms:

Data type	Description	Example
INTEGER	A whole number	`1475, 0, -5`
REAL	A number with a decimal point	`56.75, 6.0, -2.456, 0.0`
BOOLEAN	Either TRUE or FALSE	`TRUE, FALSE`
CHARACTER	A single alphabetic or numeric character	`'a', 'K', '4', '@', '%'`
STRING	A sequence of one or more characters	`"Jo Hobson", "123"`

Comparison operators

- The following operators are used to compare two values

>	greater than
>=	greater than or equal to
<	less than
<=	less than or equal to
==	equal to
!=	not equal to

Boolean expressions

- A Boolean expression evaluates to TRUE or FALSE
- What do each of the following expressions evaluate to?
  - `35 >= 5 * 7`
  - `'A' < 'B'`
  - `100 > 10^2`
  - `25 <= 2 * 5`
  - `2^4 == 16`
  - `n^2 == n * n`

Boolean expressions

Answers

- A Boolean expression evaluates to TRUE or FALSE
- What do each of the following expressions evaluate to?
  - `35 >= 5 * 7` True
  - `'A' < 'B'` True
  - `100 > 10^2` False
  - `25 <= 2 * 5` False
  - `2^4 == 16` True
  - `n^2 == n * n` True

Worksheet 5

- Now complete **Task 1** and **Task 2** on **Worksheet 5**

Introducing pseudocode

- Pseudocode is a kind of structured English for describing algorithms
  - It allows a programmer to focus on the logic of the algorithm without being distracted by the exact syntax of the programming language
  - You will see pseudocode statements written in a consistent style in exam questions, but you can use alternative statements so long as the meaning is clear
  - You may see the exam board style of pseudocode referred to as *OCR Exam Reference Language* or *ERL*

Programming constructs

- Last lesson we covered three basic ways of controlling the flow of a program:
  - Sequence
  - Selection
  - Iteration
- What do each of these mean?

Sequence

- The statements are executed one by one, in the order they are written
  - An example in pseudocode would be:

Order of execution

```
mealCost = 4.00

drinkCost = 2.00

total = mealCost + drinkcost
```

Selection

- An **IF** statement is a type of selection statement
- The next statement to be executed depends on whether the condition being tested is TRUE or FALSE

```
hoursPerNight = int(input("How many hours a night do
                                you sleep?"))
if hoursPerNight < 8 then
    print("That's not enough!")
else
    print("That's plenty!")
endif
```

Execute this if TRUE

Execute this if FALSE

The switch/case statement

- The switch/case statement is used if there are several possible options to be tested

```
switch optionChosen:
    case 1:
        print("You chose option 1")
    case 2:
        print("You chose option 2")
    case 3:
        print("You chose option 3")
    default:
        print("Please select a valid choice")
endswitch
```

Execute if optionChosen is 1

Execute if optionChosen is 2

Execute if optionChosen is 3

Execute if no cases match

PG ONLINE

Assignment pseudocode

- To assign a value to a variable, you can write statements such as

```
total = 0
cost = adult * 2 + child * 3
counter = counter + 1
```

1. Write a statement of pseudocode to subtract discount from markedPrice to give salePrice
2. Write a statement of pseudocode to change the value stored in total to include VAT (at 20%)

Assignment pseudocode

Answers

1. Write a pseudocode statement to subtract discount from markedPrice to give salePrice
   - `salePrice = markedPrice - discount`
2. Write a pseudocode statement to change the value stored in total to include VAT (at 20%)
   - `total = total * 0.2 + total`
     OR
   - `total = total * 1.2`
     OR
   - `const VAT = 0.2`
     `total = total * (1 + VAT)`

Simple correct solution

Rewritten to require fewer operations

Rewritten to make use of a constant

Pseudocode for input/output

- Most programs that you will write will ask the user to enter some data, and then accept the user input and assign it to a variable
- The pseudocode used in an exam will look like this:

```
firstName = input("Please enter your name")
```

- This is equivalent to two statements:

```
print("Please enter your name")

firstName = input()
```

Inputting data – examples

Write a single statement to do each of the following:

- Display a prompt "How old are you?"
  Accept an answer from the user
  Assign the answer to a variable called age

- Display a prompt "Press Enter to continue"
  When the user presses Enter, continue to the next statement

Inputting

Answers

Write a single statement to do each of the following:

- Display a prompt "How old are you?"
  Accept an answer from the user
  Assign the answer to a variable called age

```
age = input("How old are you?")
```

- Display a prompt "Press Enter to continue"
  When the user presses Enter, continue to the next statement

```
input("Press Enter to continue")
```

Writing pseudocode

- Write an algorithm in pseudocode for a program which asks the user to enter the cost of two items, adds the two costs and if the cost is greater than £10.00, displays a message “Sorry, too much”. Otherwise, it displays the change due from £10.00

PG ONLINE

Pseudocode solution

Answers

```
item1 = input("Please enter price of first item:")
item2 = input("Please enter price of second item:")
total = item1 + item2
if total > 10.0 then
    print("Sorry, too much")
else
    change = 10.0 – item1 - item2
    print("Change from £10.00 is £" + str(change))
endif
```

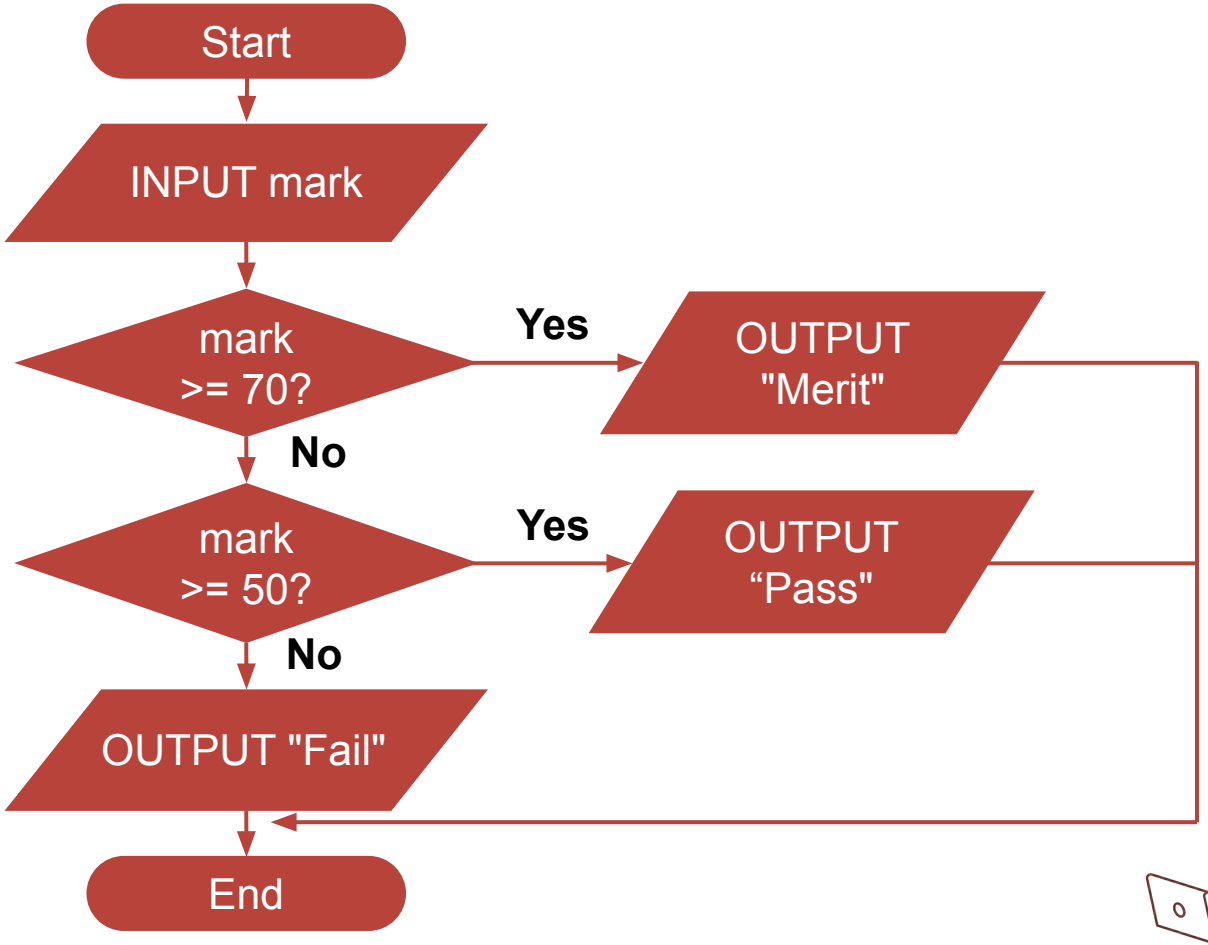
Execute if total > 10 is TRUE

Execute if total > 10 is FALSE

str(change) converts change into a string

Flowchart or pseudocode?

- Write pseudocode corresponding to this flowchart:



Pseudocode

Answers

```
mark = input("Enter mark")
if mark >= 70 then
   print("Merit")
elseif mark >= 50 then
   print("Pass")
else
   print("Fail")
endif
```

Iteration

- Iteration means repetition
- There are three types of iteration statement that you need to know
  - for … next
  - while … endwhile
  - do … until
- Some languages such as Python do not have the do … until statement but you still need to know it

for … next loop

- Use this when you want to execute the loop a specific number of times

```
total = 0
for counter=1 to 7
maxTemp = int(input("Enter max temp: "))   total =
total + maxTemp
next counter
averageWeeksTemp = total / 7
print("This week’s average is:")
print(averageWeeksTemp)
```

Repeat the loop 7 times

for … next loop with step

- For loops usually increase a counter by 1 each time an iteration occurs
  - It is possible to increase the counter by any number using the step keyword. For example:

```
for i=0 to 6 step 2
print(i)
next i
```

Output: 0,2,4,6

```
for i=10 to 0 step -3
print(i)
next i
```

Output: 10,7,4,1

while … end while

- Use this when you want to execute the loop **while** a certain condition is true.

```
password = ""
while password != "rE5Bh9dP"
    password = input("Enter password")
endwhile
print("Correct password")
```

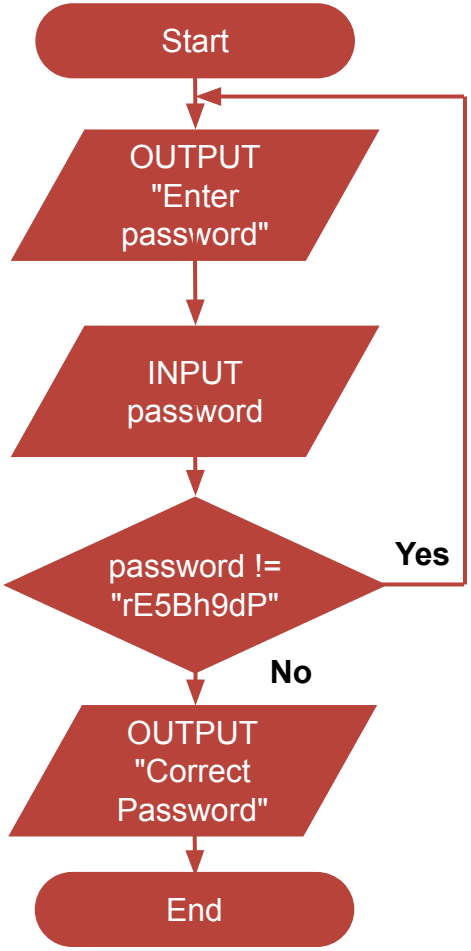
Repeat while password is not equal to “rE5Bh9dP”

Runs when password equals “rE5Bh9dP”

- The condition is tested at the beginning of the loop
- How many times will the loop execute if "rE5Bh9dP" is entered the first time the user is asked?

Flowchart while … end while

```
password = ""
while password != "rE5Bh9dP"
    password = input("Enter password")
endwhile
print("Correct password")
```



do … until

- Use this when you want to keep on executing a loop **until** a certain condition is TRUE
- The condition is tested at the **end** of the loop
- Rewrite this algorithm using a **do while** loop instead of a **while** loop

```
password = ""
while password != "rE5Bh9dP"
password = input("Enter password")
endwhile
print("Correct password")
```

Using do … until

- The condition is not tested until the end of the loop
  - It is always executed at least once

```
password = ""
do
 password = input("Enter password")
until password == "rE5Bh9dP"
print("Correct password")
```

Worksheet 5

- Now complete **Task 3** on **Worksheet 5**

Plenary

- Look at the symbols and keywords below
  - What do each of them mean?

Symbol / keyword	Meaning	Symbol / keyword	Meaning
`<`		`+`	
`<=`		`if elseif else`	
`>`		`switch case default`	
`>=`		`input()`	
`==`		`print()`	
`=`		`for`	
`!=`		`while`	
`*`		`do until`	
`^`		`str()`	
`+`		`int()`	

Plenary

Answers

- Look at the symbols and keywords below
  - How many more do you know than at the start of the lesson?

Symbol / keyword	Meaning	Symbol / keyword	Meaning
`<`	Less than	`+`	Concatenation
`<=`	Less than or equal to	`if elseif else`	Branch depending on condition
`>`	Greater than	`switch case default`	Branch depending on case
`>=`	Greater than or equal to	`input()`	Get user input
`==`	Equal to	`print()`	Output to the user
`=`	Assignment	`for`	Repeat a set number of times
`!=`	Not equal to	`while`	Repeat while a condition is true
`*`	Multiply	`do until`	Do a loop until a condition is true
`^`	Exponent	`str()`	Convert to a string
`+`	Addition	`int()`	Convert to an integer

Copyright

© 2024 PG Online Limited

The contents of this unit are protected by copyright.

This unit and all the worksheets, PowerPoint presentations, teaching guides and other associated files distributed with it are supplied to you by PG Online Limited under licence and may be used and copied by you only in accordance with the terms of the licence. Except as expressly permitted by the licence, no part of the materials distributed with this unit may be used, reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic or otherwise, without the prior written permission of PG Online Limited.

Licence agreement

This is a legal agreement between you, the end user, and PG Online Limited. This unit and all the worksheets, PowerPoint presentations, teaching guides and other associated files distributed with it is licensed, not sold, to you by PG Online Limited for use under the terms of the licence.

The materials distributed with this unit may be freely copied and used by members of a single institution on a single site only. You are not permitted to share in any way any of the materials or part of the materials with any third party, including users on another site or individuals who are members of a separate institution. You acknowledge that the materials must remain with you, the licencing institution, and no part of the materials may be transferred to another institution. You also agree not to procure, authorise, encourage, facilitate or enable any third party to reproduce these materials in whole or in part without the prior permission of PG Online Limited.

OCR
Oxford Cambridge and RSA
An OCR endorsed teaching and learning tool

GCSE OCR

Computer Science
J277

6

Interpret, correct and complete algorithms

Unit 6
Algorithms

PG ONLINE

Objectives

- Understand the purpose of a given algorithm and how an algorithm works
- Understand how to determine the correct output of an algorithm for a given set of data
- Understand how to identify and correct errors in algorithms
- Create and use of trace tables to follow an algorithm

Starter

- When programming, it is common to make mistakes
  - What techniques can a programmer use to find mistakes in their programs?

PG ONLINE

Starter

Answers

- There are a number of ways
  - The first is to use a trace table – this is where the programmer goes through the code, line by line, writing down the values of variables
  - Alternatively IDEs (Integrated development environments) can be used
  - These have the ability to set breakpoints, step through code and watch variables. This is similar to trace tables and allows the computer to do the processing

Trace tables

- Sometimes you may be given an algorithm in the form of a flowchart or pseudocode, and asked to determine its purpose
- One way to do this is to use a trace table
  - Trace tables are used to determine the outputs from a program as it runs
  - They enable a programmer to find errors in their programs

Using a trace table

- The value of each variable is recorded as it changes
- What value is output from the code below?

```
num = 3
n = 0
while n < 4
    num = num + n
    n = n + 1
endwhile
print(num)
```

num	n	n < 4	OUTPUT
3	0	TRUE	

Using a trace table

- The value of each variable is recorded as it changes
- What value is output from the code below?

```
num = 3
n = 0
while n < 4
    num = num + n
    n = n + 1
endwhile
print(num)
```

num	n	n < 4	OUTPUT
3	0	TRUE	
3	1	TRUE	

Using a trace table

- The value of each variable is recorded as it changes
- What value is output from the code below?

```
num = 3
n = 0
while n < 4
    num = num + n
    n = n + 1
endwhile
print(num)
```

num	n	n < 4	OUTPUT
3	0	TRUE	
3	1	TRUE	
4	2	TRUE	

Using a trace table

- The value of each variable is recorded as it changes
- What value is output from the code below?

```
num = 3
n = 0
while n < 4
    num = num + n
    n = n + 1
endwhile
print(num)
```

num	n	n < 4	OUTPUT
3	0	TRUE	
3	1	TRUE	
4	2	TRUE	
6	3	TRUE	

Using a trace table

- The value of each variable is recorded as it changes
- What value is output from the code below?

```
num = 3
n = 0
while n < 4
    num = num + n
    n = n + 1
endwhile
print(num)
```

num	n	n < 4	OUTPUT
3	0	TRUE	
3	1	TRUE	
4	2	TRUE	
6	3	TRUE	
9	4	FALSE	9

Creating a trace table

- A trace table is useful for
  - Determining the purpose of an algorithm
  - Finding the output of an algorithm
  - Finding errors in an algorithm
- To draw a trace table, make a column for each variable used, in the order in which they appear
- You don’t need to fill in a value for a variable which does not change in a particular row

Determining the function of an algorithm

- Complete the trace table for the algorithm and state its function:
  - Assume the user enters values 8, 3, 6, 5, 10, 6

```
total = 0
for count = 1 to 3
    base = input()
    height = input()
    area =
(base*height)/2
    total = total + area
next count
result = total / 3
print(result)
```

total	count	base	height	area	output
0	1	8	3	12	

Determining the function of an algorithm

- Complete the trace table for the algorithm and state its function:
  - Assume the user enters values 8, 3, 6, 5, 10, 6

```
total = 0
for count = 1 to 3
    base = input()
    height = input()
    area =
(base*height)/2
    total = total + area
next count
result = total / 3
print(result)
```

total	count	base	height	area	output
0	1	8	3	12	
12	2	6	5	15	

Determining the function of an algorithm

- Complete the trace table for the algorithm and state its function:
  - Assume the user enters values 8, 3, 6, 5, 10, 6

```
total = 0
for count = 1 to 3
    base = input()
    height = input()
    area =
(base*height)/2
    total = total + area
next count
result = total / 3
print(result)
```

total	count	base	height	area	output
0	1	8	3	12	
12	2	6	5	15	
27	3	10	6	30	

Determining the function of an algorithm

- The algorithm finds the average of the areas of the triangles
  - Assume the user enters values 8, 3, 6, 5, 10, 6

```
total = 0
for count = 1 to 3
    base = input()
    height = input()
    area =
(base*height)/2
    total = total + area
next count
result = total / 3
print(result)
```

total	count	base	height	area	output
0	1	8	3	12	
12	2	6	5	15	
27	3	10	6	30	
57	4				**19**

Worksheet 6

- Now complete **Task 1**, **Task 2**, **Task 3**, **Task 4** and **Task 5** on **Worksheet 6**

Finding errors

- Trace tables are often used to find errors
  - The Fibonacci sequence takes the previous two numbers to find the next number – e.g. 0, 1, 1, 2, 3, 5, 8, 13…
  - Look at the following code and complete the trace table for it

```
num1 = 0
num2 = 1
print(num1)
print(num2)
for i = 1 to 5
    newNum = num1 + num2
    print(newNum)
    num1 = num2
    num2 = num1 + num2
endfor
```

i	newNum	num1	Num2	output
		0	1	0 1

Finding errors

Answers

- The trace table shows the output will be:
  0, 1, 1, 3, 6, 12, 24 – it should be 0, 1, 1, 2, 3, 5, 8
  - Change one line of code below to fix the problem
  - Create another table and check your algorithm

```
num1 = 0
num2 = 1
print(num1)
print(num2)
for i = 1 to 5
    newNum = num1 + num2
    print(newNum)
    num1 = num2
    num2 = num1 + num2
endfor
```

i	newNum	num1	Num2	output
		0	1	0 1
1	1	1	2	1
2	3	2	4	3
3	6	4	8	6
4	12	8	16	12
5	24	16	32	24

Finding errors

Answers

- The trace table shows the output will be:
  0, 1, 1, 3, 6, 12, 24 – it should be 0, 1, 1, 2, 3, 5, 8
  - Change one line of code below to fix the problem
  - Create another table and check your algorithm

```
num1 = 0
num2 = 1
print(num1)
print(num2)
for i = 1 to 5
    newNum = num1 + num2
    print(newNum)
    num1 = num2
    num2 = newNum
endfor
```

i	newNum	num1	Num2	output
		0	1	0 1
1	1	1	2	1
2	3	2	4	3
3	6	4	8	6
4	12	8	16	12
5	24	16	32	24

Errors

- The original Fibonacci program worked, but gave a result that wasn't intended by the programmer
  - This is an example of a **logical error**
- The below program has three **syntax errors** – what are they?

```
name = input("Type in your name)
if name = "George"
print("hello" name)
else
print("Your name isn't George")
endif
```

Errors

Answers

- Corrected code:

```
name = input("Type in your name")

if name == "George"

print("hello" + name)

else

print("Your name isn't George")

endif
```

Missing quote mark to end string

== for equality (= means assignment)

+ needed to concatenate two strings

Plenary

- In pairs answer the following:
  - How is a trace table used to help find errors in a program?
  - Two types of error are syntax errors and logical errors. Explain what both of these mean with an example

Plenary

Answers

- How is a trace table used to help find errors in a program?
  - Variable names and outputs are put in columns
  - The programmer traces through the program line by line updating the values of variables and outputs
  - A row is used for each iteration
- Two types of error are syntax errors and logical errors
  - Syntax errors - doesn't follow the rules of the language e.g. print("hello) - this has no final " for a string
  - Logical errors – the logic of the program is incorrect e.g. average = ((num1 + num2) / 3)

Copyright

© 2024 PG Online Limited

The contents of this unit are protected by copyright.

This unit and all the worksheets, PowerPoint presentations, teaching guides and other associated files distributed with it are supplied to you by PG Online Limited under licence and may be used and copied by you only in accordance with the terms of the licence. Except as expressly permitted by the licence, no part of the materials distributed with this unit may be used, reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic or otherwise, without the prior written permission of PG Online Limited.

Licence agreement

This is a legal agreement between you, the end user, and PG Online Limited. This unit and all the worksheets, PowerPoint presentations, teaching guides and other associated files distributed with it is licensed, not sold, to you by PG Online Limited for use under the terms of the licence.

The materials distributed with this unit may be freely copied and used by members of a single institution on a single site only. You are not permitted to share in any way any of the materials or part of the materials with any third party, including users on another site or individuals who are members of a separate institution. You acknowledge that the materials must remain with you, the licencing institution, and no part of the materials may be transferred to another institution. You also agree not to procure, authorise, encourage, facilitate or enable any third party to reproduce these materials in whole or in part without the prior permission of PG Online Limited.

OCR
Oxford Cambridge and RSA
An OCR endorsed teaching and learning tool

GCSE OCR

Computer Science
J277

1

Programming fundamentals

Unit 7 Programming

PG ONLINE

Objectives

- Understand and use data types: integer, real, Boolean, character and string
- Declare and use constants and variables
- Use input, output and assignment statements
- Use arithmetic operators including MOD and DIV
- Use string handling and conversion functions

Starter

- What is a data type?
  - How many can you name?
- An arithmetic operator is a symbol that will perform an operation on numbers
  - + is an example of an arithmetic operator that uses two numbers, e.g. 5 + 2
  - How many arithmetic operators can you name?

Starter

Answers

- What is a data type?
  The kind of values that can be used in a data item
  - How many can you name?
    integer, float / real, character, string, Boolean
- An arithmetic operator is a symbol that will perform an operation on numbers
  - + is an example of an arithmetic operator that uses two numbers, e.g. 5 + 2
  - How many arithmetic operators can you name?
    +, -, *, /, ^, MOD, DIV

OCR Reference Language

- Throughout these lessons, computer code will be written using OCR Exam Reference Language
  - This is the language that examination questions will be written in
  - Sometimes the syntax may be different or not possible in the programming language you are studying
- Sometimes examples will demonstrate a concept in a real language – in which case, the language used will be stated

PG ONLINE

Using variables in a program

- You have probably used variables in many different ways in programs you have already written

```
numberOfStudents = numberOfStudents + 1
circleArea = 3.142 * radius^2
while NOT found...
answer = 'Y'
print(studentName)
```

- What are the data types of each of the above variables?

