

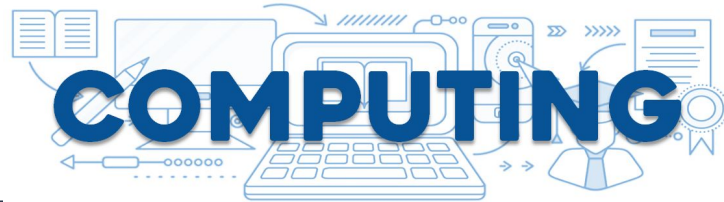
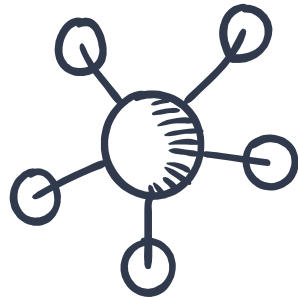


King's Academy  
Bay House




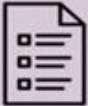














# Computer Science GCSE

## KNOWLEDGE ORGANISER

### COMPUTING



# How to use a knowledge organiser – step by step guide

	Look, Cover, Write, Check	Definitions of Key Words	Flash Cards	Self Quizzing	Mind Maps	Paired Retrieval
Step 1	<p>Look at and study a specific area of your KO.</p> 	<p>Write down the key words and definitions.</p> 	<p>Use your KO to condense and write down key facts or information onto flash cards.</p> 	<p>Use your KO to create a mini quiz. Write down your questions using your KO.</p> 	<p>Create a mind map with all the information you can remember from your KO.</p> 	<p>Ask a friend or family member to have the KO or flash cards in their hands.</p> 
Step 2	<p>Cover or flip the KO over and write down everything you can remember.</p> 	<p>Try not to use your KO to help you.</p> 	<p>Add pictures to help support. Then self-quiz using the flash cards. You could write questions on one side, and answers on the other!</p> 	<p>Answer the questions and remember to use full sentences.</p> 	<p>Check your KO to see if there are any mistakes on your mind map.</p> 	<p>They can test you by asking you questions on different sections of your KO.</p> 
Step 3	<p>Check what you have written down. Correct any mistakes in green pen and add anything you have missed. Repeat.</p> 	<p>Use your green pen to check your work.</p> 	<p>Ask a friend or family member to quiz you on the knowledge.</p> 	<p>Ask a friend or family member to quiz you using the questions.</p> 	<p>Try to make connections, linking the information together.</p> 	<p>Write down your answers,</p> 



## Sorting Algorithms

### Bubble Sort

- The purpose of sorting algorithms is to order an unordered list. Item can be ordered alphabetically or by number.
- Bubble sort steps through a list and compares pairs of adjacent numbers. The numbers are swapped if they are in the wrong order. For an ascending list if the left number is bigger than the right number the items are swapped otherwise the numbers are not swapped.
- The algorithm repeatedly passes through the list until no more swaps are needed.

#### Example

Sort the following sequence in ascending order using bubble sort: 5,3,4,1,2.

Pass 1	5	3	4	1	2	
	3	5	4	1	2	Compare 5 and 3 – swap
	3	4	5	1	2	Compare 5 and 4 – swap
	3	4	1	5	2	Compare 5 and 1 – swap
	3	4	1	2	5	Compare 5 and 2 – swap; end of pass 1
Pass 2	3	4	1	2	5	Compare 3 and 4 – no swap
	3	1	4	2	5	Compare 4 and 1 – swap
	3	1	2	4	5	Compare 4 and 2 – swap
	3	1	2	4	5	Compare 4 and 5 – no swap; end of pass 2
Pass 3	1	3	2	4	5	Compare 3 and 1 – swap
	1	2	3	4	5	Compare 3 and 2 – swap
	1	2	3	4	5	Compare 3 and 4 – no swap
	1	2	3	4	5	Compare 4 and 5 – no swap; end of pass 3
	1	2	3	4	5	

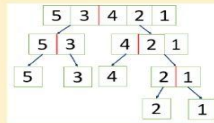
### Bubble sort Pseudocode

```
A=[5, 3, 4, 1, 2]
sorted ← False
WHILE not sorted
  sorted ← True
  FOR I TO LEN(A)-1 :
    IF A[i] > A[i+1] :
      temp ← A[i]
      A[i] ← A[i+1]
      A[i+1] ← temp
    sorted ← False
  ENDFOR
ENDWHILE
OUTPUT A
```

### Merge Sort

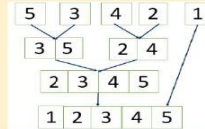
- Merge sort is a type of divide and conquer algorithm.
- There are two steps: divide and combine
- Merge sort works by dividing the unsorted list sublists. It keeps on doing this until there is 1 item in each list.
- Pairs of sublists are combined into an ordered list containing all items in the two sublists. The algorithm keeps going until there is only 1 ordered list remaining.
- Merge sort is a recursive function, that calls itself.

#### Step 1: Divide



Keep dividing until there is only 1 item in each list

#### Step2: Combine



- The first items in the two sublists are compared, and the smallest value is copied to the parent list.
- The copied item is then removed from the sublist.
- When there are no items left in one of the sublists the remaining items in the other sublist are then copied in order to the parent list.

### Merge sort Versus Bubble sort

	Advantages	Disadvantages
<b>Bubble sort</b>	Very simple and robust algorithm	Can be slow particularly for long lists. As the number of items increases the time taken for the algorithm to run increases dramatically.
<b>Merge sort</b>	Much faster than bubble sort especially when the number of elements is large	More complex to understand Step 1: Divide Step 2: Combine

## Algorithms

An **algorithm** is a sequence of ordered instructions that are