Data types

Answers

- Variables will typically be one of the following types:
  - Integer, Boolean, real / float, character or string
- For the variables given, the data types would be:
  - `numberOfStudents` Integer
  - `circleArea` Real / floating point number
  - `found` Boolean
  - `answer` Character
  - `studentName` String
- The data type used will determine the amount of memory that needs to be allocated for the variable

Definitions of data types

Data type	Type of data	Typical amount of memory

Constants

- As well as variables, you can define constants in a program

```
const PI = 3.14157926535
const VAT = 0.2
const MAX_PLAYERS = 6
```

- Constants are typically shown in uppercase
  - Words are separated with an underscore, e.g. MIN_AGE
  - This is known as snake case
- *Why declare a constant instead of a variable?*
- *Can a constant ever change its value?*

Constants

Answers

- Why declare a constant instead of a variable?
  - This prevents the value from being changed accidentally by a part of code
  - It shows a programmer that the value should stay the same throughout the program
  - Processing with constants is a little faster as the actual value is used rather than needing to be looked up in memory (RAM)
- Can a constant ever change its value?
  - A constant cannot be changed when the program is running
  - A constant can be changed by a programmer before the program is compiled or interpreted

Input and output statements

- Most programs accept data from the user, process it in some way, and output a result

```
print("How many hours a night do you sleep?")
hoursPerNight = int(input())
hoursPerWeek = hoursPerNight * 8
print(str(hoursPerWeek))
```

INPUT statement

- An input statement can have a prompt for the user:

```
firstName = input("What is your name? ")
```

- This statement first displays the message “What is your name?” and then waits for the user to enter some text and press Enter
- The response is then assigned to the variable `firstName`

Arithmetic operators

- The operators +, -, * and / are used for addition, subtraction, multiplication and division
  - ^ is used for an exponent (power of)
- The operator DIV is used for integer division, also known as quotient
- MOD (modulo) is used to find the remainder when dividing one integer by another
- What is the result of the following operations?
  - weeks = 31 DIV 7
  - daysLeft = 31 MOD 7

MOD and DIV

Answers

- weeks = 31 DIV 7
  - The variable weeks is assigned the value 4
  - This is because 7 goes into 31 four times (remainder 3)
- daysLeft = 31 MOD 7
  - The variable daysLeft is assigned the value 3
  - This is because the remainder after the division is 3

Orders of precedence

- Remember BIDMAS
  - Brackets
  - Indices
  - Division
  - Multiplication
  - Addition
  - Subtraction
- Calculate: x = (5 – 2) + (16 – 6 / 2)

  y = 7 * 3 + 10 / 2

Are brackets needed in the first expression?

Orders of precedence

Answers

- Calculate: x = (5 – 2) + (16 – 6 / 2)

  x = 3 + (16 – 3)
  x = 3 + 13
  x = 16

  y = 7 * 3 + 10 / 2
  y = 21 + 5
  y = 26

Are brackets needed in the first expression?

No – the answer will still be 16 without the brackets

Worksheet 1

- Now complete **Task 1** on **Worksheet 1**

Strings and numbers

- Strings and numbers used in calculations are represented differently in binary
- Remember that a string is anything held within quote marks
- The string "17" is represented in ASCII binary as
  - 0011000100110111
- The integer 17 is held as in binary as
  - 00010001

Conversions and casting

- Functions are used to convert data types
  - This is also known as casting
- Examples of functions (written in pseudocode) :
  - int(s) converts a string s to an integer
  - float(s) converts a string s to a number with a decimal point
  - str(x) converts an integer or floating point number x to a string
  - bool(s) converts a string (e.g. "True") to a Boolean
  - ASC('a') evaluates to 97, using ASCII
  - CHR(97) evaluates to 'a'

Inputting numeric variables

- Inputs from the user are strings
- Therefore, if you are inputting an integer or real number, you have to convert it before it can be used in a calculation

```
tickets = input("Please enter number of tickets required: ")
tickets = int(tickets)
```

- Or alternatively, in a single statement:

```
tickets = int(input("Please enter number of tickets required: "))
```

Concatenating strings

- **Concatenating** means joining together
  - The + concatenate operator is used to join together strings

```
firstname = "Rose"
surname = "Chan"
fullname = firstname + " " + surname
print(fullname)
```

- What will be output?
- What will be output by the program?

```
x = "6" + "3"
print(x)
```

Concatenating strings

Answers

```
firstname = "Rose"
surname = "Chan"
fullname = firstname + " " + surname
print(fullname)
```

- What will be output? "Rose Chan"
- What will be output by the program

```
x = "6" + "3"
print(x)
"63"
```

String handling functions

Function	Example	Result

- What will be the values of a, b, c and d below?

```
zooName = "London Zoo"
a = zooName.length
b = zooName.substring(1,4)
c = zooName.left(8)
d = zooName.right(5)
```

String handling functions

Answers

Function	Example	Result
`str.length`	`word = "Algorithm"` `print(word.length)`	`9`
`str.substring(start, end)`	`print(word.substring(3,6))`	`"orit"`
`str.left(n)`	`Print(word.left(3))`	`"Alg"`
`str.right(n)`	`Print(word.right(4))`	`"ithm"`

- What will be the values of a, b, c and d below?

```
zooName = "London Zoo"
a = zooName.length          10
b = zooName.substring(1,4)    "ondo"
c = zooName.left(8)         "London Z"
d = zooName.right(5)        "n Zoo"
```

Character conversion functions

- Characters may be converted to their ASCII value and vice versa using the functions:

  ASC(character)

  and

  CHR(integer)

- For example:

```
x = ASC('a')      // sets x = 97
y = CHR(97)       // sets y to 'a'
```

- Evaluate num and letter where

```
num = ASC('c')
letter = CHR(102)
```

Uppercase and lowercase

- A string can be converted to uppercase or lowercase letters as follows:

```
phrase1 = "Good morning"
phrase2 = "HAPPY BIRTHDAY"
print(phrase1.upper)      //"GOOD MORNING"
print(phrase2.lower)      //"happy birthday"
```

- What is the output of the following program?

```
a = "The quality of mercy is not strained."
b = a.length – 30
c = a.substring(0,b)
d = c.upper
print(d)
```

Uppercase and lowercase

Answers

- What is the output of the following program?

```
a = "The quality of mercy is not strained."

b = a.length – 30        b = 37 - 30
              b = 7
c = a.substring(0,b)        c = a.substring(0,7)
              c = "The qua"
d = c.upper           d = "THE QUA"

print(d)              "THE QUA"
```

Using comments

- You should use comments in your programs:
  - to describe the purpose of the program
  - to state the author of the program
  - to explain what the code does
  - to switch off code to debug sections of code
- Pseudocode comments start with a double slash `//`
  - In Python, comments start with #
  - In VB, comments start with `'`
- Comments are ignored when your program is translated to machine code and executed

Worksheet 1

- Now complete **Task 2** and **Task 3** on **Worksheet 1**

Plenary

- In pairs give:
  - 5 different data types
  - 4 arithmetic operators
  - 3 functions used for type conversion / casting
  - 2 arithmetic operators used for integer division and remainder
  - 1 concatenate operator

PG ONLINE

Plenary

Answers

- 5 different data types
  - Integer, real / float, character, string, Boolean
- 4 arithmetic operators
  - +, -, *, /, MOD, DIV, ^
- 3 functions used for type conversion / casting
  - str(), int(), float() / real(), bool()
- 2 arithmetic operators used for integer division and remainder
  - DIV (integer division), MOD (remainder)
- 1 concatenate operator +

Copyright

© 2024 PG Online Limited

The contents of this unit are protected by copyright.

This unit and all the worksheets, PowerPoint presentations, teaching guides and other associated files distributed with it are supplied to you by PG Online Limited under licence and may be used and copied by you only in accordance with the terms of the licence. Except as expressly permitted by the licence, no part of the materials distributed with this unit may be used, reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic or otherwise, without the prior written permission of PG Online Limited.

Licence agreement

This is a legal agreement between you, the end user, and PG Online Limited. This unit and all the worksheets, PowerPoint presentations, teaching guides and other associated files distributed with it is licensed, not sold, to you by PG Online Limited for use under the terms of the licence.

The materials distributed with this unit may be freely copied and used by members of a single institution on a single site only. You are not permitted to share in any way any of the materials or part of the materials with any third party, including users on another site or individuals who are members of a separate institution. You acknowledge that the materials must remain with you, the licencing institution, and no part of the materials may be transferred to another institution. You also agree not to procure, authorise, encourage, facilitate or enable any third party to reproduce these materials in whole or in part without the prior permission of PG Online Limited.

OCR
Oxford Cambridge and RSA
An OCR endorsed teaching and learning tool

GCSE OCR

Computer Science
J277

2

Sequence and selection

Unit 7 Programming

PG ONLINE

Objectives

- Use selection and nested selection statements
- Use NOT, AND and OR when creating Boolean expressions
- Use random number generation

Starter

- What are the **three** basic control structures used in programming?
  - What are **three** operators used in Boolean expressions?

Starter

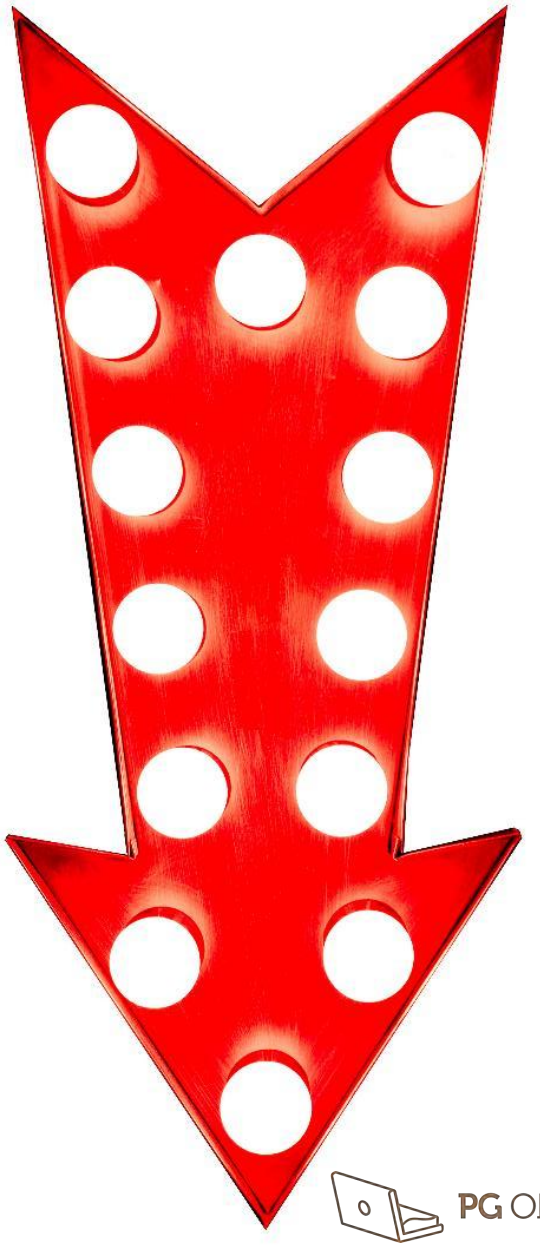
Answers

- What are the **three** basic control structures used in programming?
  - Sequence, selection and iteration
- What are **three** operators used in Boolean expressions?
  - AND, OR, NOT

Sequence

- The statements are executed one by one in the order they are written:

```
mark1 = 78
mark2 = 67
total = mark1 + mark2
average = total / 2
print(average)
```



Selection

- An IF statement is a selection statement
- The next statement to be executed depends on whether the condition being tested is True or False

```
if average >= 80 then
print("Distinction")
else
print("Pass")
endif
```

PG ONLINE

Comparison expressions

- The condition average >= 80 is a Boolean expression
  - The outcome will always evaluate to TRUE or FALSE
- Comparison operators include

==	equal to
!=	not equal to
>	greater than
<	less than

- What are **two** other comparison operators?
  - Why is = not used to mean equal to?

Boolean expressions

Answers

- Here is a full list of comparison operators

Comparison operators	Meaning	Pseudocode example	Result	Notes
==	Equal to	5 == 5	True	Some languages use a single =
!=	Not equal to	5 != 5	False	Visual Basic uses <>
>	Greater than	5 > 5	False	
>=	Greater than or equal to	5 >= 5	True	
<	Less than	5 < 5	False	
<=	Less than or equal to	5 <= 5	True	

- A single = is used for assignment - e.g. age = 14

If statements

- If statements allow different branches to be executed based on the result of a Boolean expression

```
if average >= 80 then
 print("Distinction")
elseif average >= 60 then
 print("Merit")
elseif average >= 40
 print("Pass")
else
    print("Fail")
endif
```

if

[else if]

[...]

[else]

Selection used in games

100%

99 - 60%

59 -25%

<24%

L.O.: To understand and use **selection** (IF **ELIF** ELSE or CASE)

For when there are >2 options to choose from e.g.

A grading system (**A* to U**)

Any time you might need to look down a list to get the answer

Comparing numbers (**greater, less than or equal**)

Etc.

L.O.: To understand and use **selection** (IF **ELIF** ELSE or CASE)

Switch Case

Equivalent to If ELIF ELSE but where there is only one expression.

```
Input a Grade

If the Grade >= 100 Then
   Report "Perfect Score"
ElseIf the Grade > 89 Then
   Report "Grade = A"
ElseIf the Grade > 79 Then
   Report "Grade = B"
ElseIf the Grade > 69 Then
   Report "Grade = C"
ElseIf the Grade > 59 Then
   Report "Grade = D"
Else
   Report "Grade = F"
End If
```

```
Input a Grade

Case based on Grade
   Case >=100
      Report "Perfect Score"
   Case > 89
      Report "Grade = A"
   Case > 79
      Report "Grade = B"
   Case > 69
      Report "Grade = C"
   Case > 59
      Report "Grade = D"
   Default
      Report "Grade = F"
End Case
```

Nested if statements

- If statements may be nested:

```
if member == "child" then
if day == "Saturday" then
    swimPrice = 2.00
else
    swimPrice = 2.50
endif
else
swimPrice = 4.00
endif
```

  - What is the price for an adult on Saturday?
  - What is the price for a child on Sunday?

Nested if statements

Answers

```
if member == "child" then
if day == "Saturday" then
    swimPrice = 2.00
else
    swimPrice = 2.50
endif
else
swimPrice = 4.00
endif
```

- What is the price for an adult on Saturday? 4.00
- What is the price for a child on Sunday? 2.50

Complex Boolean expressions

- Boolean expressions can include the Boolean operators AND, OR and NOT
- For example:

```
if (mark < 0) OR (mark > 100) then ...
```

Operator	Description
AND	Returns TRUE if both conditions are TRUE
OR	Returns TRUE if either of the conditions are TRUE
NOT	A TRUE expression becomes FALSE and vice versa

WHITEBOARDS!

True or False?

- Complete the table:

Mark1	Mark2	Condition	True or False?
80	67	`(mark1 >= 80) AND (mark2 >= 80)`	
82	80	`(mark1 >= 80) OR (mark2 >= 80)`	
35		`(mark1 > 30) OR (mark1 < 50)`	
65		`(mark1 < 30) OR (mark1 > 80)`	
0	75	`NOT(mark1 > 50) AND (mark2 > 50)`	
65	85	`NOT(mark1 < 60) AND NOT (mark2 < 80)`	

True or False?

Answers

- Complete the table:

Mark1	Mark2	Condition	True or False?
80	67	`(mark1 >= 80) AND (mark2 >= 80)`	False
82	80	`(mark1 >= 80) OR (mark2 >= 80)`	True
35		`(mark1 > 30) OR (mark1 < 50)`	True
65		`(mark1 < 30) OR (mark1 > 80)`	False
0	75	`NOT(mark1 > 50) AND (mark2 > 50)`	True
65	85	`NOT(mark1 < 60) AND NOT (mark2 < 80)`	True

COLLABORATIVE
PG ONLINE

Planner

WHAT IS YOUR ROLE?

- **Guide The Team:** You help to decide what the group should do next.
- **Make A Plan:** You break down the task into steps that are easy to follow.
- **Manage Resources:** You make sure that everyone has what they need.

COLLABORATIVE
PG ONLINE

Checker

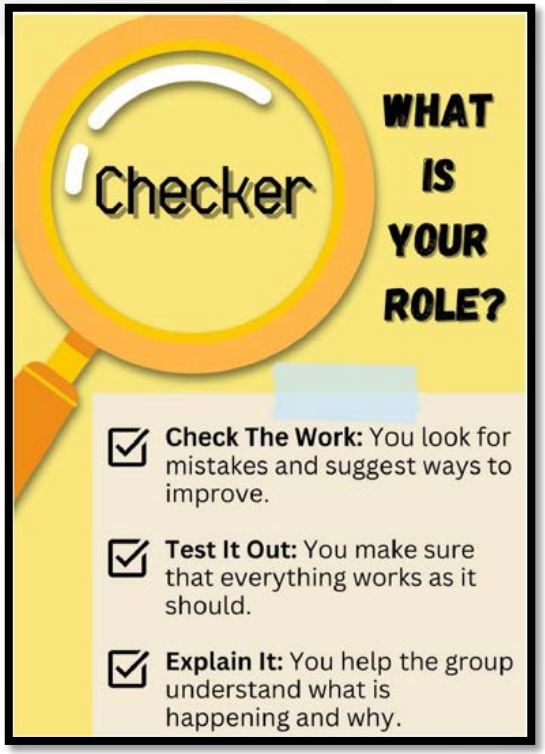
WHAT IS YOUR ROLE?

- **Check The Work:** You look for mistakes and suggest ways to improve.
- **Test It Out:** You make sure that everything works as it should.
- **Explain It:** You help the group understand what is happening and why.

COLLABORATIVE
PG ONLINE

Worksheet 2

- Now complete **Task 1** on **Worksheet 2**



The switch/case statement

- This statement may be used when a selection is to be made from several alternatives, for example when choosing from a menu

```
switch menuChoice:
case "1":
    print("You selected 1")
case "2":
    print("You selected 2")
case "3":
    print("You selected 3")
default:
    print(“Invalid choice")
endswitch
```

```
if menuChoice == “1”:
    print(“You selected 1”)
elif menuChoice == “2”:
    print(“You selected 2”)
elif menuChoice == “3”:
    print(“You selected 3”)
else:
    print(“Invalid choice”)
endif
```

Random numbers

- Programming languages will provide a number of built-in functions that can be used
  - To use them a library may need to be imported
  - For example, in Python, `import random` will import the random library of functions
- A random number between 0 and 100 can then be generated with a statement such as

  `random(0, 100)`

  In Python this would be `random.randint(0, 100)`
- How could you simulate the throw of a die?

Random numbers

Answers

- How could you simulate the throw of a die?
- Pseudocode:

```
die = random(1, 6)
print(die)
```

- Python:

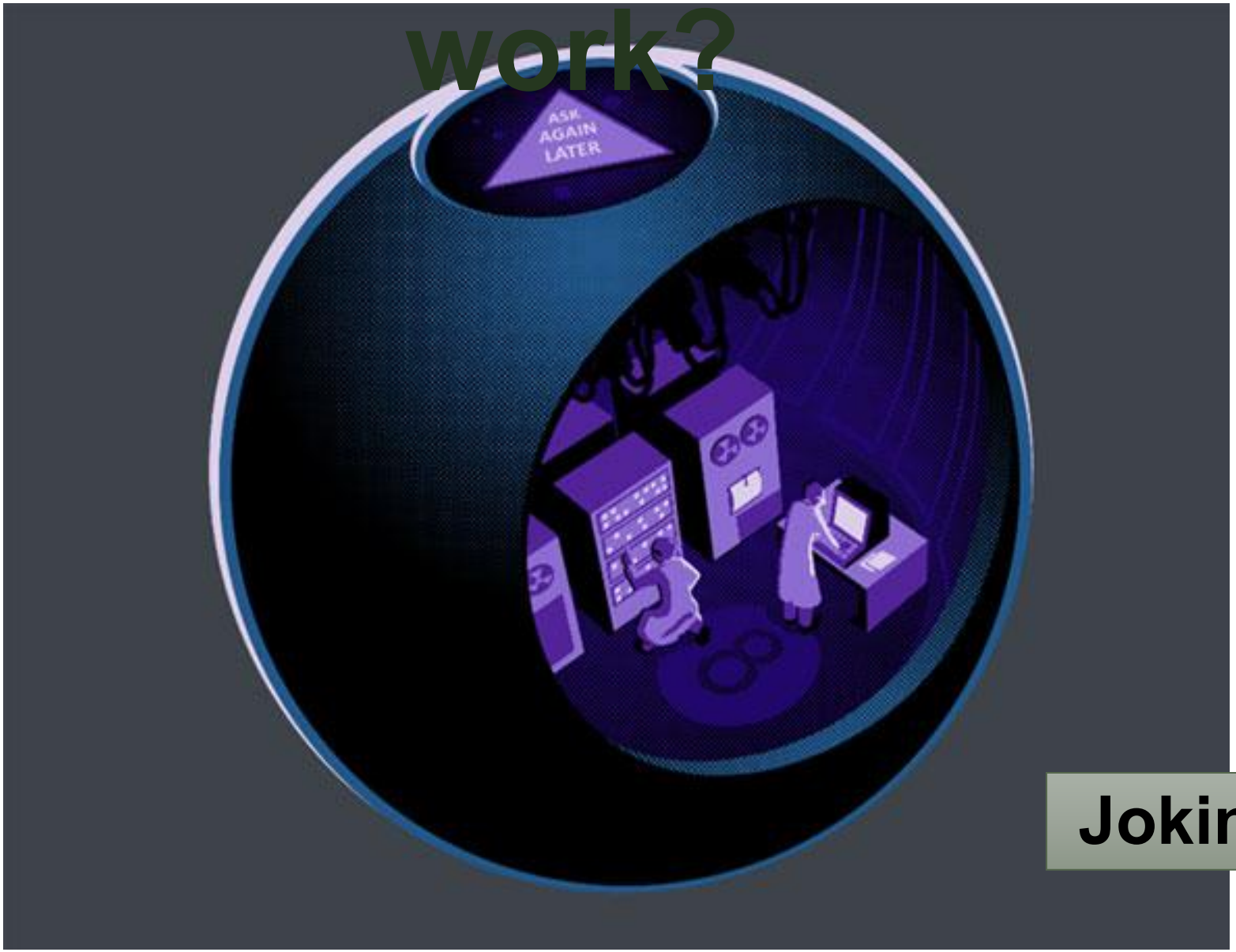
```
import random
die = random.randint(1, 6)
print(die)
```

To create a random real number between -100.0 and 100.0, you would use:

```
randomNumber = random(-100.0, 100.0)
```

https://www.youtube.com/watch?v=mFOracFClBg

How do they work?



Joking!

So how *do* they work?

Each Magic8Ball contains a spin-able disk, with a selection of non-specific answers on the disk's edge.

Shaking the ball spins the disk.

Once it finishes spinning the answer it lands on is shown.

This is purely random and each answer is non specific which gives the illusion of magic!

What must our algorithm do?

1. It must have a bank of “responses” suitable for Yes / No questions,
2. It must ask the user for questions which allows for Yes / No answers,
3. It must **randomly** select an answer from one of the responses in the response bank,
4. *OPTIONAL It could provide flavour text e.g. “Consulting the digital pixies…”,*
5. It must output the answer for the user.

Possible answers [edit]

The 20 answers inside a standard Magic 8 Ball are:

1. It is certain
2. It is decidedly so
3. Without a doubt
4. Yes definitely
5. You may rely on it
6. As I see it, yes
7. Most likely
8. Outlook good
9. Yes
10. Signs point to yes
11. Reply hazy try again
12. Ask again later
13. Better not tell you now
14. Cannot predict now
15. Concentrate and ask again
16. Don't count on it
17. My reply is no
18. My sources say no
19. Outlook not so good
20. Very doubtful

Ten of the possible answers are Positive (●), five are Negative (●), and five are Neutral (●).

5 will do!

5 will do!

```
# Write the possible responses
response1 = "Go for it!"
response2 = "No way, Jose!"
response3 = "I'm not sure, ask me again"
response4 = "Fear of the unknown is what imprisons us"
response5 = "It would be madness to do that!"
```

We need to randomly choose one of the responses

Humans 1 : Computers 0

Unlike humans, computers cannot do anything randomly!

That means Python can't either – obviously…

We need to help our programming language by **importing** a **Random Number Generator**.

Random Number Generators will use things in the real world to simulate randomness. Good ones will use things like the background radiation noise for this

Python uses the computer's internal clock's time.

RANDOMPICKER

import random

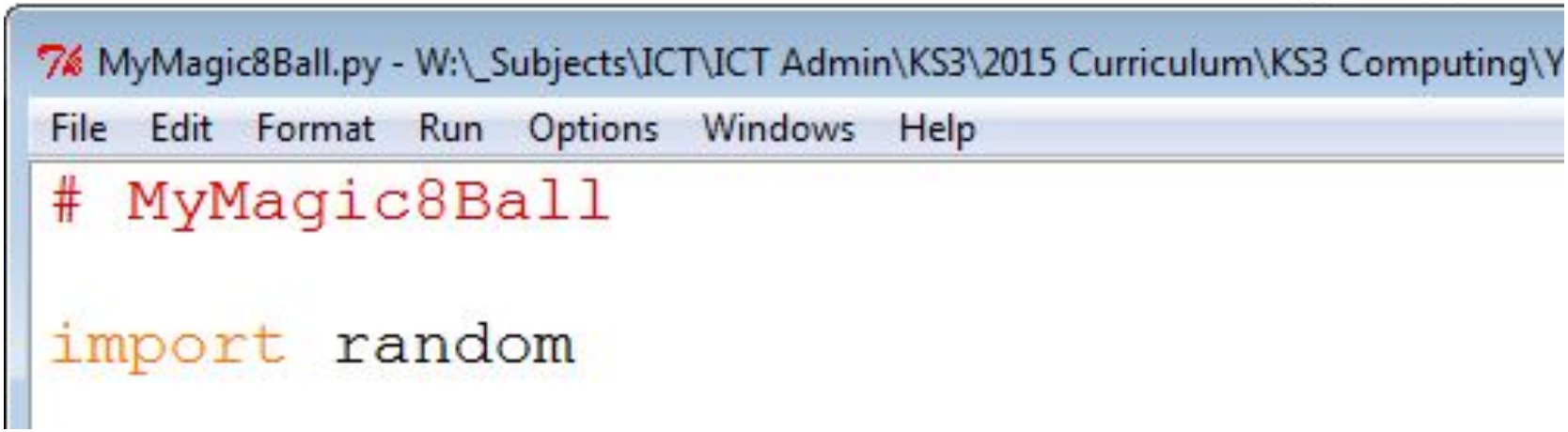
This part generates a random integer from 1 to 5…

randomNumber = random.randint(1,5)

… and stores it in a variable called: **randomNumber**

import

We usually import the libraries we will need for our algorithm e.g. **random** at the start of the program.



RNG
Random
Number
Generator

Without **RNG**s **most** of your games would be BORING!

What games don’t use any RNGs, or anything random?

What must our algorithm

0.It must import the random library

1. It must have a bank of “responses” suitable for Yes / No questions,
2. It must ask the user for questions which allows for Yes / No answers,
3. It must randomly select one of the responses from the response bank,
4. *OPTIONAL It could provide flavour text e.g. “Consulting the digital pixies…”,*
5. It must output the answer for the user.

Begin writing the first half of the program! ☺

0. It must import the random library

1. It must ask the user for questions which allows for Yes / No answers,

2. It must have a bank of “answers” suitable for Yes / No questions,

Second

4. It must randomly select one of the answers from the answer bank,

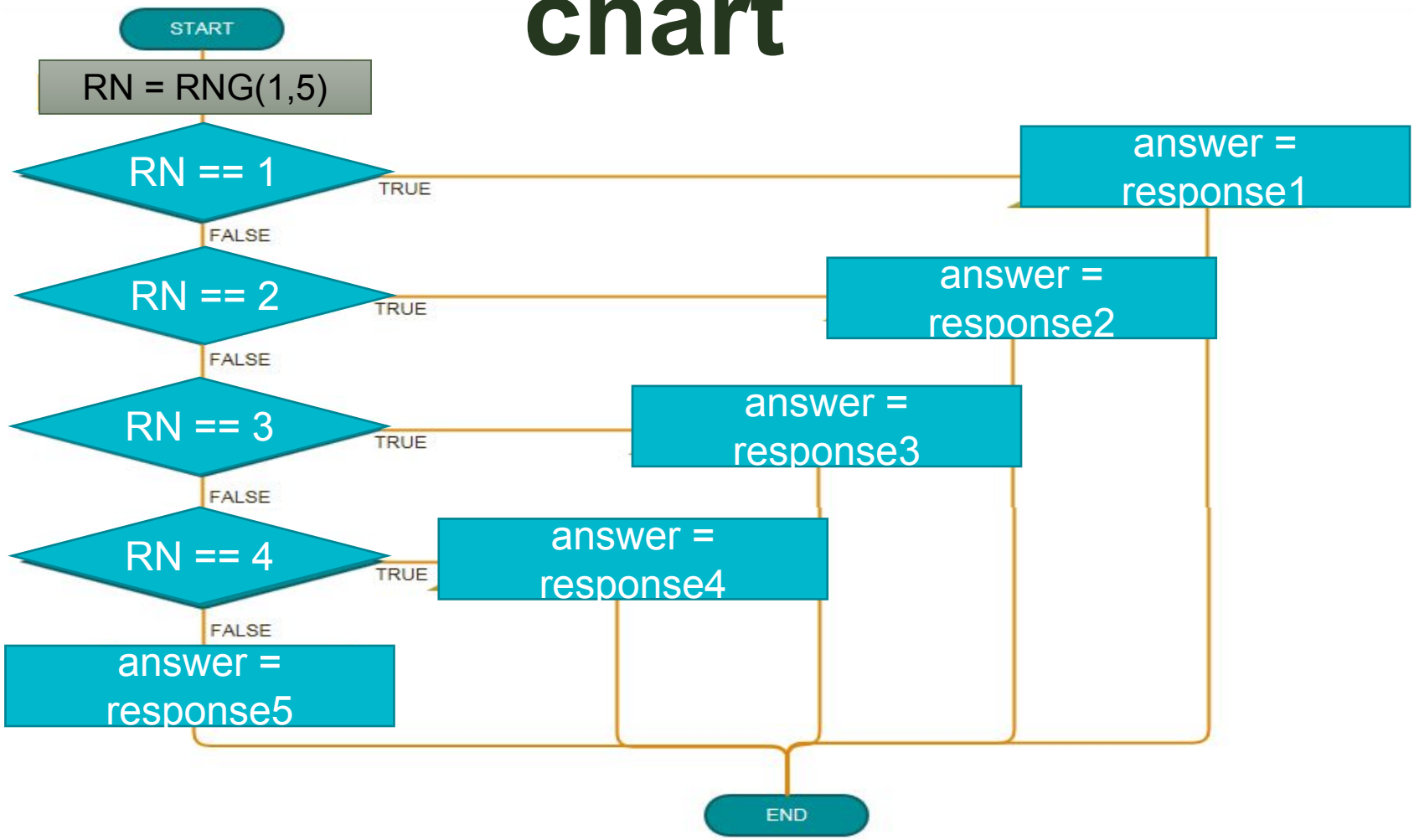
5. *OPTIONAL It could provide flavour text e.g. “Consulting the digital pixies…”,*

6. It must output the answer for the user.

How do we do #4?

‘Grade System’ flow chart

‘Magic 8 Ball’ flow chart



So

```
Grade = 68

if Grade > 80:
    print("You scored an A")

elif Grade > 70
    print("You scored a B")

elif Grade > 60
    print("You scored a C")

elif Grade > 50
    print("You scored a D")

else:
    print("You failed")
```

```
...
randomNumber = random.randint(1,5)

if randomNumber == 1:
    answer = ans1

elif randomNumber == 2:
    answer = ans2

elif randomNumber == 3:
    answer = ans3

elif randomNumber == 4:
    answer = ans4

else:
    answer = ans5
```

Complete the

1. It must import the random library
2. It must ask the user for questions which allows for Yes / No answers,
3. It must have a bank of “answers” suitable for Yes / No questions,
4. It must randomly select one of the answers from the answer bank,
5. *OPTIONAL It could provide flavour text e.g. “Consulting the digital pixies…”,*
6. It must output the answer for the

You can see what is happening behind the scenes but your user can't ☹

Stuff your user ***can*** see / do	Stuff the computer does but user ***doesn't*** see
	Assigning the different response variables
Inputting the question	Storing the question for later
	Generating RNG variable
	Evaluating RNG and assigning a response to answer variable
Outputting the answer to the question	

Worksheet 2

- Now complete **Task 2** and **Task 3** on **Worksheet 2**

Plenary

- Look at the following code:

```
hourlyRate = 15.50
hours = int(input("Enter hours worked this week: "))
if hours > 168 then
print("That’s impossible")
else
totalPay = hours * hourlyRate
print(totalPay)
endif
```

- With a partner, identify each of the following:
  - A comparison operator, a Boolean expression
    a selection structure, a sequence, three assignment operators,
    an output statement, one mathematical operator

Plenary

Answers

Sequence

Selection structure

```
hourlyRate = 15.50

hours = int(input("Enter hours worked this week: "))

if hours > 168 then

    print("That's impossible")

else

    totalPay = hours * hourlyRate

    print(totalPay)

endif
```

Assignment operators

Comparison operator

Boolean expression

Mathematical operator

Output statements

Copyright

© 2024 PG Online Limited

The contents of this unit are protected by copyright.

This unit and all the worksheets, PowerPoint presentations, teaching guides and other associated files distributed with it are supplied to you by PG Online Limited under licence and may be used and copied by you only in accordance with the terms of the licence. Except as expressly permitted by the licence, no part of the materials distributed with this unit may be used, reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic or otherwise, without the prior written permission of PG Online Limited.

Licence agreement

This is a legal agreement between you, the end user, and PG Online Limited. This unit and all the worksheets, PowerPoint presentations, teaching guides and other associated files distributed with it is licensed, not sold, to you by PG Online Limited for use under the terms of the licence.

The materials distributed with this unit may be freely copied and used by members of a single institution on a single site only. You are not permitted to share in any way any of the materials or part of the materials with any third party, including users on another site or individuals who are members of a separate institution. You acknowledge that the materials must remain with you, the licencing institution, and no part of the materials may be transferred to another institution. You also agree not to procure, authorise, encourage, facilitate or enable any third party to reproduce these materials in whole or in part without the prior permission of PG Online Limited.

OCR
Oxford Cambridge and RSA
An OCR endorsed teaching and learning tool

GCSE OCR

Computer Science
J277

3

Iteration

Unit 7 Programming

PG ONLINE

Objectives

- Understand and use iteration in an algorithm
- Write algorithms in pseudocode involving sequence, selection and iteration

Starter

- What does iteration mean?
  - What are **three** types of loop that can be used in programming languages for iteration?

Iteration

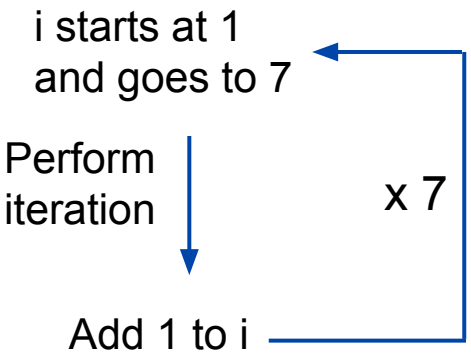
Answers

- Iteration means repetition of a section of code
- There are three types of iteration statement in most languages such as Visual Basic, Java, and C#
  - `for … next`
  - `while … endwhile`
  - `do … until`
- Some languages, such as Python, do not have the `do … until` statement

For loops

- Use this when you want to execute the loop a specified number of times

```
total = 0
for i=1 to 7
  dailyTemperature = int(input())
  total = total +  dailyTemperature
next i
averageWeeksTemp = total / 7
print(averageWeeksTemp)
```



While loops

- Use this when you want to execute the loop **while** a certain condition is true.

```
password = input("Please enter password: ")
while password != "jd8UpP2+d"
  password = input("Invalid password – try again")
endwhile
print("Correct password")
```

Check condition

Perform iteration if the condition is True

- The condition is tested at the **beginning** of the loop
    - How many times will the loop be executed if the user always enters "jd8Upp2d"?

While loops

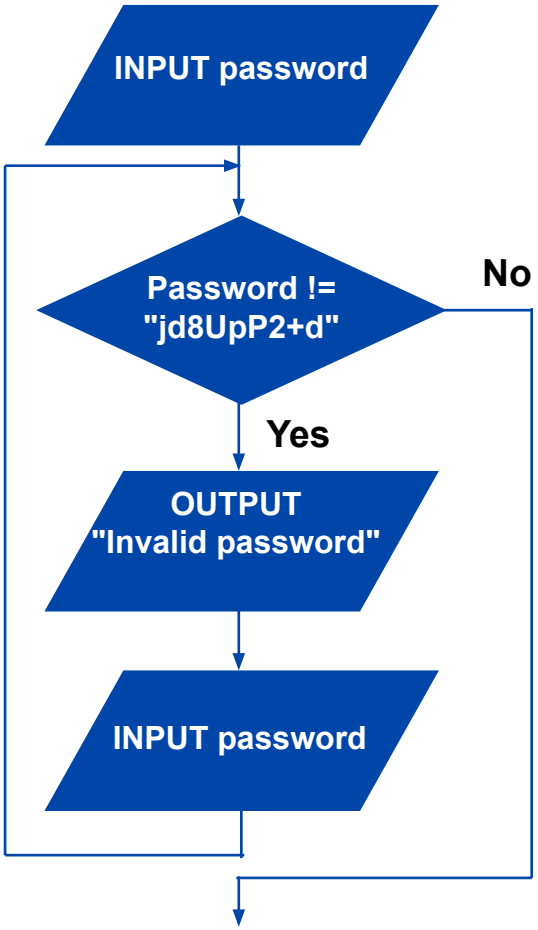
Answers

```
password = input("Please enter password: ")
while password != "jd8UpP2+d"
password = input("Invalid password – try again")
endwhile
print("Correct password")
```

- How many times will the loop be executed if the user always enters "jd8Upp2d"?
  - The loop will continue forever as the password entered is incorrect
  - If a loop continues forever and there is no way to exit it, it is called an **infinite loop**

Flowcharting while … endwhile

```
password = input("Please enter password: ")
while password != "jd8UpP2+d"
 password = input("Invalid password
            - try again")
endwhile
print("Correct password")
```



Worksheet 3

- Now complete **Task 1**,

- **Next Lesson: Task 2** and **Task** 3 on **Worksheet 3**

do … until loops

- Use this when you want to execute the loop **until** a certain condition is True
- The condition is tested at the **end** of the loop
- Rewrite this algorithm using a DO…UNTIL loop instead of a WHILE loop

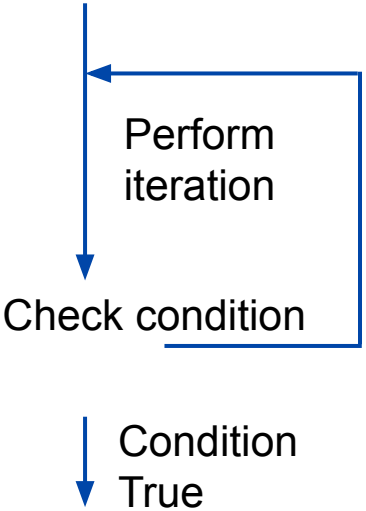
```
password = input("Please enter password: ")
while password != "jd8UpP2+d"
  password = input("Invalid password – try again")
endwhile
print("Correct password")
```

do … until loops

Answers

- Use this when you want to execute the loop **until** a certain condition is True
- The loop will always execute once whereas a while loop might not execute at all
    - The condition is tested at the **end** of the loop

```
password = ""
do
  password = input("Invalid password – try again")
until password == "jd8UpP2+d"
print("Correct password")
```



Nested loops

- It is possible to put one loop inside another
  - This is known as a nested loop

```
for i=1 to 12
  for j=1 to 12
    result = i * j
    print(i + "*" + j + "=" + result)
  next j
next i
```

- What does the above code do?

Nested loops

Answers

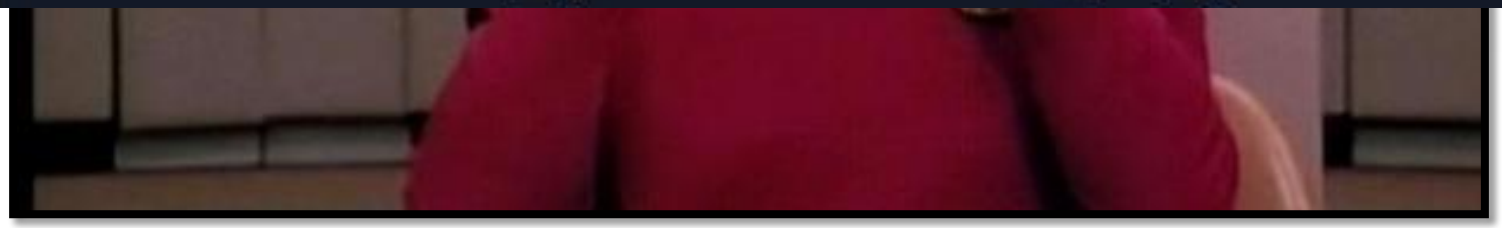
```
for i=1 to 12
  for j=1 to 12
    result = i * j
    print(i + "*" + j + "=" + result)
  next j
next i
```

- What does the above code do?
  - The code displays each of the times tables from 1 to 12 with each times table having 12 rows

Worksheet 3

- Now complete **Task 4** on **Worksheet 3**

```python
import random

number_to_guess = random.randint(1,100)

current_guess = input("Enter a guess: ")
current_guess = int(current_guess)


while current_guess != number_to_guess:
    if current_guess > number_to_guess:
```



Number Guessing Game

1 What variables do we need?

2 What do we need to do with those variables?

Plenary

- What are **three** types of programming loops?
- For each of the following situations, explain how you could use a loop to solve it:
  - Ask a user to enter their name – if they don't enter a name longer than one letter, then keep on asking them to enter their name
  - Count from 1 to 100
  - Repeatedly check the state of a game to see if one of the players has won

Plenary

Answers

- What are **three** types of programming loops?
  - `FOR loop, WHILE loop, DO WHILE loop`
- Ask a user to enter their name – if they don't enter a name longer than one letter, then keep on asking them to enter their name
  - `DO WHILE loop`
- Count from 1 to 100 – `FOR loop`
- Repeatedly check the state of a game to see if one of the players has won
  - `WHILE loop`

Copyright

© 2024 PG Online Limited

The contents of this unit are protected by copyright.

This unit and all the worksheets, PowerPoint presentations, teaching guides and other associated files distributed with it are supplied to you by PG Online Limited under licence and may be used and copied by you only in accordance with the terms of the licence. Except as expressly permitted by the licence, no part of the materials distributed with this unit may be used, reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic or otherwise, without the prior written permission of PG Online Limited.

Licence agreement

This is a legal agreement between you, the end user, and PG Online Limited. This unit and all the worksheets, PowerPoint presentations, teaching guides and other associated files distributed with it is licensed, not sold, to you by PG Online Limited for use under the terms of the licence.

The materials distributed with this unit may be freely copied and used by members of a single institution on a single site only. You are not permitted to share in any way any of the materials or part of the materials with any third party, including users on another site or individuals who are members of a separate institution. You acknowledge that the materials must remain with you, the licencing institution, and no part of the materials may be transferred to another institution. You also agree not to procure, authorise, encourage, facilitate or enable any third party to reproduce these materials in whole or in part without the prior permission of PG Online Limited.

OCR
Oxford Cambridge and RSA
An OCR endorsed teaching and learning tool

GCSE OCR

Computer Science
J277

4

Arrays

Unit 7 Programming

PG ONLINE

Objectives

- Use one- and two-dimensional arrays in the design of solutions to simple problems

Starter

- Imagine a medieval computer game for two players that uses variables to store their data – e.g.

```
playerName1 = "Olivia"
playerName2 = "Noah"
print(playerName1)
print(playerName2)
```

- How would you write this code if the game had an army with 500 players in it?

Starter

Answers

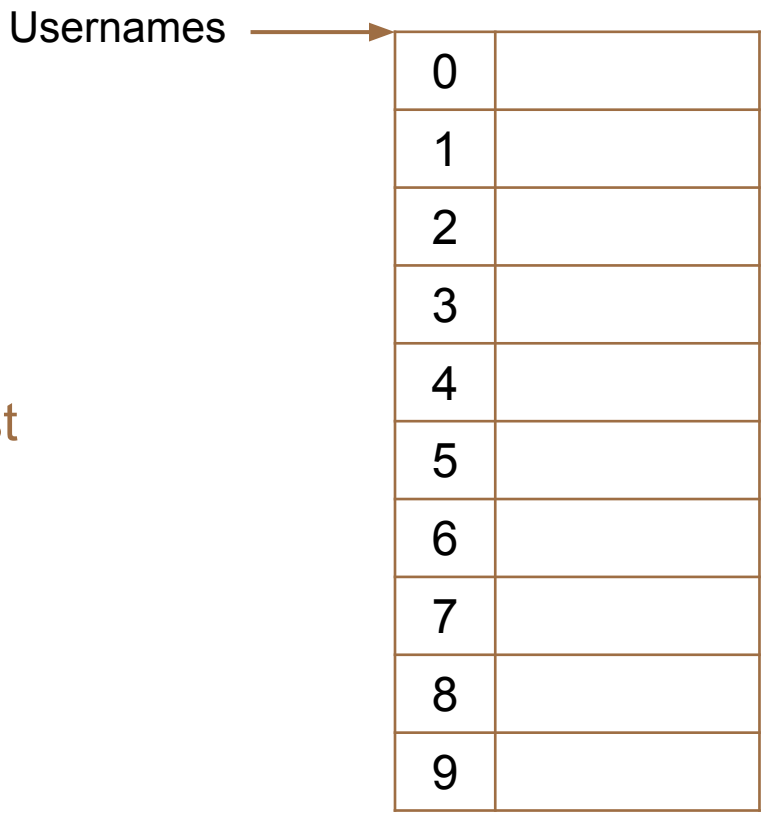
- One solution would be to create a variable for every player

```
playerName1 = "Olivia"
playerName2 = "Noah"
...
playerName500 = "Elijah"
```

- This would involve lots of copying and pasting of code and become a big problem with writing algorithms
  - Instead other data structures such as arrays are used

Arrays

- An array is a data structure that allows you to hold many items of data which is referenced by one identifier
  - All items of data in the array must be of the same data type
- An array of 10 strings called `userNames` could be declared as:
  - `array usernames[10]`



Arrays

- Information can be read from or added to the array using the index number
  - Indexes in most languages start at 0 not 1
- To add five usernames we would use:

```
array usernames[10]
usernames[0] = "psmith"
usernames[1] = "ltorvalds"
usernames[2] = "pwatts"
usernames[3] = "sjones"
usernames[4] = "mpatel"
```

Usernames

0	"psmith"
1	"ltorvalds"
2	"pwatts"
3	"sjones"
4	"mpatel"
5	
6	
7	
8	
9	

- How would you add five more usernames?
- How would you find an array's length?

Arrays

Answers

- How would you add five more usernames?

```
usernames[5] = "bwright"
usernames[6] = "mgreen"
usernames[7] = "dthomas"
usernames[8] = "nwhite"
usernames[9] = "fdavies"
```

- How would you find an array's length?

```
usernames.length //this returns 10
```

Usernames

0	"psmith"
1	"ltorvalds"
2	"pwatts"
3	"sjones"
4	"mpatel"
5	"bwright"
6	"mgreen"
7	"dthomas"
8	"nwhite"
9	"fdavies"

Arrays

- An array has a fixed length
  - We cannot add any more usernames to the array shown
  - To add more usernames, we would need to create a new larger array and copy the names across
  - Python programs typically use lists instead of arrays. It is possible to alter the length of lists
- How would you write a FOR loop to output all the usernames in the array?

Usernames

0	"psmith"
1	"ltorvalds"
2	"pwatts"
3	"sjones"
4	"mpatel"
5	"bwright"
6	"mgreen"
7	"dthomas"
8	"nwhite"
9	"fdavies"

Arrays

Answers

- How would you write a FOR loop to output all the usernames in the array?

```
for i=0 to usernames.length - 1
    print(username[i])
next i
```

- Adapt the code so that the user can enter a username and find out if it is present in the array
  - What type of search algorithm have you used?

Usernames

0	"psmith"
1	"ltorvalds"
2	"pwatts"
3	"sjones"
4	"mpatel"
5	"bwright"
6	"mgreen"
7	"dthomas"
8	"nwhite"
9	"fdavies"

Linear searching an array

Answers

- Adapt the code so that the user can enter a username and find out if it is present in the array

```
search = input("Type in username: ")
for i=0 to usernames.length - 1
    if usernames[i] == search
        print("Name found")
    endif
next i
```

- What type of search algorithm have you used?
  - This algorithm is a linear search

Worksheet 4

- Now complete **Task 1** on **Worksheet 4**

Advantages of using arrays

- Imagine that you want to input, sort and print 100 user names. It would be very inconvenient to use 100 different variable names
  - Using an array, you can write:

```
for i=0 to 99
    userNames[i] = input("Enter next name: ")
next i
(Sort the names)
for i=0 to 99
 print(UserName[i])
next i
```

  - What algorithm could be used to sort a list?

Linear search

- In a linear search, each item is examined in sequence until the item is found or the end of the list is reached
- How can this algorithm be made more efficient?

```
resort = ["Margate", "Swansea", "Brighton",
          "St Ives", "Cromer"]
for n = 0 to resort.length - 1
    if townName == resort[n] then
    found = True
    endif
next n
```

- Suggest a different loop and Boolean expression

Linear search

Answers

- The search should stop as soon as the item is found

```
resort = ["Margate", "Swansea", "Brighton",
          "St Ives", "Cromer"]
found = False
i = 0
while NOT found AND i < resort.length
    if townName == resort[i]  then
    found = True
    endif
    i = i + 1
endwhile
```

- How many times is the loop performed if you are searching for Brighton? Or Bournemouth?

The bubble sort

- Each item in a list is compared to the one next to it, and if it is greater, they swap places
- At the end of one pass through the list, the largest item is at the end of the list
- This is repeated until the items are sorted
- Suppose you have an array of five names to be sorted:

Sam	Ron	Tom	Bob	Mo

Bubble sort algorithm

```
names = ["Sam", "Ron", "Tom", "Bob", "Mo"]
numItems = len[names]
for i=0 to numItems - 2
for j=0 to numItems – i – 2
    if names[j] > names[j + 1] then
        swap the items
    endif
next j
next i
```

- How can you perform the part of the algorithm that says ‘swap the items’?

PG ONLINE

Swapping items in an array

Answers

```
names = [“Sam”, “Ron”, “Tom”, “Bob”, “Mo”]
numItems = len[names]
for i =  0 to numItems - 2
for j = 0 to numItems – i – 2
    if names[j] > names[j + 1] then
        temp = names[j]
        names[j] = names[j + 1]
        names[j + 1] = temp
    endif
next j
next i
```

Worksheet 4

- Now complete **Task 2** on **Worksheet 4**

Two-dimensional arrays

- Suppose you wanted to hold in a program, five marks for each of three students

	0	1	2	3	4	
Student1	12	14	8	7	17	0
Student2	6	8	5	3	13	1
Student3	16	15	9	10	18	2

- These marks could be put into a 2D array
- You can set individual items in the array like this:

```
mark[2,   4] =18
```

- What is the value of `mark[1,0]`?

Processing a 2D array

- Find and display the total mark for each student:

```
Student1    12   14   8   7   17
Student2        6    8   5   3   13
Student3    16   15   9   10  18
```

```
for s=0 to 2
    studentTotal = 0
    for m=0 to 4
        studentTotal = studentTotal + mark[s,m]
    next m
    print(studentTotal)
next s
```

Worksheet 4

- Now complete **Task 3** on **Worksheet 4**

Plenary

- Complete the following with the words beneath

  _____ allow many data items to be stored under one _____. Each item in the array can be accessed by its array _____. Indexes start at _____. By using a _____ we can easily loop through an array. Arrays have a _____ and all items in the array have the same _____.

zero **index** **FOR loop** **fixed length**

arrays **data type** **identifier**

Plenary

Answers

- Complete the following with the words beneath

Arrays allow many data items to be stored under one identifier. Each item in the array can be accessed by its array index. Indexes start at zero. By using a FOR loop we can easily loop through an array. Arrays have a fixed length and all items in the array have the same data type.

zero **index** **FOR loop** **fixed length**

arrays **data type** **identifier**

Copyright

© 2024 PG Online Limited

The contents of this unit are protected by copyright.

This unit and all the worksheets, PowerPoint presentations, teaching guides and other associated files distributed with it are supplied to you by PG Online Limited under licence and may be used and copied by you only in accordance with the terms of the licence. Except as expressly permitted by the licence, no part of the materials distributed with this unit may be used, reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic or otherwise, without the prior written permission of PG Online Limited.

Licence agreement

This is a legal agreement between you, the end user, and PG Online Limited. This unit and all the worksheets, PowerPoint presentations, teaching guides and other associated files distributed with it is licensed, not sold, to you by PG Online Limited for use under the terms of the licence.

The materials distributed with this unit may be freely copied and used by members of a single institution on a single site only. You are not permitted to share in any way any of the materials or part of the materials with any third party, including users on another site or individuals who are members of a separate institution. You acknowledge that the materials must remain with you, the licencing institution, and no part of the materials may be transferred to another institution. You also agree not to procure, authorise, encourage, facilitate or enable any third party to reproduce these materials in whole or in part without the prior permission of PG Online Limited.

OCR
Oxford Cambridge and RSA
An OCR endorsed teaching and learning tool

GCSE OCR

Computer Science
J277

5

Procedures and functions

Unit 7 Programming

PG ONLINE

Objectives

- Understand the concept of sub programs
- Learn how to write simple procedures and functions
- Understand and use parameters to pass data to procedures and functions
- Know that sub programs may use local variables which are accessible only within the sub program
- Use local variables and explain why it is good practice to do so
- Explain the advantages of using sub programs in programs

Starter

- A computer program needs a menu to appear
  - The following code is used within the program

```
print("menu")
print("1: New file")
print("2: Edit file")
print("3: Delete file")
print("4: Quit")
print("5: Show menu")

choice = input()
```

```
if choice == "5" then
    print("1: New file")
    print("2: Edit file")
    print("3: Delete file")
    print("4: Quit")
    print("5: Show menu")
endif
```

  - What is the problem with how the code is written?
  - What alternative could be used?

Starter

Answers

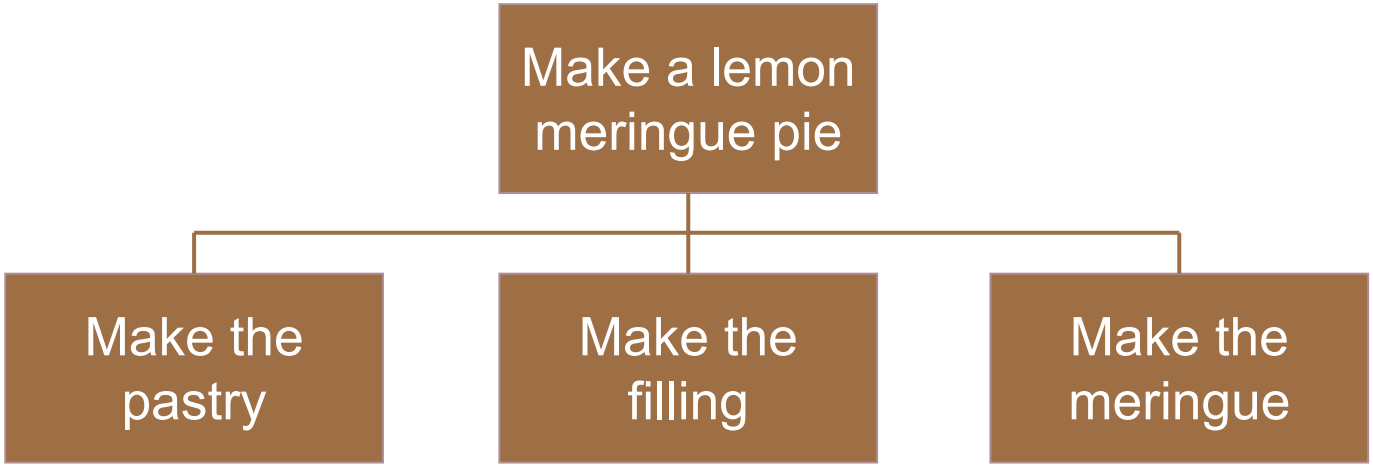
```
print("menu")
print("1: New file")
print("2: Edit file")
print("3: Delete file")
print("4: Quit")
print("5: Show menu")

choice = input()
```

```
if choice == "5" then
    print("1: New file")
    print("2: Edit file")
    print("3: Delete file")
    print("4: Quit")
    print("5: Show menu")
endif
```

- What is the problem with how the code is written?
  It has copied and pasted parts of the code which can lead to mistakes being introduced
- What alternative could be used?
  Sub programs – in this case a procedure

Using sub programs

- As your programs become larger and more complex, they need to be broken down into smaller, self-contained sections
- Here is an illustration of the concept:



Functions and procedures

- Sub programs allow code that you intend to use a number of time to be grouped together under one name
  - Both functions and procedures are sub programs
  - Values can be passed to both procedures and functions
  - Functions will return a value after processing has taken place

Built-in sub programs

- All programming languages have built-in procedures and functions to perform common tasks
  - You have already used built-in sub programs

```
print("Hello")
input("Type in your name: ")

random(1,6)
```

- For each of the above sub programs, explain whether they are functions or procedures

Built-in sub programs

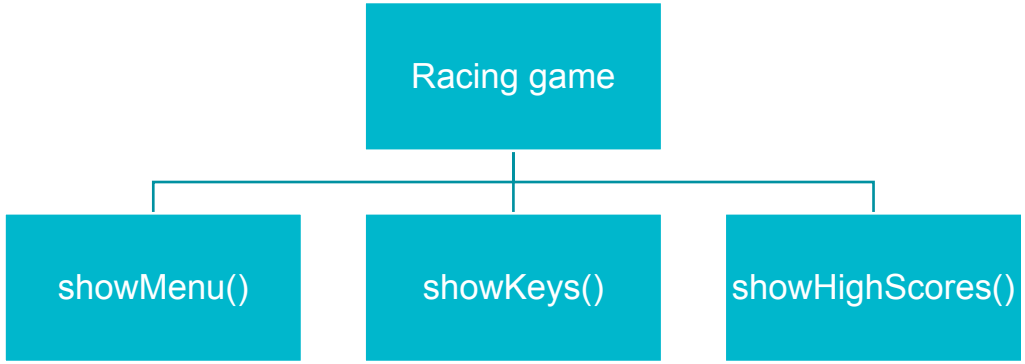
Answers

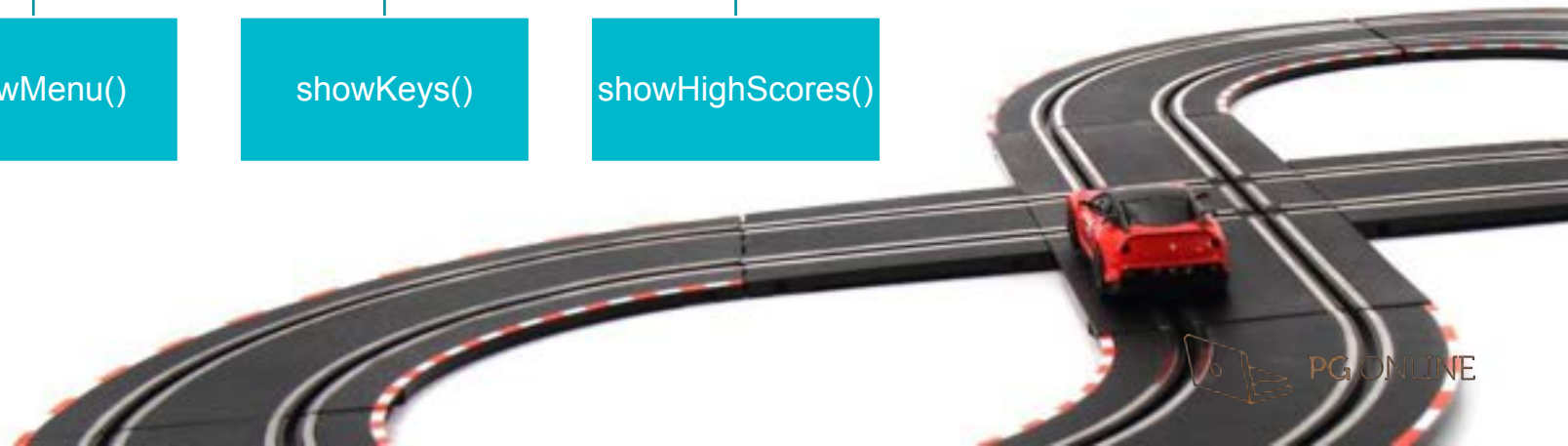
```
print("Hello")        //a procedure to output text

input("Type in your name: ")     //a function that returns
                      //what the user enters

random(1,6)      //a function that returns an
             //integer between 1 and 6
```

- Procedures execute code in a sub program – but they don't return anything
  - Functions will return something – so input returns what the user typed in, random returns a number
- Both functions and procedures can have arguments:
  - Arguments above are: "Hello", "Type in your name: ", 1, 6

A program using procedures

- A racing game requires the following:
  - Display a menu
  - Display the keyboard controls for the game
  - Display high scores in the game





PG ONLINE

showMenu procedure

```
procedure showMenu()
    print("      Menu ")
    print("==================")
    print("1: Play game")
    print("2: Show key controls")
    print("3: High scores")
    print("4: Return to main menu")
    print("5: Exit game")
endprocedure
showMenu()
```

This calls the procedure named showMenu

- How many parameters does showMenu have?
- Write a procedure that will show the keyboard controls of the game

showKeys procedure

Answers

```
procedure showKeys()
    print("Keyboard controls")
    print("=================")
    print("1:   Up arrow - forwards")
    print("2: Down arrow – backwards/brake")
    print("3: Left arrow – turn left")
    print("4: Right arrow – turn right")
    print("5: R - refuel")
endprocedure

showKeys()
```

- Both showKeys and showMenu have no parameters

Worksheet 5

- Now complete **Task 1** on **Worksheet 5**

Functions

- Functions work just like procedures, except at the end they will return a value

```
function sum(a, b)
 total = a + b
 return total
endfunction
answer = sum(5, 3)
```

- What is the value stored in `answer` once this program has been run?
  - Explain how the program works

Functions

Answers

```
function sum(a, b)
 total = a + b
 return total
endfunction

answer = sum(5, 3)
```

- First sum(5,3) calls the sum function with the arguments 5 and 3
- The sum function adds 5 + 3 and stores 8 in total
- The sum function returns the result of 8 to the function call
- This is then assigned to the variable called answer
- answer (which contains 8) is then output

- answer: 8
- Write a function named ‘greatest’ that has three parameters which are numbers
  - The function needs to return the greatest of the three numbers

Functions

Answers

```
function greatest(a, b, c)
    if a > b AND a > c then
        return a
    if b > a AND b > c then
        return b
    else
        return c
    endif
endfunction
```

Passing parameters

- When you create a function with the statement:

```
sum(a, b)
```

- a and b are known as **parameters**

Try this problem:

An average score is calculated by dividing a total score by the number of attempts

- Create a function to calculate an average score

Pseudocode for average

Answers

```
function averageScore(totalScore, numScores)
    avg = totalScore / numScores
    return avg
endfunction
```

- To call the function:

```
total = int(input("Enter total score: "))
n = int(input("Enter number of scores: "))
print(averageScore(total, n))
```

Worksheet 5

- Now complete **Task 2** and **Task 3** on **Worksheet 5**

Why use sub programs?

- They are useful to break up a large program into self-contained units
  - Each sub program can be tested separately to make sure it works correctly
  - Many programmers can work on a large program at the same time, cutting down development time
  - Sub programs can be re-used in other programs
- What is **one** other advantage of using sub programs?

Answers

More advantages of using sub programs

- Sub programs can be stored in a sub programs library and used in different programs if required
- Program maintenance is easier – if requirements change then just the affected sub programs need to be modified

Local variables and scope

- The “scope” of a variable, constant, procedure or function defines the parts of the program in which it is recognised and can be used
- When you use an inbuilt function or procedure, you can write, for example

```
x = sqrt(y)
```

- In the random function you don’t need to worry about the internal variables or how they are used
  - This is because the variables used are only recognised within the function itself

Variable scope

- Variables var1 and var2 are global variables and can be seen anywhere in the program
- Variables a, b and c can only be seen and used inside f1
- Variables x, y and z can only be seen and used inside f2

Variables var1, var2

```
function f1()
variables a, b, c
```

```
function func2()
variables x, y, z
```

Variable scope

- Think of each function as being a house with reflective windows – you can see out, but no one outside can see the variables in the “house”

Variable scope

Function A

Function B

Function C

- If you are in the main program, or in one of the functions, you cannot see any variables in the other functions
- If you are in a function “house”, you can see global variables in the main program, and you can use them but you can’t change them

Local and global variables

- Local variables only exist while the sub program is executing
  - They are only accessible within the sub program
- Global variables are accessible anywhere in the program
  - They are written in pseudocode using the global keyword:

    global score

What is printed by this program?

```
function changeNum()
    num = 6
    print(num)
endfunction
//Main program
num = 5
print(num)
//call function changenum
changeNum()
print(num)
```

This is a local variable

Prints 6

This doesn't have the global keyword so isn't affected by anything happening in changeNum

Prints 5

Calls changeNum

Prints 5 – the num variable in changeNum is different

Example

- Identify the local variables in function calcmax

```
function calcmax(heights)
max = heights[0]
n = heights.length
for count=0 to n – 1
    if heights[count] > max then
        max = heights[count]
    endif
next count
return max
endfunction
```

- What type of data structure is heights?
- How will the function be called?

Answer

Answers

- max, n and count are all local variables

```
function CalcMax(heights)    # heights is an array
max = heights[0]
n = len (heights)
for count=0 to n – 1
    if heights[count] > max then
        max = heights[count]
    endif
next count
return max
endfunction
```

- The function is called and the result stored with a statement such as: tallest = CalcMax (pupilHeights)
- The value stored in max will be returned and assigned to the variable tallest

Global and local constants

- Constants follow the same rules as variables
  - A constant declared in the main program can be seen and used, but not changed, anywhere else

```
function addVAT(netPrice)
    vat = netPrice * VAT_RATE
    priceWithVAT = netPrice + vat
    return priceWithVAT
endfunction

global const VAT_RATE = 0.20
priceNoVAT = 50.00
sellingPrice = addVAT(priceNoVAT)
print(sellingPrice)
```

Advantages of using local variables

- Local variables keep a sub program self-contained, so it can be used in any program without variable names conflicting with those used in the calling program
  - This also allows the same sub program to be used in other programs
  - Generally, you should use local variables wherever you possibly can

Worksheet 5

- Now complete **Task 4** on **Worksheet 5**

Plenary

- Answer the following questions in pairs
  - What are **two** types of sub program used in programming?
  - How many parameters can functions and procedures have?
  - Explain the difference between local and global variables
  - What are **three** reasons that you should try to use sub programs where possible?

Plenary

Answers

- Two types of sub program used in programming:
  - Functions and procedures
- How many parameters can functions and procedures have? – Zero, one or many
- Explain the difference between a local and global variables
  - Globals are accessible anywhere in the program – locals only within the program block in which they were created
- What are **three** reasons that you should try to use sub programs where possible?
  - Reuse code, decomposition, more maintainable code

Copyright

© 2024 PG Online Limited

The contents of this unit are protected by copyright.

This unit and all the worksheets, PowerPoint presentations, teaching guides and other associated files distributed with it are supplied to you by PG Online Limited under licence and may be used and copied by you only in accordance with the terms of the licence. Except as expressly permitted by the licence, no part of the materials distributed with this unit may be used, reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic or otherwise, without the prior written permission of PG Online Limited.

Licence agreement

This is a legal agreement between you, the end user, and PG Online Limited. This unit and all the worksheets, PowerPoint presentations, teaching guides and other associated files distributed with it is licensed, not sold, to you by PG Online Limited for use under the terms of the licence.

The materials distributed with this unit may be freely copied and used by members of a single institution on a single site only. You are not permitted to share in any way any of the materials or part of the materials with any third party, including users on another site or individuals who are members of a separate institution. You acknowledge that the materials must remain with you, the licencing institution, and no part of the materials may be transferred to another institution. You also agree not to procure, authorise, encourage, facilitate or enable any third party to reproduce these materials in whole or in part without the prior permission of PG Online Limited.

OCR
Oxford Cambridge and RSA
An OCR endorsed teaching and learning tool

GCSE OCR

Computer Science
J277

6

Records and files

Unit 7 Programming

PG ONLINE

Objectives

- Understand and use basic file handling operations:
  - open
  - read
  - write
  - close
- Read from and write to a text file
- Understand how records are used to store data and how a 2D array may be used to emulate a database table

Starter

- In a programming language, when we read from, or write to a file what **three** steps always need to be taken?

PG ONLINE

Starter

Answers

- Reading from and writing to files requires the following steps:
  - Open the file
  - Read from, or write to, the file
  - Close the file
- If the file doesn't yet exist it will need to be created first

Data structures

- A data structure is a collection of elementary data types such as integer, real, Boolean, char or string
- The last lesson described the array data structure
  - An array can only hold items of an identical data type – all integers, all strings, and so on
  - Some data structures, such as a Python list, are able to hold items with different data types

Records in a database

- Records are held in tables in a database
  - This example shows three records

FilmID	Title	YearReleased
F365	The Legend of Tarzan	2016
F380	The Magnificent Seven	2016
F250	Paddington	2015
F249	Strange Magic	2015

Storing records in a 2D array

- Alternatively, simple tables may be stored in a 2D array or 2D list (if there are multiple data types)

films

	0	1	2
0	F365	The Legend of Tarzan	2016
1	F380	The Magnificent Seven	2016
2	F250	Paddington	2015
3	F249	Strange Magic	2015

- What would the following store:

films[0,0] films[0,3] films[2,3]

Storing records in a 2D array

Answers

films

	0	1	2
0	F365	The Legend of Tarzan	2016
1	F380	The Magnificent Seven	2016
2	F250	Paddington	2015
3	F249	Strange Magic	2015

- What would the following store:

films[0,0]	films[0,3]	films[2,3]
F365	F249	2015

Text files

- A text file is able to store a collection of records
  - With text files, the only data type is “string”
- A file may have several text fields, typically separated by commas
  - Each record is placed on a new line

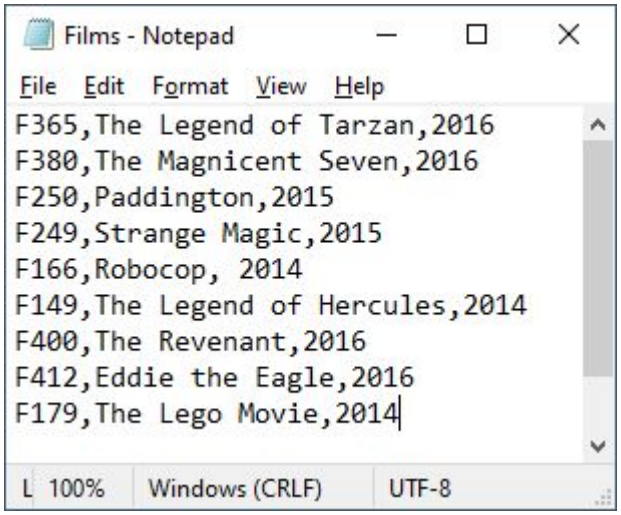
    F365, The Legend of Tarzan, 2016

    F380, The Magnificent Seven, 2016

    F250, Paddington, 2015
- This is known as a CSV (comma-separated values) file

Creating a CSV file

- You can create, read and print a CSV file with any text editor
  - Here is one created in Notepad:



  - This could also be saved as Films.csv and opened with either a text editor or spreadsheet

Worksheet 6

- Now complete **Task 1** on **Worksheet 6**

Creating a file

- To create a new file use the code:

```
newFile("names.txt")
```

- `newFile("names.txt")` creates a new file with the filename "names.txt"

Opening a file

- Programming languages have different syntax for writing to and reading from a file
- Typically, a statement to open the file will be required
- Many languages have three modes that files can be opened in:
  - Write mode – allows writing to a file
  - Append mode – allows more data to be added to a file
  - Read mode – only allows the file to be read
- The actual file name and path needs to be assigned to the identifier used for the file in the program

Reading from a text file

- Reading from a text file involves opening the file, processing each line then closing the file

```
marksFile = open("marks.txt")
while NOT markFile.endOfFile()
    line = marksFile.readLine()
    print(line)
endwhile
marksFile.close()
```

Open “marks.txt”

Loop until end of file

Read one line and store in variable *line*

Close “marks.txt”

- How do you think we can write to a file?

Writing data to a text file

- Here is an algorithm which writes to a text file:

```
marksFile = open("marks.txt")
name = input("Input student name: ")
mark = input("Input student mark: ")
marksFile.writeLine(name + "," + mark + "\n")
marksFile.close()
```

- If the user's name is "Sandra" and their mark is 85 the following will be added to the file:

  Sandra, 85

- "\n" is the newline character indicating then end of this record

Closing a file

- Once you have finished reading or writing to a file it is important to close it
  - This frees up any memory used by having it open
  - Once closed, the file cannot be read from or written to without opening it again
- To close a file, use:

```
myFile.close()
```

Worksheet 6

- Now complete **Task 2** on **Worksheet 6**

Plenary

- With a partner, answer the following questions:
  - What are the **three** steps required to read from, or write to, a file?
  - What is the meaning of the following:

```
while NOT myFile.endOfFile()
    print(myFile.readLine())
endwhile
```

PG ONLINE

Plenary

Answers

- What are the **three** steps required to read from, or write to, a file?
  - Open, Read/Write, Close
- What is the meaning of the following

```
while NOT myFile.endOfFile()
    print(myFile.readLine())
endwhile
```

  - Keep printing each line of a file until the end of the file is reached (`myFile.endOfFile()`)

Copyright

© 2024 PG Online Limited

The contents of this unit are protected by copyright.

This unit and all the worksheets, PowerPoint presentations, teaching guides and other associated files distributed with it are supplied to you by PG Online Limited under licence and may be used and copied by you only in accordance with the terms of the licence. Except as expressly permitted by the licence, no part of the materials distributed with this unit may be used, reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic or otherwise, without the prior written permission of PG Online Limited.

Licence agreement

This is a legal agreement between you, the end user, and PG Online Limited. This unit and all the worksheets, PowerPoint presentations, teaching guides and other associated files distributed with it is licensed, not sold, to you by PG Online Limited for use under the terms of the licence.

The materials distributed with this unit may be freely copied and used by members of a single institution on a single site only. You are not permitted to share in any way any of the materials or part of the materials with any third party, including users on another site or individuals who are members of a separate institution. You acknowledge that the materials must remain with you, the licencing institution, and no part of the materials may be transferred to another institution. You also agree not to procure, authorise, encourage, facilitate or enable any third party to reproduce these materials in whole or in part without the prior permission of PG Online Limited.

OCR
Oxford Cambridge and RSA
An OCR endorsed teaching and learning tool

GCSE OCR

Computer Science
J277

7

Introduction to SQL

Unit 7 Programming

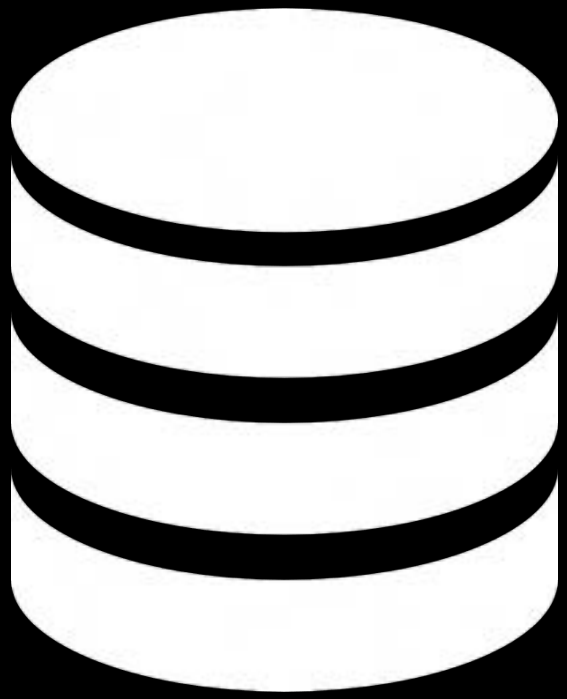
PG ONLINE

Objectives

- Use SQL (Structured Query Language) statements to search for data:
  - Formulate criteria involving AND, OR and LIKE
  - Use SELECT, FROM, WHERE, ORDER BY statements
  - Use the wildcard *

Starter

- In the previous topic you saw how data could be stored and accessed using a file
  - Where else can data be stored and accessed?
  - Give some examples from the real world that use such a system



Starter

Answers

- Data can be stored in a structured way in databases
- Examples include:
  - Social media content
  - Internet shopping sites – for products, prices, descriptions etc.
  - School student records
  - Tax records

What is SQL?

- SQL stands for Structured Query Language
- It is a language which allows you to create, query and add data to databases
  - SQL is used within most Database Management Systems, including MySQL, SQL Server and MS Access
  - SQL queries can be used within high level programming languages such as Python, Visual Basic or C#

Writing a query using SQL

- The SQL syntax for querying a database is:

  SELECT … (list the fields to be displayed)

  FROM … (specify the table name)

  WHERE … (list the search criteria)

The SELECT statement

MemberID	FirstName	Surname	Gender	Town
1	David	Johnson	M	Ipswich
2	Christine	Bates	F	Woodbridge
3	Jasmine	Hamid	F	Ipswich
4	Peter	Okello	M	Colchester
5	Stephen	Hines	M	Woodbridge

- The table above is named members
  - The following SQL statement will select all the records and fields from the table

```
SELECT MemberID, FirstName, Surname, Gender, Town
FROM members
```

Using a wild card

MemberID	FirstName	Surname	Gender	Town
1	David	Johnson	M	Ipswich
2	Christine	Bates	F	Woodbridge
3	Jasmine	Hamid	F	Ipswich
4	Peter	Okello	M	Colchester
5	Stephen	Hines	M	Woodbridge

- You can use a “wild card” * to mean “all columns”

```
SELECT *
FROM members
```

- You can select specific columns in a query:

```
SELECT FirstName, Surname
FROM members
```

The WHERE clause

MemberID	FirstName	Surname	Gender	Town
1	David	Johnson	M	Ipswich
2	Christine	Bates	F	Woodbridge
3	Jasmine	Hamid	F	Ipswich
4	Peter	Okello	M	Colchester
5	Stephen	Hines	M	Woodbridge

- The WHERE clause is used to select only records satisfying a specified condition:

```
SELECT FirstName, Surname
FROM members
WHERE Town = 'Ipswich'
```

- What is the result of this query?

The WHERE clause

Answers

MemberID	FirstName	Surname	Gender	Town
1	David	Johnson	M	Ipswich
2	Christine	Bates	F	Woodbridge
3	Jasmine	Hamid	F	Ipswich
4	Peter	Okello	M	Colchester
5	Stephen	Hines	M	Woodbridge

```
SELECT FirstName, Surname
FROM members
WHERE Town = 'Ipswich'
```

- Query result:

FirstName	Surname
David	Johnson
Jasmine	Hamid

Operators in the WHERE clause

- You have already used some of these operators in queries:

= != > < >= <= AND, OR, NOT

- You can use these operators in SQL queries:

```
SELECT *
FROM members
WHERE Town = 'Colchester' OR Town = 'IPSWICH'

SELECT FirstName, Surname
FROM members
WHERE (Town = 'Colchester' OR Town = 'IPSWICH') AND
Gender = 'F'
```

Operators in the WHERE clause

- You can also use the following in SQL queries:

BETWEEN between an inclusive range

LIKE search for a pattern

For example:

```
SELECT *
FROM members
WHERE Surname LIKE 'H*'
```

This selects all members whose surname begins with H

Note the wildcard * in the LIKE string which acts as a substitute for zero, one or more characters

SELECT queries

- Look at the following table named “Dogs”

DogID	Name	Breed	Colour	Gender	Age
1	Coco	Labrador	Brown	M	3
2	Milly	Spaniel	Black	F	5
3	Sasha	Retriever	Golden	F	4
4	Mark	Labrador	Black	M	3
5	Marlee	Retriever	Golden	F	2
6	Alfie	Spaniel	Brown	M	6
7	Georgie	Labrador	Brown	M	4

- Write SQL queries to find:
  - The name and breed all female dogs
  - All fields for the dogs that are four or older

SELECT queries

Answers

- Dogs table

DogID	Name	Breed	Colour	Gender	Age
1	Coco	Labrador	Brown	M	3
2	Milly	Spaniel	Black	F	5
3	Sasha	Retriever	Golden	F	4
4	Mark	Labrador	Black	M	3
5	Marlee	Retriever	Golden	F	2
6	Alfie	Spaniel	Brown	M	6
7	Georgie	Labrador	Brown	M	4

- The name and breed of all female dogs
  - SELECT Name, Breed FROM Dogs WHERE Gender='F’
- All fields for the dogs that are four or older
  - SELECT * FROM Dogs WHERE Age >= 4

Worksheet 7

- Now complete **Task 1** on **Worksheet 7**

Sorting: the ORDER BY keyword

- ORDER BY allows a query to sort data by ascending or descending order
- For **ascending** order

```
SELECT * FROM members
ORDER BY Surname ASC
```

- For **descending** order

```
SELECT * FROM members
ORDER BY Surname DESC
```

Sorting

MemberID	FirstName	Surname	Gender	Town
1	David	Johnson	M	Ipswich
2	Christine	Bates	F	Woodbridge
3	Jasmine	Hamid	F	Ipswich
4	Peter	Okello	M	Colchester
5	Stephen	Hines	M	Woodbridge

- What will the result be for the following query?

```
SELECT FirstName, Surname
FROM members
ORDER BY Town ASC
```

Result of sorting

Answers

- Result after execution of the statement:

```
SELECT FirstName, Surname, Town
FROM members
ORDER BY Town ASC
```

FirstName	Surname	Town
Peter	Okello	Colchester
David	Johnson	Ipswich
Jasmine	Hamid	Ipswich
Christine	Bates	Woodbridge
Stephen	Hines	Woodbridge

Worksheet 7

- Now complete **Task 2** on **Worksheet 7**

Plenary

- Look at the table called Animals on the right
- In pairs create queries for the following:
  - All animal names in alphabetical order
  - The names and weights for all animals that are over 1000 kg
  - All fields for the animals that are over 2 m

Animals

Animal	Height_m	Weight_kg
Rhino	1.8	2000
Giraffe	5.5	1800
Emu	1.8	55
Llama	1.7	200
Sealion	2.4	360

Plenary

Answers

- All animal names in alphabetical order
  - SELECT Animal FROM Animals ORDER BY Animal ASC
- The names and weights for all animals that are over 1000 kg
  - SELECT Animal, Weight_kg FROM Animals WHERE Weight_kg > 1000
- All fields for the animals that are over 2 m
  - SELECT * FROM Animal WHERE Height_m > 2

Animals

Animal	Height_m	Weight_kg
Rhino	1.8	2000
Giraffe	5.5	1800
Emu	1.8	55
Llama	1.7	200
Sealion	2.4	360

Copyright

© 2024 PG Online Limited

The contents of this unit are protected by copyright.

This unit and all the worksheets, PowerPoint presentations, teaching guides and other associated files distributed with it are supplied to you by PG Online Limited under licence and may be used and copied by you only in accordance with the terms of the licence. Except as expressly permitted by the licence, no part of the materials distributed with this unit may be used, reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic or otherwise, without the prior written permission of PG Online Limited.

Licence agreement

This is a legal agreement between you, the end user, and PG Online Limited. This unit and all the worksheets, PowerPoint presentations, teaching guides and other associated files distributed with it is licensed, not sold, to you by PG Online Limited for use under the terms of the licence.

The materials distributed with this unit may be freely copied and used by members of a single institution on a single site only. You are not permitted to share in any way any of the materials or part of the materials with any third party, including users on another site or individuals who are members of a separate institution. You acknowledge that the materials must remain with you, the licencing institution, and no part of the materials may be transferred to another institution. You also agree not to procure, authorise, encourage, facilitate or enable any third party to reproduce these materials in whole or in part without the prior permission of PG Online Limited.

OCR
Oxford Cambridge and RSA
An OCR endorsed teaching and learning tool

GCSE OCR

Computer Science
J277

Logic diagrams and truth tables

Unit 8
Logic and languages

PG ONLINE

1

Objectives

- Construct truth tables for the following logic gates:
  - NOT
  - AND
  - OR
- Construct truth tables for simple logic circuits
- Interpret the results of truth tables
- Create, modify and interpret simple logic circuit diagrams

Starter

- Consider a safe with two keys
- If both keys are used, the safe will open

```
if key1 AND key2 then
    safeOpen = True
else
    safeOpen = False
endif
```

- What are the only possible values that key1 and key2 can be?
  - What values must they be to open the safe?

Starter

Answers

```
if key1 AND key2 then
safeOpen = True

else
    safeOpen = False
endif
```

- What are the only possible values that key1 and key2 can be?
  - Key1 and key2 can only be True or False (lock or unlock)
- What values must they be to open the safe?
  - They must both be True to open the safe

Binary situations

- Binary situations are common in daily life and can refer to things that can be in only one of two states:
  - Stop or Go
  - Pass or Fail
  - On or Off
- In computing terms
  - A binary 1 can represent True
  - A binary 0 can represent False

Booleans in loops

- Booleans are used in IF statements and loops

```
while doorOpen
turnOnLight()
endwhile
```

True or False?

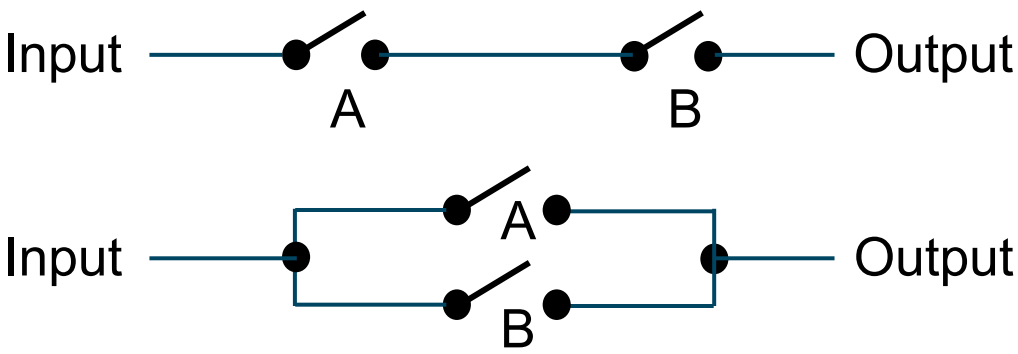
Unlocking a smartphone:

- using a known fingerprint **or**
- using a correct PIN code?

```
if fingerprint OR correctPIN then
    unlock = True
else
    unlock = False
endif
```

Boolean functions

- AND, OR and NOT are Boolean operators
- A computer can calculate the results of **A AND B**, **A OR B**, or **NOT A**
  - What do these two circuit diagrams represent?

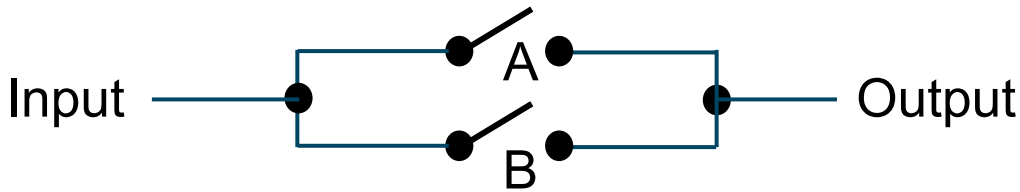


Boolean functions

Answers

- What do these two circuit diagrams represent?

AND
Both switches must be down to complete the circuit



OR
Either switch needs to be down to complete the circuit

Truth tables

- A **truth table** shows the output from all possible combinations of inputs from a Boolean expression
- If there are two inputs A AND B, there are four possible combinations of TRUE and FALSE
  - Both A and B must be True for the output to be True

Input A	Input B	A AND B
False	False	False
False	True	False
True	False	False
True	True	True

Boolean function – AND

- It is also possible to use 1 and 0 to represent True and False
  - The below truth tables represent: IF a lift is on the correct floor **AND** the call button has been pressed THEN open the lift door
  - What are the values for *Open Lift Door* and *P* below

Lift on correct floor	Button Pressed	Open Lift Door
False	False	?
False	True	?
True	False	?
True	True	?

A	B	P
0	0	?
0	1	?
1	0	?
1	1	?

Truth table – AND

Answers

- IF both inputs are **TRUE** then the output is **TRUE**

Lift on correct floor	Button Pressed	Open Lift Door
0	0	0
0	1	0
1	0	0
1	1	1

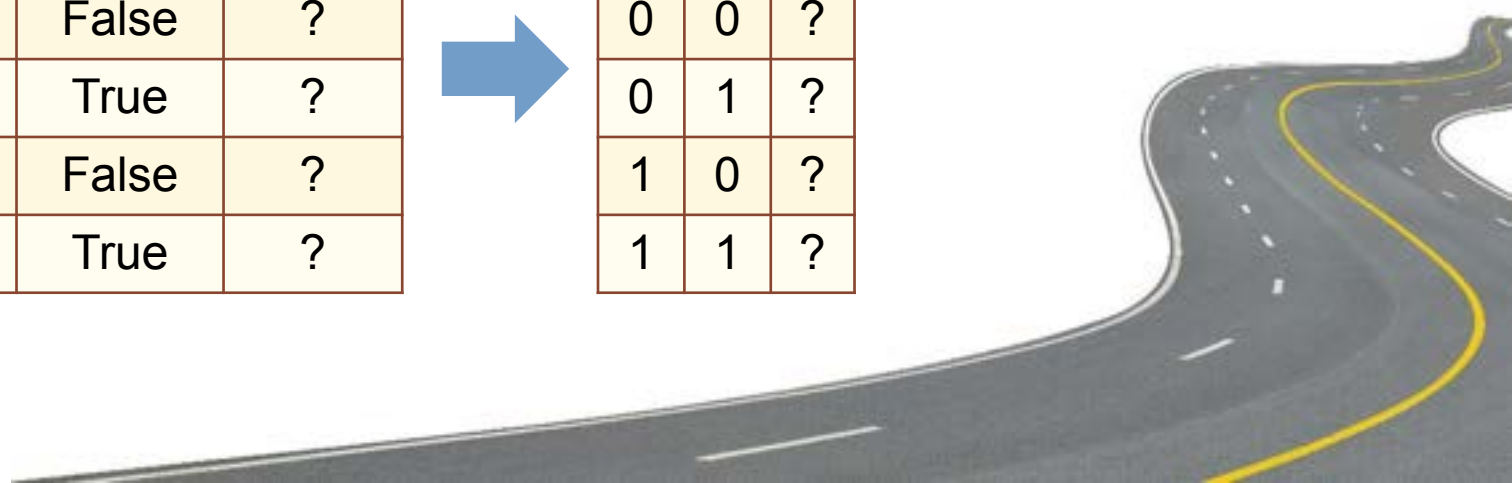
A	B	P
0	0	0
0	1	0
1	0	0
1	1	1

Boolean function – OR

- IF lane1Open OR lane2Open
  - If lane one is open OR lane 2 is open then you can drive down the road
  - What is P?

lane1 Open	Lane2 Open	P
False	False	?
False	True	?
True	False	?
True	True	?

A	B	P
0	0	?
0	1	?
1	0	?
1	1	?



Boolean function – OR

Answers

- IF lane1Open OR lane2Open
  - If lane one is open OR lane 2 is open then you can drive down the road
  - What is P?

lane1 Open	Lane2 Open	P
False	False	False
False	True	True
True	False	True
True	True	True

A	B	P
0	0	0
0	1	1
1	0	1
1	1	1

PG ONLINE

Boolean function – NOT

- IF the input is **True** then the output is **False**
- IF the input is **False** then the output is **True**

Input A	Output P = NOT A
False	True
True	False

More logic

Which of the following statements are **True?**

- (4 > 3) AND (5 > 7)
- (2 < 8) OR (8 > 10)
- NOT (5 * 7 > 30)
- ((7 DIV 3) >= 2) OR ((7 DIV 3) < 2)
- ((12 MOD 5) < 2) AND ((12 MOD 5) == 2)
  - DIV gives the integer division (the result of division without any fractional component)
  - MOD gives the remainder after a division

More logic

Answers

Which of the following statements are **True?**

- (4 > 3) AND (5 > 7) False
- (2 < 8) OR (8 > 10) True
- NOT (5 * 7 > 30) False
- ((7 DIV 3) >= 2) OR ((7 DIV 3) < 2) True
- ((12 MOD 5) < 2) AND ((12 MOD 5) == 2) False

Worksheet 1

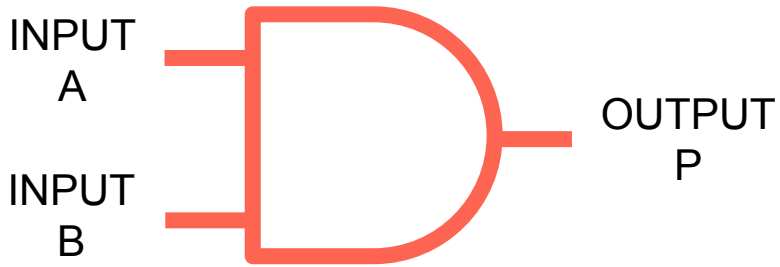
- Now complete **Task 1** on **Worksheet 1**

From truth tables to logic gates

- Physical computer circuits are built using logic gates
- The first three fundamental gates used to build circuits are:
  - AND gate
  - OR gate
  - NOT gate

Binary logic – AND gate

- If both inputs are 1 (True) then the output is 1 (True)
- Otherwise the output is 0 (False)



Logic statement: P = A AND B

Logic Diagram

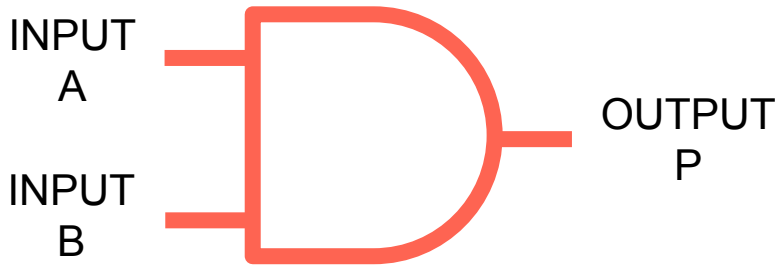
A	B	P
0	0	?
0	1	?
1	0	?
1	1	?

Truth Table

Binary logic – AND gate

Answers

- If both inputs are 1 (True) then the output is 1 (True)
- Otherwise the output is 0 (False)



Logic statement: P = A AND B

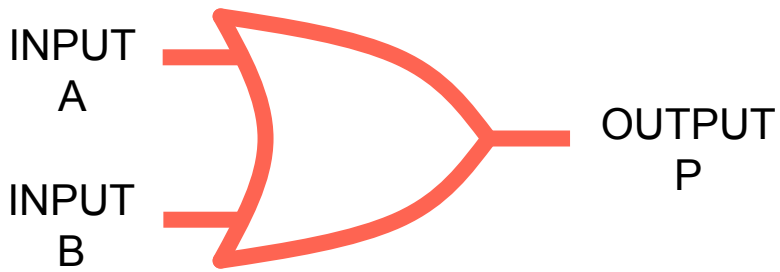
Logic Diagram

A	B	P
0	0	0
0	1	0
1	0	0
1	1	1

Truth Table

Binary logic – OR gate

- If either input is 1 (True) then the output is 1 (True)
- Otherwise the output is 0 (False)



Logic statement: P = A OR B

Logic Diagram

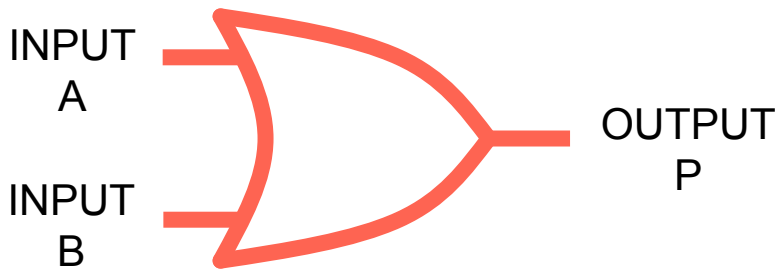
A	B	P
0	0	?
0	1	?
1	0	?
1	1	?

Truth Table

Binary logic – OR gate

Answers

- If either input is 1 (True) then the output is 1 (True)
- Otherwise the output is 0 (False)



Logic statement: P = A OR B

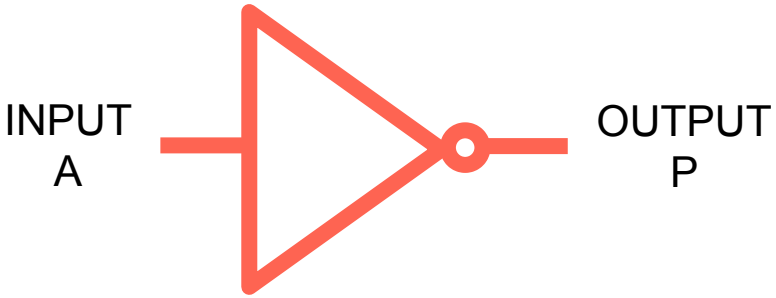
Logic Diagram

A	B	P
0	0	0
0	1	1
1	0	1
1	1	1

Truth Table

Binary logic – NOT gate

- If 0 is input it outputs 1 (True)
- If 1 is input it outputs 0 (False)



Logic statement: P = NOT A

Logic Diagram

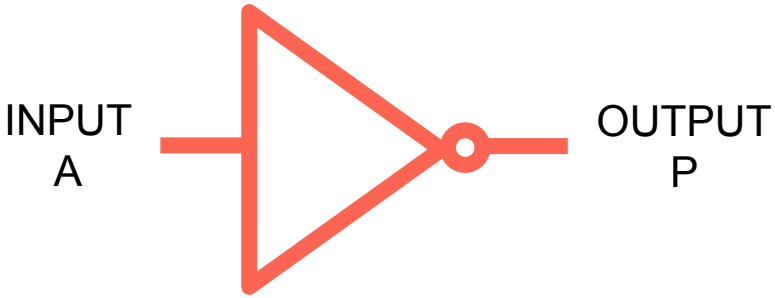
A	P
0	?
1	?

Truth Table

Binary logic – NOT gate

Answers

- If 0 is input it outputs 1 (True)
- If 1 is input it outputs 0 (False)



Logic statement: P = NOT A

A	P
0	1
1	0

Logic Diagram

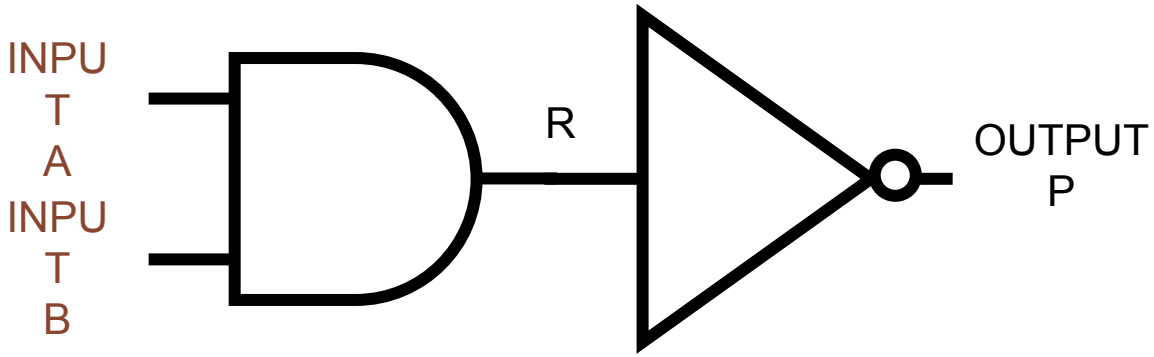
Truth Table

Worksheet 1

- Now complete **Task 2** on **Worksheet 1**

Combining logic gates

- We can combine logic gates together to make more complex circuits



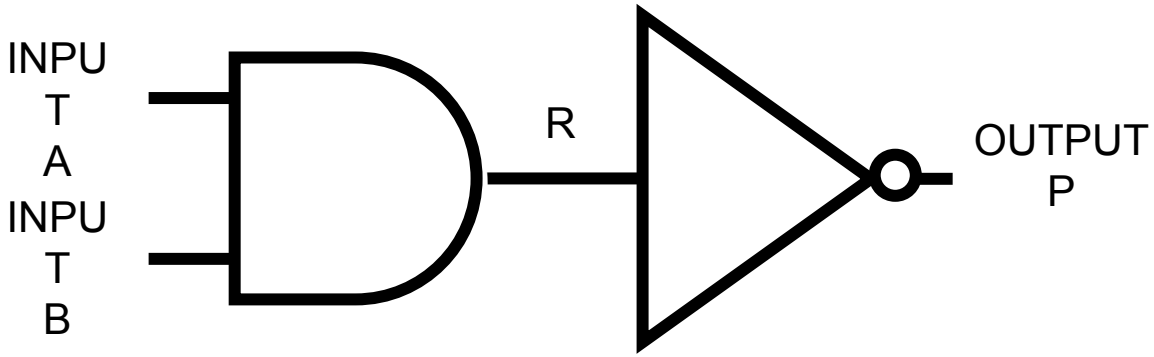
A	B	R = A AND B	P = NOT R
0	0		
0	1		
1	0		
1	1		

Logic statement:
P = NOT(A AND B)

PG ONLINE

Combining logic gates

- We can combine logic gates together to make more complex circuits



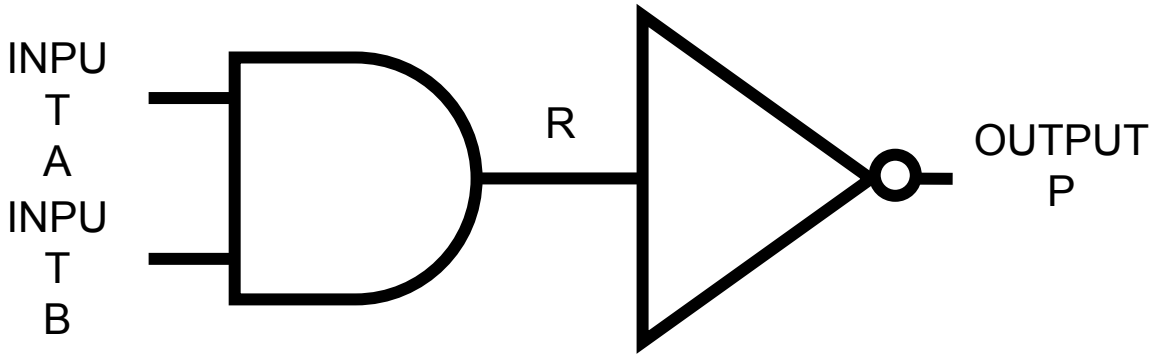
A	B	R = A AND B	P = NOT R
0	0	0	
0	1	0	
1	0	0	
1	1	1	

Logic statement:
P = NOT (A AND B)

PG ONLINE

Combining logic gates

- We can combine logic gates together to make more complex circuits



A	B	R = A AND B	P = NOT R
0	0	0	1
0	1	0	1
1	0	0	1
1	1	1	0

Logic statement:
P = NOT (A AND B)

PG ONLINE

Modelling real life – security lighting

- What would the circuit diagram for this situation look like?
  - The security light must come on if it senses movement AND it is night time, OR if someone presses a manual override switch

Modelling real life

Answers

Sensor detects movement = TRUE

AND

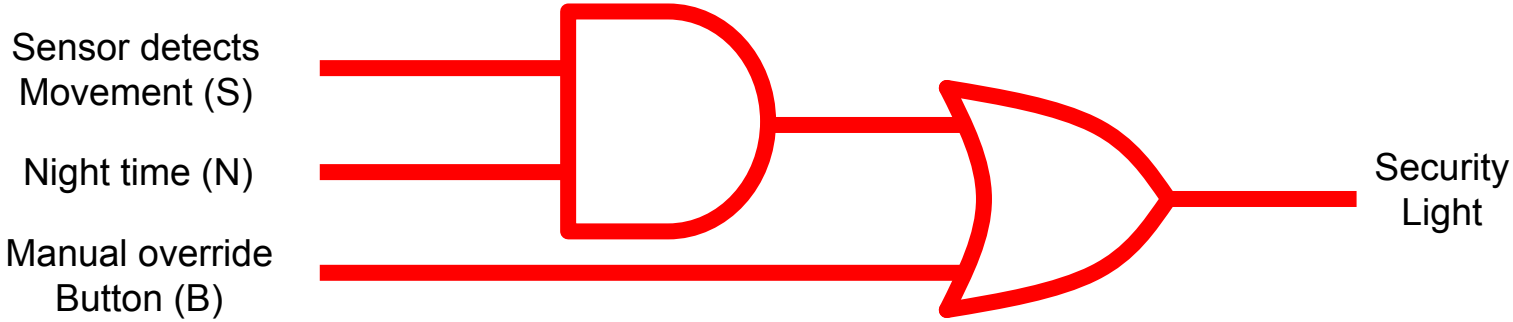
Night time = TRUE

OR

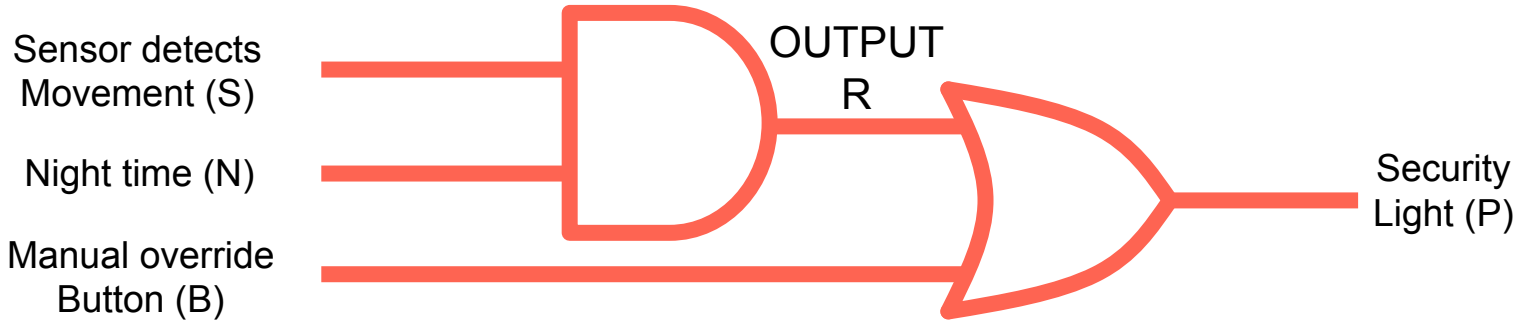
Manual override Button = TRUE

=

Security Light On



The truth table

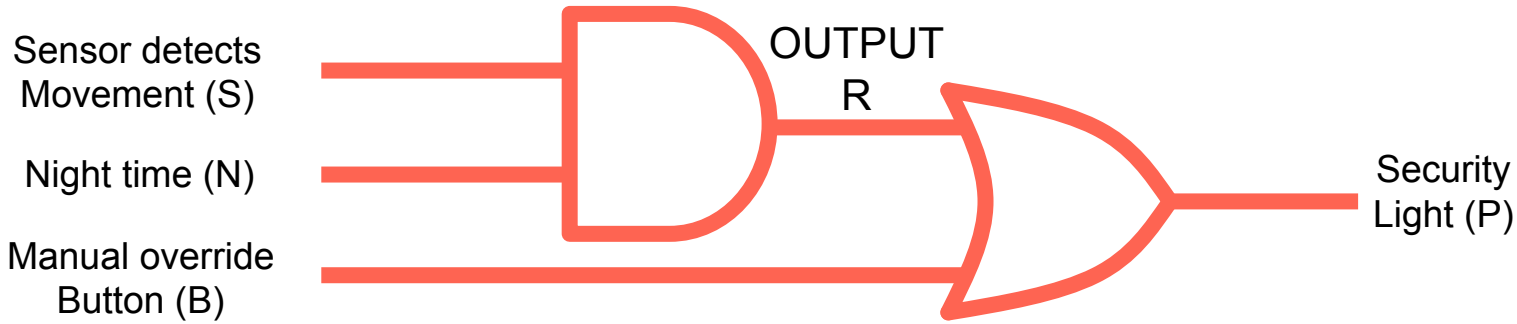


S	N	B	R = S AND N	P = R OR B

Logic statement:
P = (S AND N) OR B

PG ONLINE

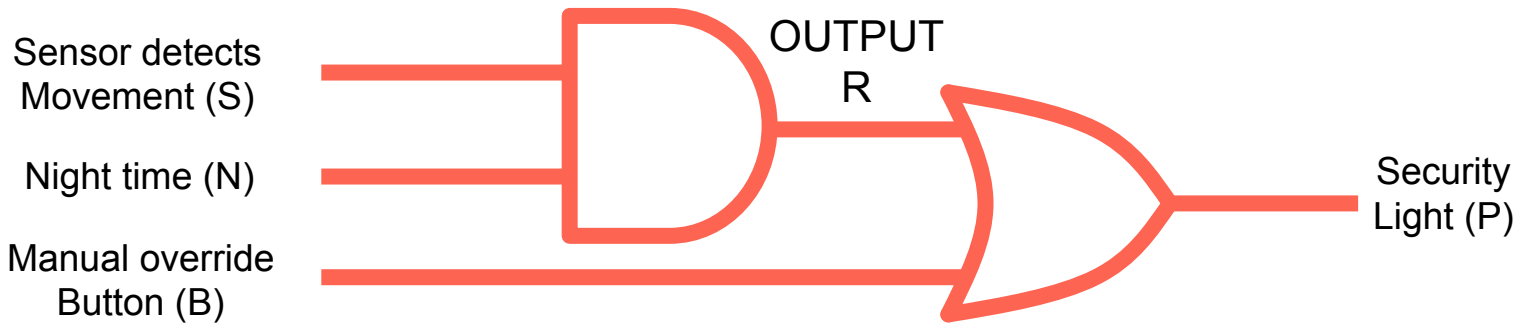
The truth table



S	N	B	R = S AND N	P = R OR B
0	0	0		
0	0	1		
0	1	0		
0	1	1		
1	0	0		
1	0	1		
1	1	0		
1	1	1		

Logic statement:
P = (S AND N) OR B

The truth table

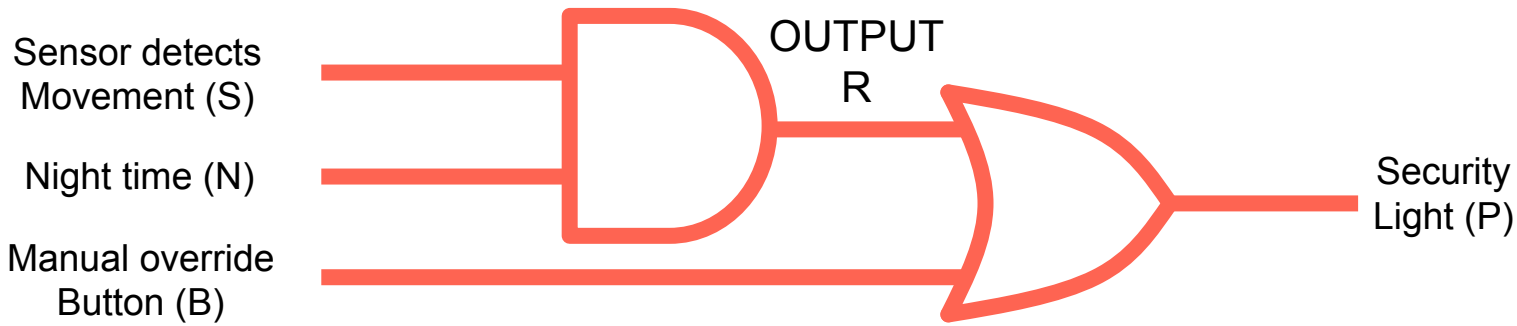


S	N	B	R = S AND N	P = R OR B
0	0	0	0	
0	0	1	0	
0	1	0	0	
0	1	1	0	
1	0	0	0	
1	0	1	0	
1	1	0	1	
1	1	1	1	

Logic statement:
P = (S AND N) OR B

PG ONLINE

The truth table



S	N	B	R = S AND N	P = R OR B
0	0	0	0	0
0	0	1	0	1
0	1	0	0	0
0	1	1	0	1
1	0	0	0	0
1	0	1	0	1
1	1	0	1	1
1	1	1	1	1

Logic statement:
P = (S AND N) OR B

PG ONLINE

Worksheet 1

- Now complete **Task 3** on **Worksheet 1**

Plenary

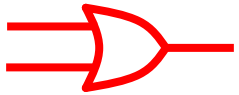
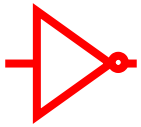
- Answer the following questions with a partner:
  1. What are the three basic logic gates?
  2. What are the symbols for each of the logic gates?
  3. Explain what each logic gate does
  4. What is a truth table?

PG ONLINE

Plenary

Answers

1. What are the three basic logic gates?
   - AND, OR, NOT
2. What are the symbols for each of the logic gates?
   - AND

NOT 

3. Explain what each logic gate does
   - AND – Both inputs must be True for output to be True
   - OR – At least one input must be True for output to be True
   - NOT – The output is opposite to the input
4. What is a truth table?
   - It shows all possible combinations of inputs and the outputs they create

PG ONLINE

Copyright

© 2024 PG Online Limited

The contents of this unit are protected by copyright.

This unit and all the worksheets, PowerPoint presentations, teaching guides and other associated files distributed with it are supplied to you by PG Online Limited under licence and may be used and copied by you only in accordance with the terms of the licence. Except as expressly permitted by the licence, no part of the materials distributed with this unit may be used, reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic or otherwise, without the prior written permission of PG Online Limited.

Licence agreement

This is a legal agreement between you, the end user, and PG Online Limited. This unit and all the worksheets, PowerPoint presentations, teaching guides and other associated files distributed with it is licensed, not sold, to you by PG Online Limited for use under the terms of the licence.

The materials distributed with this unit may be freely copied and used by members of a single institution on a single site only. You are not permitted to share in any way any of the materials or part of the materials with any third party, including users on another site or individuals who are members of a separate institution. You acknowledge that the materials must remain with you, the licencing institution, and no part of the materials may be transferred to another institution. You also agree not to procure, authorise, encourage, facilitate or enable any third party to reproduce these materials in whole or in part without the prior permission of PG Online Limited.

OCR
Oxford Cambridge and RSA
An OCR endorsed teaching and learning tool

GCSE OCR

Computer Science
J277

Defensive design

Unit 8
Logic and languages

PG ONLINE

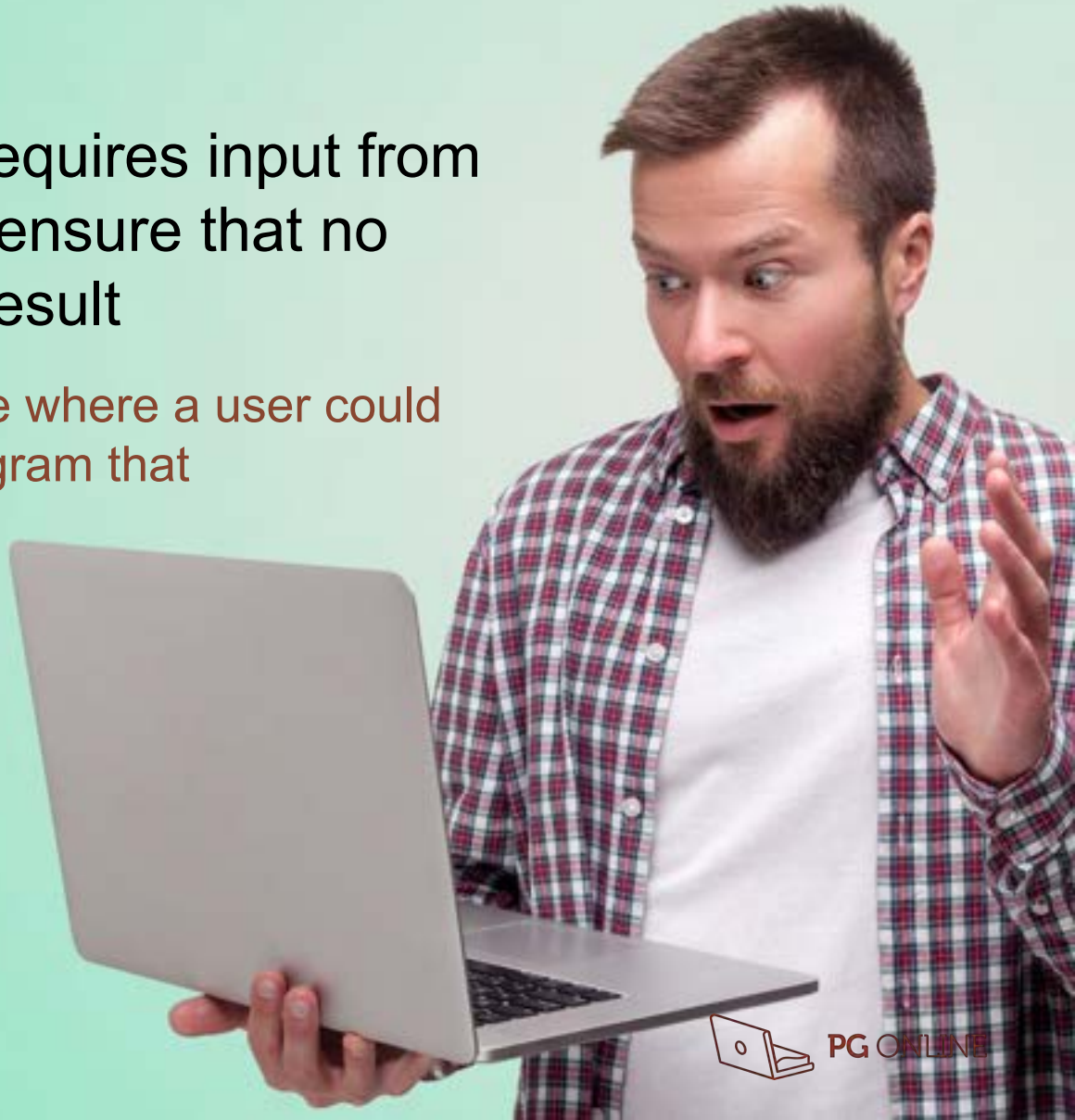
2

Objectives

- Describe defensive design considerations:
  - Input validation
  - Anticipating misuse
  - Authentication
- Understand how to make maintainable programs including:
  - The use of sub programs
  - Naming conventions
  - Indentation
  - Commenting

Starter

- When a program requires input from a user, it needs to ensure that no errors occur as a result
  - What is an example where a user could enter data to a program that causes it to crash?



PG ONLINE

Starter

Answers

- What is an example where a user could enter data to a program that causes it to crash?

```
age = input("Please enter your age: ")
age = age + 1
```

Results in a crash as an input is a string and it is trying to do a calculation with it

Data validation

- Data validation routines can ensure that data entered is of the right type – for example, an integer
  - Validation cannot ensure that the user has not entered a wrong value, or made a spelling mistake in a name
  - It can only ensure that the data is reasonable and conforms to a set of rules
- What other validation checks could you apply to data entered by the user?

Types of validation check

Answers

Check	Example
Range check	A number or date is within a sensible/allowed range
Type check	Data is of the right type, such as integer, letter or text
Length check	Text entered is not too long or too short – for example, a password is between 8 and 15 characters
Presence check	Checks that data has been entered, i.e. the field has not been left blank
Format check	Checks that the format of, for example, a postcode or email address is correct

Example

- What sort of validation check is made in this algorithm?

```
postcode = input("Please enter postcode:")
if postcode.length < 6 OR
   postcode.length > 8 then
print("Invalid postcode")
endif
```

- Rewrite the algorithm so that the algorithm keeps asking the user to enter a postcode until the entry is valid

Solution

Answers

- You need a WHILE or a REPEAT loop

```
postcode = input("Please enter postcode:")
while postcode.length < 6 OR
      postcode.length > 8
    postcode = input("Invalid postcode –
          please re-enter")
endwhile
```

- How many times will the loop be performed if the user enters IP6 4DF?

Verification

- Validation can only check that the data entered is reasonable
- Verification is used to double-check that the data has been typed in accurately
- For example, a user setting a new password may be asked to type it in twice
  - If the two passwords don't match, they will be asked to enter the password again
  - This is known as double-entry verification

Worksheet 2

- Now complete **Task 1** on **Worksheet 2**

Authentication routines

- Authentication routines are used to make sure a person is who they claim to be
  - What is a common method of online authentication, for example when you log in to a website with which you have previously registered?

Password routines

Answers

- Commonly, you are asked to enter a user ID and a password
  - Once you have entered the user ID, the website looks up your password in a database
  - If the user ID cannot be found, an error message is displayed
- What happens if you enter the wrong password?

Entering a password

- You usually get three attempts to get your password, and then you will be locked out

Anticipating misuse

- Why are you often only allowed a finite number of tries before being locked out?

Three tries and you're out!

- It may be that you have forgotten your password, and you need to be given a reminder, so three tries is enough for the average user
- BUT a hacker may be trying out dozens of likely passwords to try and get the correct one
  - There are software programs which will try out every combination of letters, numbers and special characters – this is known as a brute-force attack
  - Use a password of 8 characters or more to make it more difficult to hack!

Worksheet 2

- Now complete **Task 2** on **Worksheet 2**

Maintainable programs

- Programs need to be maintained
  - This will be to improve the code, fix bugs or add new features to the program
  - It may be carried out by the original programmer or different programmers
- It is important that programs are written in a way to make them easily maintainable. This includes:
  - The use of sub programs (functions and procedures)
  - Using appropriate naming conventions
  - Indentation
  - Commenting

Using sub programs

- Sub programs include functions and procedures
  - Well written sub programs will take inputs (through parameters) and if necessary return a value
  - They should be written so that they can be reused multiple times in the program or by other programs
  - The two programs below are for a function that works out the area of a circle. Which is more easy to reuse and will help create maintainable code?

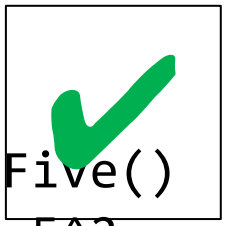
```
function circle(radius)
  area = 3.14 * radius^2
  return area
endfunction

function circleFive()
  area = 3.14 * 5^2
```

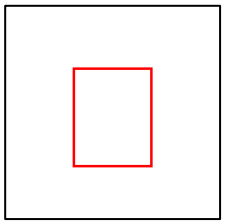
Using sub programs

Answers

- The program on the left takes any sized radius as an input
  - This means that it is reusable many times in the program and in other programs
  - This will make a larger program easier to maintain as there will be just one function to calculate the area of a circle

```
function circle(radius)
  area = 3.14 * radius^2
  return area
endfunction

function circleFive()
  area = 3.14 * 5^2
  return area
endfunction
```





Naming conventions

- The two programs below are the same algorithm
  - Which is easier to understand? Why?

```
a = float(input())
b = float(input())
c = float(input())
d = a + b + c
e = d / 3
print(e)
```

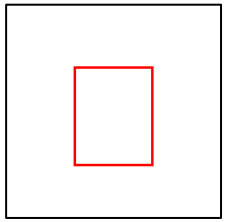
```
num1 = float(input())
num2 = float(input())
num3 = float(input())
total = num1 + num2 + num3
average = total / 3
print(average)
```

Naming conventions

Answers

- The two programs below are the same algorithm
  - Which is easier to understand? Why?

```
a = float(input())
b = float(input())
c = float(input())
d = a + b + c
e = d / 3
print(e)
```



```
num1 = float(input())
num2 = float(input())
num3 = float(input())
total = num1 + num2 + num3
average = total / 3
print(average)
```

- It is important to use meaningful names for variables, constants, functions and procedures
  - This makes code easier to read and understand

Indentation

- Look at the following pseudocode:

```
tables = 12
rows = 12
for i=1 to tables
for j=1 to rows
answer = i * j
next j
next i
print(str(i) + "x" + str(j) + " = " + str(answer))
```

- Why would the use of indentation improve it?
  - Why may indentation be essential in some languages?

Indentation

Answers

```
tables = 12
rows = 12
for i=1 to tables
    for j=1 to rows
         answer = i * j
    next j
next i
print(str(i) + "x" + str(j) + " = " + str(answer))
```

- Why would the use of indentation improve it?
  - Indentation makes it possible to easily see which lines of code are part of different structures
- Why may indentation be essential in some languages?
  - Some languages use braces { } to show where structures start and end, but some, such as Python use indentation

Commenting

- Comments in code help other programmers to understand your code
  - They also help you understand your code when you go back to it at a later time
- Which parts of programming code tend to be commented?
  - Which parts are typically not commented?

Commenting

Answers

- Comments are usually written for:
  - Parts of a program/algorithm that are difficult to understand
  - At the start of a function or procedure to explain what it does
- Comments usually aren't written for:
  - Every line of code
  - To explain parts of code that are obvious
  - To explain syntax used in the programming language – programmers are expected to look up and learn parts of the language that they don't understand

Plenary

- Work in pairs to explain the following terms and how they can improve the design of programs
  - Input validation
  - Anticipating misuse
  - Authentication
  - Use of sub programs
  - Naming conventions
  - Indentation
  - Commenting

Plenary

Answers

- Input validation – checking input meets certain rules, e.g. the type of data
- Anticipating misuse – predicting how a system could be use in a wrong way, e.g. preventing too many entries of a password to make it harder for hackers to guess
- Authentication – checking that a user or device is allowed access a system
- Use of sub programs – creates reusable code where bugs can easily be fixed
- Naming conventions – good use of variables and sub program names makes programs easier to read
- Indentation – makes programs easier to read
- Commenting – helps programmers understand what a program does and how it does it

Copyright

© 2024 PG Online Limited

The contents of this unit are protected by copyright.

This unit and all the worksheets, PowerPoint presentations, teaching guides and other associated files distributed with it are supplied to you by PG Online Limited under licence and may be used and copied by you only in accordance with the terms of the licence. Except as expressly permitted by the licence, no part of the materials distributed with this unit may be used, reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic or otherwise, without the prior written permission of PG Online Limited.

Licence agreement

This is a legal agreement between you, the end user, and PG Online Limited. This unit and all the worksheets, PowerPoint presentations, teaching guides and other associated files distributed with it is licensed, not sold, to you by PG Online Limited for use under the terms of the licence.

The materials distributed with this unit may be freely copied and used by members of a single institution on a single site only. You are not permitted to share in any way any of the materials or part of the materials with any third party, including users on another site or individuals who are members of a separate institution. You acknowledge that the materials must remain with you, the licencing institution, and no part of the materials may be transferred to another institution. You also agree not to procure, authorise, encourage, facilitate or enable any third party to reproduce these materials in whole or in part without the prior permission of PG Online Limited.

OCR
Oxford Cambridge and RSA
An OCR endorsed teaching and learning tool

GCSE OCR

Computer Science
J277

3

Errors and testing

Unit 8
Logic and languages

PG ONLINE

Objectives

- Understand the purpose of testing including:
  - Iterative testing
  - Final/terminal testing
- Identify syntax and logic errors
- Select and use suitable test data including:
  - Normal
  - Boundary
  - Invalid
  - Erroneous

Starter

```
function max(a,b,c)
    if a >= b AND a >= c
        return a
    elseif b >= a AD b >= c
        return b
    else
        return a
    endif
endfunction

num1 = input("Enter number: ")
num2 = input("Enter number: "
num3 = input("Enter number: ")

maxNum = max(num1, num2, nu3)

print "maximum number is: ")
print(maxNum)
```

- What are the **six** mistakes in the code above

Starter

Answers

AND

```
function max(a,b,c)
    if a >= b AND a > c
        return a
    elseif b >= a AD b >= c
        return b
    else
        return a
endif
endfunction

num1 = input("Enter number: )
num2 = input("Enter number: "
num3 = input("Enter number: ")

maxNum = max(num1, num2, nu3)

print "maximum number is: ")
print(maxNum)
```

c

(

"

)

num3

Syntax errors

- A syntax error is one where the code written doesn't conform to the rules of the language
  - The compiler doesn't know how to translate the program into machine code so will give the programmer a syntax error
  - The program cannot be run until all syntax errors are fixed
  - Which of the errors in the starter were syntax errors?

AND

```
function max(a,b,c)
    if a >= b AND a >= c
        return a
    elseif b >= a AD b >= c
        return b
    else
        return a
endif
endfunction
num1 = input("Enter number: )
```

c

"

)

num3

(

PG ONLINE

Syntax errors

Answers

- All the errors are syntax errors except `return a` should be `return c`
  - This is because they all don't conform to the grammar of the language, so the compiler doesn't know how to translate them

AND

```
function max(a,b,c)
    if a >= b AND a >= c
        return a
    elseif b >= a AD b >= c
        return b
    else
        return a
endif
endfunction

num1 = input("Enter number: )
num2 = input("Enter number: "
num3 = input("Enter number: ")
```

"

)

num3

(

Logical errors

- The remaining error is a logical error
  - The program will run, but it won't work as the programmer intended
  - It will give the wrong maximum number if c is greatest

```
function max(a,b,c)
    if a >= b AND a >= c
        return a
    elseif b >= a AD b >= c
        return b
    else
        return a
endif
endfunction
```

c

```
num1 = input("Enter number: )
num2 = input("Enter number: "
num3 = input("Enter number: ")
```

Worksheet 3

- Now complete **Task 1** on **Worksheet 3**

Testing

- It is important that programs are fully tested to find logical errors
- Before testing, a test plan is created which will test for:
  - Normal data
  - Boundary data
  - Invalid data
  - Erroneous data
- What do you think each of these types of test mean?

Test data

- Suppose the user is asked to enter a number between 1 and 100
  - What test data would you use to test your program?

PG ONLINE

Tests

Answers

- Program: enter a number between 1 and 100
- Some examples of test data:
  - **Normal data:** 5 (checks a single digit), 14 (checks two digits)
  - **Boundary data:** 1 (checks lowest valid data), 100 (checks highest valid data)
  - **Invalid data:** -50 (checks negative numbers), 173 (checks data of the correct type that is invalid), 0 (checks just below lower boundary), 101 (checks just above upper boundary)
  - **Erroneous:** "ade", "#$" (checks data of the wrong type is rejected)

Finding logic errors

- A trace table is useful for tracing through a program in order to find a logic error
  - The value of each variable is recorded as it changes
- Another useful technique is to insert print statements at various points in the program to check the values of variables

Using a trace table

- The value of each variable is recorded as it changes
- What value is output in the code below?

```
num = 3
n = 0
while n < 4
   num = num + n
   n = n + 1
endwhile
output(num)
```

num	n	n < 4	OUTPUT
3	0	TRUE	

Using a trace table

- The value of each variable is recorded as it changes
- What value is output in the code below?

```
num = 3
n = 0
while n < 4
   num = num + n
   n = n + 1
endwhile
output(num)
```

num	n	n < 4	OUTPUT
3	0	TRUE	
3	1	TRUE	

Using a trace table

- The value of each variable is recorded as it changes
- What value is output in the code below?

```
num = 3
n = 0
while n < 4
   num = num + n
   n = n + 1
endwhile
output(num)
```

num	n	n < 4	OUTPUT
3	0	TRUE	
3	1	TRUE	
4	2	TRUE	

Using a trace table

- The value of each variable is recorded as it changes
- What value is output in the code below?

```
num = 3
n = 0
while n < 4
   num = num + n
   n = n + 1
endwhile
output(num)
```

num	n	n < 4	OUTPUT
3	0	TRUE	
3	1	TRUE	
4	2	TRUE	
6	3	TRUE	

Using a trace table

- The value of each variable is recorded as it changes
- What value is output in the code below?

```
num = 3
n = 0
while n < 4
   num = num + n
   n = n + 1
endwhile
output(num)
```

num	n	n < 4	OUTPUT
3	0	TRUE	
3	1	TRUE	
4	2	TRUE	
6	3	TRUE	
9	4	FALSE	**9**

Creating a trace table

- A trace table is useful for
  - Determining the purpose of an algorithm
  - Finding the output of an algorithm
  - Finding errors in an algorithm
- To draw a trace table, make a column for each variable used, in the order in which they appear
- You don’t need to fill in a value for a variable which does not change in a particular row

Determine the function of an algorithm

- Complete the trace table for the algorithm and state its function:
  - Assume the user enters values 8, 3, 6, 5, 10, 6

```
total = 0
for i=1 to 3
    base = int(input())
    height = int(input())
    x = (base * height)/2
    total = total + x
next i
result = total / 3
print(result)
```

total	i	base	height	x	result
0	1	8	3	12	

Determining the function of this algorithm

- Complete the trace table for the algorithm and state its function:
  - Assume the user enters values 8, 3, 6, 5, 10, 6

```
total = 0
for i=1 to 3
    base = int(input())
    height = int(input())
    x = (base * height)/2
    total = total + x
next i
result = total / 3
print(result)
```

total	i	base	height	x	result
0	1	8	3	12	
12	2	6	5	15	

Determining the function of this algorithm

- Complete the trace table for the algorithm and state its function:
  - Assume the user enters values 8, 3, 6, 5, 10, 6

```
total = 0
for i=1 to 3
    base = int(input())
    height = int(input())
    x = (base * height)/2
    total = total + x
next i
result = total / 3
print(result)
```

total	i	base	height	x	result
0	1	8	3	12	
12	2	6	5	15	
27	3	10	6	30	

Determining the function of this algorithm

- The algorithm finds the average of the areas of the triangles
  - Assume the user enters values 8, 3, 6, 5, 10, 6

```
total = 0
for i=1 to 3
    base = int(input())
    height = int(input())
    x = (base * height)/2
    total = total + x
next i
result = total / 3
print(result)
```

total	i	base	height	x	result
0	1	8	3	12	
12	2	6	5	15	
27	3	10	6	30	
57					**19**

Worksheet 3

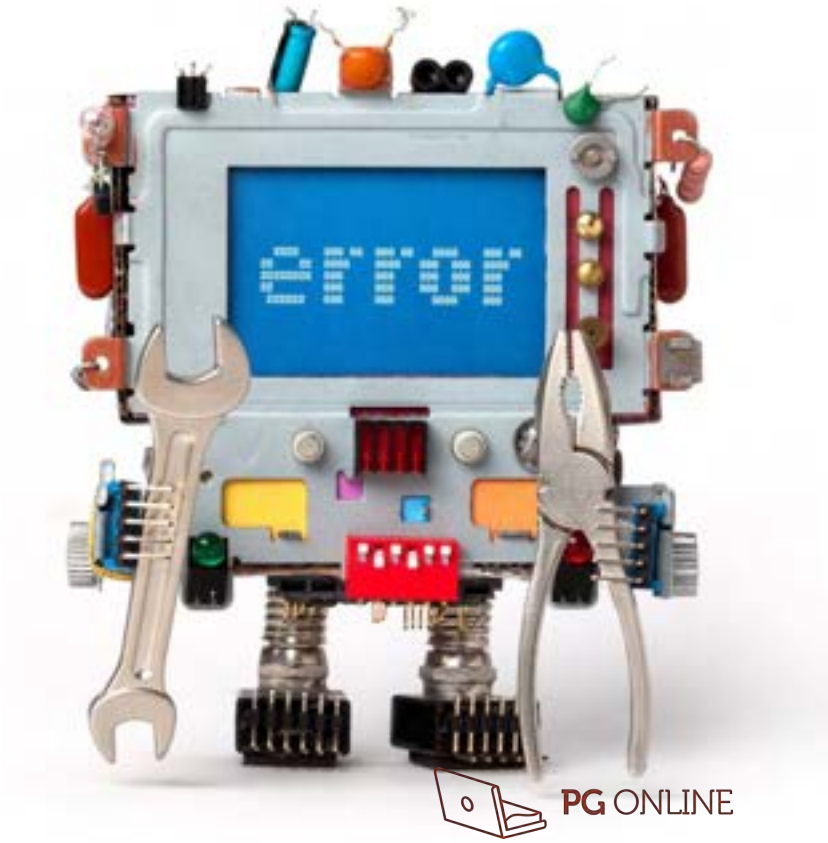
- Now complete **Task 2**, **Task 3** and **Task 4** on **Worksheet 3**

Iterative and final testing

- There are two types of testing:
  - **Iterative testing** tests modules and parts of a program as the program is developed
  - **Final testing** (also known as terminal testing) tests the whole program at the end of production
- In iterative testing, the programmer will usually test the code with knowledge of how it works
  - In final testing, the program may be tested by the programmer or end user – as they are testing the whole program, they won't necessarily know how it works

Plenary

- In a pair, answer the following questions
  - Name **four** types of test data that can be used when testing
  - Explain what a syntax error is
  - Give **three** examples of a syntax error
  - Explain with an example what a logic error is



Plenary

Answers

- Name **four** types of test data:
  - Normal, boundary, invalid, erroneous
- Explain what a syntax error is:
  - An error where the programming code written doesn't conform to the rules/grammar of the language
- Give **three** examples of a syntax error:
  - print("hello) 14 = age remainder = 10 MD 3
- Explain with an example what a logic error is:
  - Where the program compiles/runs but it doesn't do what the programmer intended

Copyright

© 2023 PG Online Limited

The contents of this unit are protected by copyright.

This unit and all the worksheets, PowerPoint presentations, teaching guides and other associated files distributed with it are supplied to you by PG Online Limited under licence and may be used and copied by you only in accordance with the terms of the licence. Except as expressly permitted by the licence, no part of the materials distributed with this unit may be used, reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic or otherwise, without the prior written permission of PG Online Limited.

Licence agreement

This is a legal agreement between you, the end user, and PG Online Limited. This unit and all the worksheets, PowerPoint presentations, teaching guides and other associated files distributed with it is licensed, not sold, to you by PG Online Limited for use under the terms of the licence.

The materials distributed with this unit may be freely copied and used by members of a single institution on a single site only. You are not permitted to share in any way any of the materials or part of the materials with any third party, including users on another site or individuals who are members of a separate institution. You acknowledge that the materials must remain with you, the licencing institution, and no part of the materials may be transferred to another institution. You also agree not to procure, authorise, encourage, facilitate or enable any third party to reproduce these materials in whole or in part without the prior permission of PG Online Limited.

OCR
Oxford Cambridge and RSA
An OCR endorsed teaching and learning tool

GCSE OCR

Computer Science
J277

Translators and facilities of languages

Unit 8
Logic and languages

PG ONLINE

4

Objectives

- Describe the characteristics and purpose of different levels of programming language, including:
  - Low-level languages
  - High-level languages
- Understand the purpose of translators
- Describe the characteristics of a compiler and interpreter

Starter

- Name **five** programming languages
  - Are there any major differences between each of them?

```
if ( typeof types === "object" ) {

    // ( types-Object, selector, data )
    if ( typeof selector !== "string" ) {

        // ( types-Object, data )
        data = data || selector;
        selector = undefined;
    }
    for ( type in types ) {
        on( elem, type, selector, data, types[ type ], one );
    }
    return elem;
}
```

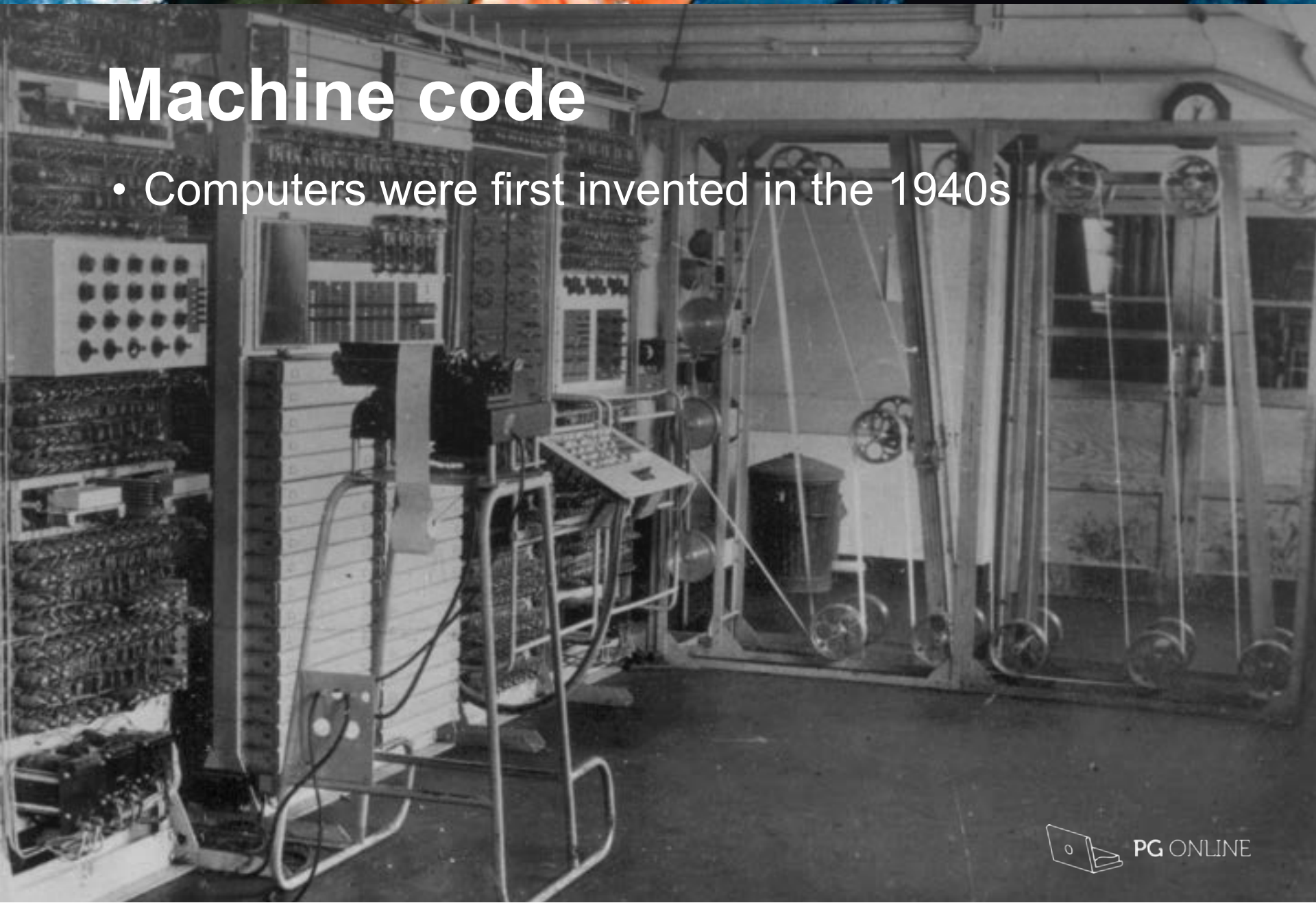
Starter

Answers

- High-level Programming languages
  - Python, Visual Basic, C#, Java, C++, PHP, Delphi, Logo
- Query languages
  - SQL
- Low-level Programming languages
  - Assembly language, machine code

Machine code

- Computers were first invented in the 1940s



PG ONLINE

Machine code

- In the first computers, all programs were written in machine code
- Instructions were written in binary, so a typical instruction looked like this:

101011001001

- Each instruction did one very small task like

LOAD the value 1 into the accumulator

- Writing programs was difficult and time-consuming

Assembly language

- Assembly language allows a programmer to create programs more easily that writing in machine code
  - Each assembly language instruction maps directly to machine code
- For example:

```
LDA   51
ADD   #FF
STO   52
```

Assembly language

→

```
10001011000001000010010100110011000000000000000000000000000000
10111111111000000000000000000000
00000100010010000010000100101001001101000000000
```

Machine code

means “Load the contents of memory location 51 into the accumulator, add hexadecimal value FF and store the result in location 52”

Assembly language

- Assembly language is processor-specific
- It has to be translated into machine code before it can be executed
- As each instruction corresponds directly to a machine code instruction, it is known as a low-level language

High-level languages

- High-level languages generally have statements that look a bit like English or maths

```
area = (base * height) / 2
print(area)
```

- This makes these languages easier to learn and understand
  easy to learn
  and understand

PG ONLINE

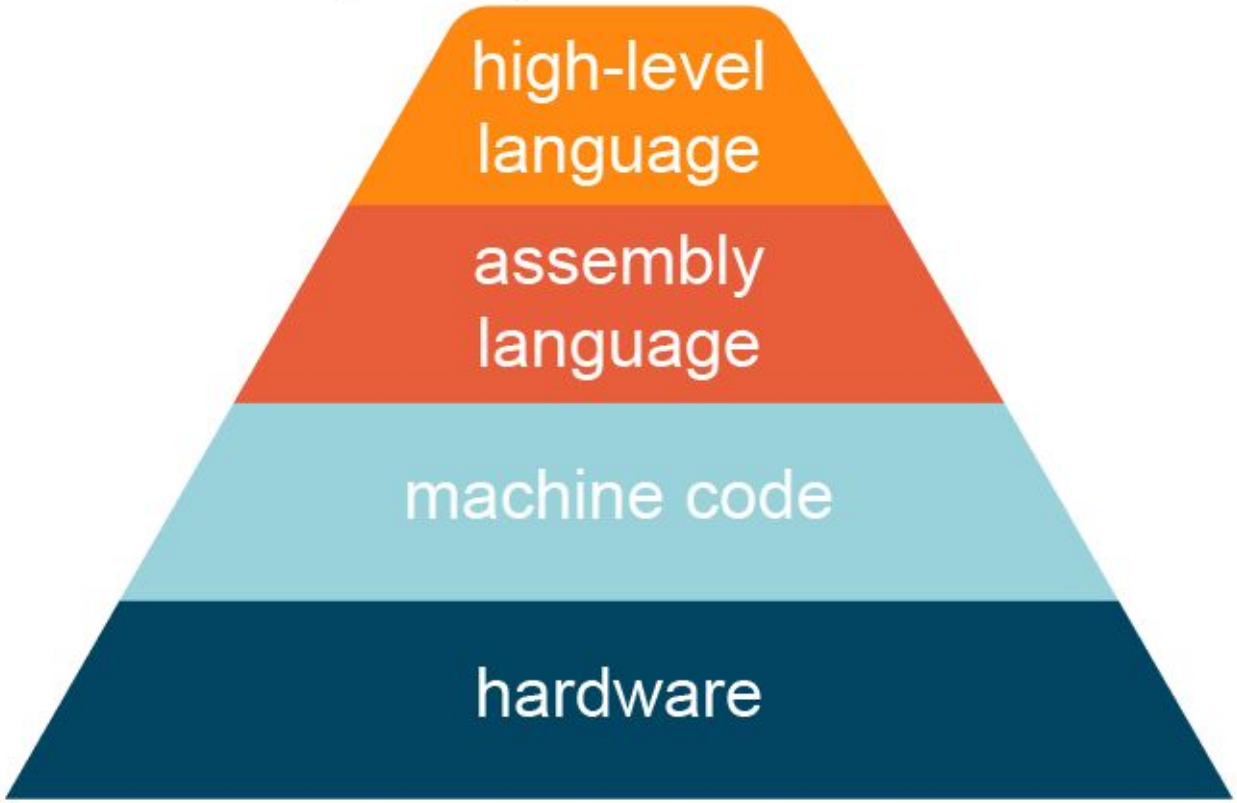
High-level languages

- High-level languages also have data structures such as arrays and records
- Many high-level languages have been specifically designed to make it as easy as possible to write programs to solve certain types of problem
  - A single statement usually translates into several machine code instructions
- The translation is done by a program which may be either a compiler or an interpreter

Machine independence

- A program written in a high-level language such as Python or VB can be run on different types of processor with very few changes, if any, to the program statements
  - A different compiler or interpreter is used for each type of processor to translate the source code (written by the programmer) into machine code for that processor

High- and low-level languages



High-level advantages

- There are a number of advantages of high-level languages, including:
  - A high-level language is easier to learn
  - Programs can be written faster in a high-level language
  - It is easier to understand and debug a high-level language
- Given all these advantages, why do you think that low-level languages are still used by some programmers?

Low-level advantages

Answers

- As a programmer has direct control over how a low-level program works they have a number of advantages, including:
  - A program written in a low-level language can run very quickly
  - The code will usually require less RAM
  - Statements in a low-level language can be used to control and manipulate specific hardware components
- Programs such as device drivers are often written in assembly code

Worksheet 4

- Now complete **Task 1** on **Worksheet 4**

Compiler

- A compiler translates a high-level language into machine code
  - The code written by the programmer is called the source code
  - The code produced by the compiler is called the object code

- The object code can be saved a storage drive and run whenever required

Interpreter

- An interpreter is another type of program that translates a high-level language one line at a time
  - Unlike a compiler, no object code is produced
  - It translates each line of code and executes it immediately
  - If it reaches a line with a syntax error, it stops and displays an error message
- Interpreted code can run on any computer with an interpreter which makes the code more portable

Compiler vs interpreter

Compiler	Interpreter
Translates the whole program to produce object code	Translates and executes one line at a time
A compiled program executes faster as it is already in machine code	Takes more time to execute as each instruction is translated before it is executed
No need for the compiler to be present when object code is run	The interpreter must be installed to run the program
The compiled program will only work on one type of processor	The program is portable to any computer with the interpreter
Customers cannot see the code when they buy it so cannot adapt it	Customers can see the source code so adapt it or see how it works

Compiler or interpreter?

- Some languages, such as Java, are compiled into an intermediate stage called bytecode
  - The bytecode can be interpreted on many different types of processor using an interpreter
- Javascript, used in creating web pages, is interpreted; the source code is included in the web page and then interpreted in the browser (e.g. Firefox, Chrome, Internet Explorer)

Worksheet 4

- Now complete **Task 2** on **Worksheet 4**

Plenary

- Complete the following by filling in the blanks

_____ languages include _____, VB and Java. They need to be _____ or interpreted before they can be run. _____ is a low-level language. It is _____ into _____ before it is run.

Compiled Assembly language Machine code Python High-level assembled

Plenary

Answers

- Complete the following by filling in the blanks

High-level languages include Python, VB and Java. They need to be compiled or interpreted before they can be run. Assembly language is a low-level language. It is assembled into machine code before it is run.

Compiled Assembly language Machine code
Python High-level assembled

Copyright

© 2024 PG Online Limited

The contents of this unit are protected by copyright.

This unit and all the worksheets, PowerPoint presentations, teaching guides and other associated files distributed with it are supplied to you by PG Online Limited under licence and may be used and copied by you only in accordance with the terms of the licence. Except as expressly permitted by the licence, no part of the materials distributed with this unit may be used, reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic or otherwise, without the prior written permission of PG Online Limited.

Licence agreement

This is a legal agreement between you, the end user, and PG Online Limited. This unit and all the worksheets, PowerPoint presentations, teaching guides and other associated files distributed with it is licensed, not sold, to you by PG Online Limited for use under the terms of the licence.

The materials distributed with this unit may be freely copied and used by members of a single institution on a single site only. You are not permitted to share in any way any of the materials or part of the materials with any third party, including users on another site or individuals who are members of a separate institution. You acknowledge that the materials must remain with you, the licencing institution, and no part of the materials may be transferred to another institution. You also agree not to procure, authorise, encourage, facilitate or enable any third party to reproduce these materials in whole or in part without the prior permission of PG Online Limited.

OCR
Oxford Cambridge and RSA
An OCR endorsed teaching and learning tool

GCSE OCR

Computer Science
J277

5

The Integrated Development Environment

Unit 8
Logic and languages

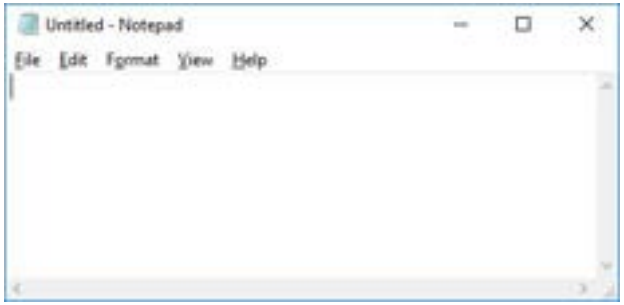
PG ONLINE

Objectives

- Understand the use of an Integrated Development Environment (IDE) to develop programs, including the following:
  - Editors
  - Error diagnostics
  - Run-time environment

Starter

- The simplest way to edit the source code to programs is to use a text editor such as Notepad
  - What problems are there with using such a simple text editor?



Starter

Answers

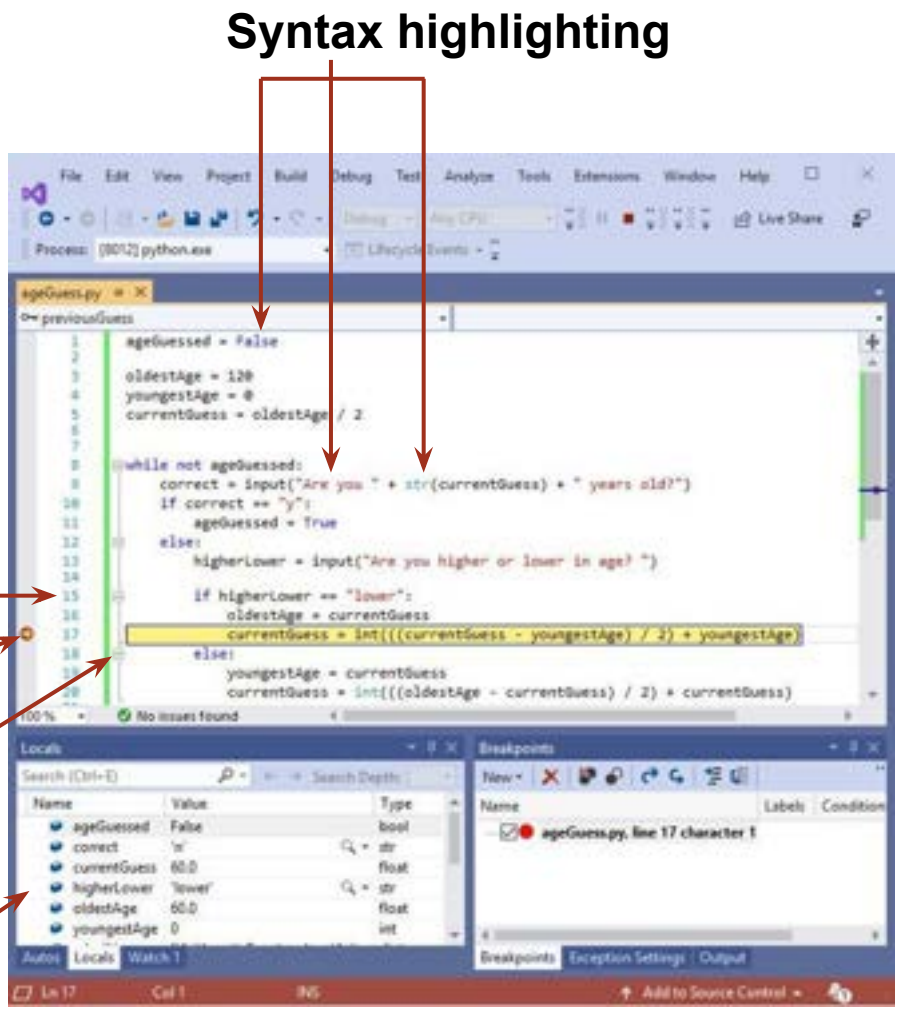
- There are many problems that such a simple text editor has, including:
  - No line numbers
  - No syntax highlighting (colour coding of different parts of code)
  - No ability to run code directly inside the editor
  - No assistance in debugging code

IDEs

- IDEs are Integrated Development Environments
  - They have a number of tools and features that help programmers when they are programming
- Commonly used IDEs include Visual Studio (for C#, VB and other languages), Eclipse (for Java and many more) and IDLE (for Python)
  - What features do IDEs provide?

IDEs

- The following screenshot shows some of the features of an IDE
  - Explain what each of these features mean



Line numbers

- Line numbers allow a programmer to clearly see each new line of code
  - When errors are found, the line number that they occur on will also be stated
  - In some IDEs, parts of the program that the programmer doesn't need to see can be folded

```
1   ageGuessed = False
2
3   oldestAge = 120
4   youngestAge = 0
5   currentGuess = oldestAge / 2
6
7
8   while not ageGuessed:
9       correct = input("Are you " + str(currentGuess)
10      if correct == "y":
11          ageGuessed = True
12      else:
13          higherLower = input("Are you higher or lowe
14
15          if higherLower == "lower":
16              oldestAge = currentGuess
17              currentGuess = int(((currentGuess - you
18          else:
19              youngestAge = currentGuess
```

```
1   ageGuessed = False
2
3   oldestAge = 120
4   youngestAge = 0
5   currentGuess = oldestAge / 2
6
7
8   while not ageGuessed:...
21
22
23
24
```

Notice the line numbers change where the code has been folded

Syntax highlighting

- Syntax highlighting is where the colour of the text changes to show different parts of the program
  - What do each of the colours mean below?

```
ageGuessed = False

oldestAge = 120
youngestAge = 0
currentGuess = oldestAge / 2


while not ageGuessed:
    correct = input("Are you " + str(currentGuess) + " years old?")
    if correct == "y":
        ageGuessed = True
    else:
        higherLower = input("Are you higher or lower in age? ")

        if higherLower == "lower":
            oldestAge = currentGuess
            currentGuess = int(((currentGuess - youngestAge) / 2) + youngestAge)
        else:
            youngestAge = currentGuess
            currentGuess = int(((oldestAge - currentGuess) / 2) + currentGuess)
```

Syntax highlighting

Answers

	Booleans (false, true), Keywords (while, not, if, else)
	Type conversions (str, int)
	Strings – e.g. "Are you higher or lower in age? "
	All other operators, variables and functions names

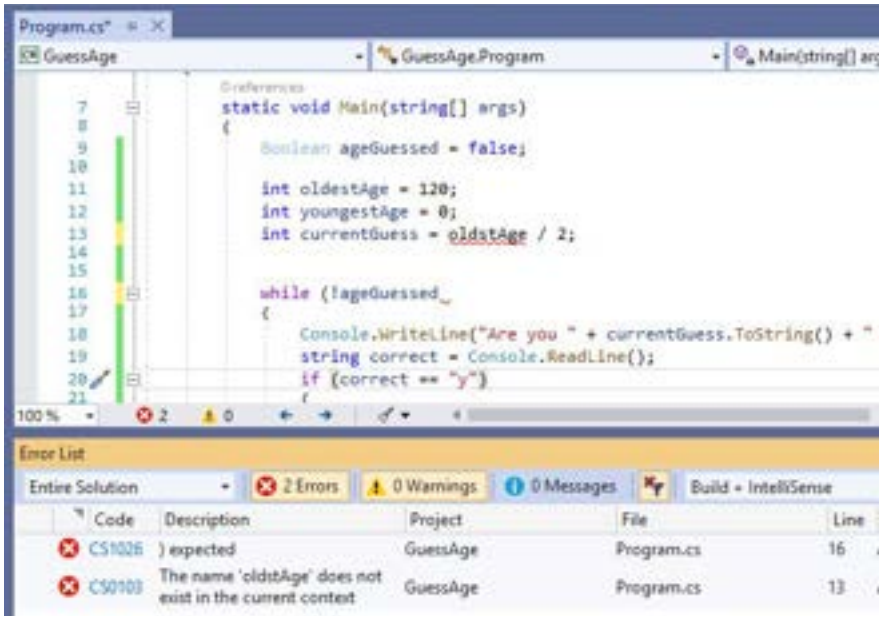
```
ageGuessed = False

oldestAge = 120
youngestAge = 0
currentGuess = oldestAge / 2


while not ageGuessed:
    correct = input("Are you " + str(currentGuess) + " years old?")
    if correct == "y":
        ageGuessed = True
    else:
        higherLower = input("Are you higher or lower in age? ")

        if higherLower == "lower":
            oldestAge = currentGuess
            currentGuess = int(((currentGuess - youngestAge) / 2) + youngestAge)
        else:
            youngestAge = currentGuess
            currentGuess = int(((oldestAge - currentGuess) / 2) + currentGuess)
```

Worksheet 5

- Now complete **Task 1** on **Worksheet 5**

Error diagnostics

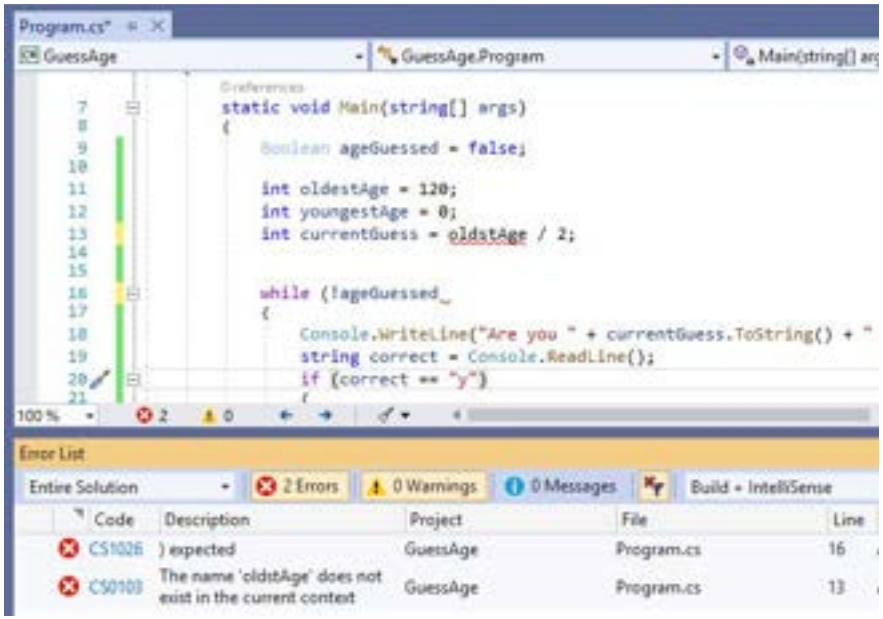
- Error diagnostics help a programmer to find where they have made a mistake
  - Errors are identified along with the line number that they occur on
  - The code may be underlined or highlighted to show the error
- This program is written C#
  - With the help of the error diagnostics fix the two errors shown



Error diagnostics

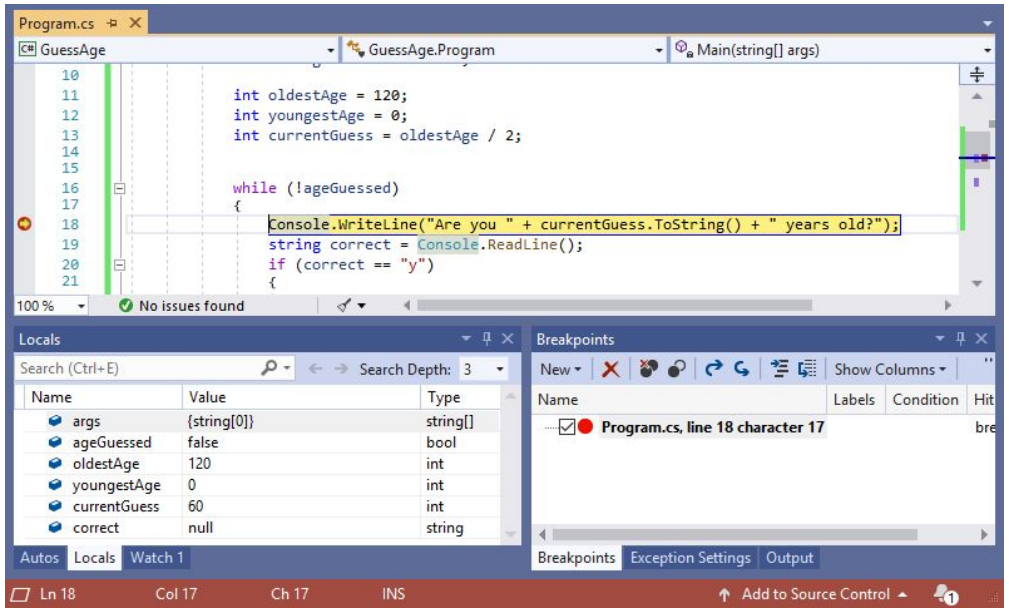
Answers

- The errors show that:
  - A bracket needs to be added to the end of line 16
  - The variable `oldstAge` in line 13 doesn't exist. The programmer should use `oldestAge` which they created in line 11



Debugging code

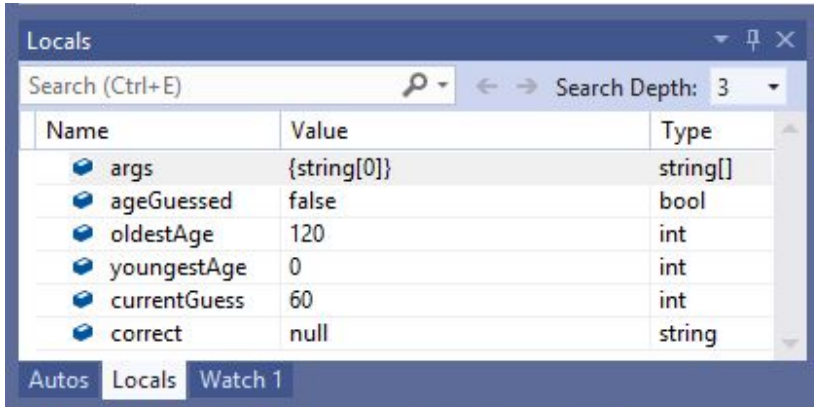
- Breakpoints are set by the programmer so that the IDE pauses the program mid-way through running
  - The programmer can the step through code line by line
  - They can watch variables as they change
- The breakpoint was set to line 18
  - What do the variables contain at this point?



Variables

Answers

- `ageGuessed` = false
- `oldestAge` = 120
- `youngestAge` = 0
- `currentGuess` = 60
- `correct = null` (it hasn't yet been assigned a value)
- `args = {string[0]}`
  These are arguments sent to the program when it is called



Run-time environment

- The run-time environment allows a programmer to test their program while it is running
  - If the program crashes the run-time environment can see what happened and give useful information to the programmer
- The run-time environment also means the libraries that come with the programming language and will be available to the programmer to use

Worksheet 5

- Now complete **Task 2** on **Worksheet 5**

Plenary

- With a partner, take it in turns to explain each of the following terms related to IDEs:
  - Editors
  - Line numbers
  - Syntax highlighting
  - Breakpoints, stepping and watching variables
  - Error diagnostics
  - Run-time environment

Plenary

Answers

- **Editors** – Use to write and edit programming code
- **Line numbers** – Give a number to each line of code, this is useful when the IDE refers to lines of code
- **Syntax highlighting** – Different colours for code that has different meanings, such as keywords, strings or variables
- **Breakpoints, stepping and watching variables** – The ability to pause code when running and then run each line of code separately whilst watching the values stored in variables as they change
- **Error diagnostics** – Comments from the IDE that help in finding the cause of syntax errors
- **Run-time environment** – The facility of the IDE that allows programs to be run

Copyright

© 2024 PG Online Limited

The contents of this unit are protected by copyright.

This unit and all the worksheets, PowerPoint presentations, teaching guides and other associated files distributed with it are supplied to you by PG Online Limited under licence and may be used and copied by you only in accordance with the terms of the licence. Except as expressly permitted by the licence, no part of the materials distributed with this unit may be used, reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic or otherwise, without the prior written permission of PG Online Limited.

Licence agreement

This is a legal agreement between you, the end user, and PG Online Limited. This unit and all the worksheets, PowerPoint presentations, teaching guides and other associated files distributed with it is licensed, not sold, to you by PG Online Limited for use under the terms of the licence.

The materials distributed with this unit may be freely copied and used by members of a single institution on a single site only. You are not permitted to share in any way any of the materials or part of the materials with any third party, including users on another site or individuals who are members of a separate institution. You acknowledge that the materials must remain with you, the licencing institution, and no part of the materials may be transferred to another institution. You also agree not to procure, authorise, encourage, facilitate or enable any third party to reproduce these materials in whole or in part without the prior permission of PG Online Limited.

Isaac Computer Science

https://isaaccomputerscience.org/topics/gcse?examBoard=all&stage=all#ocr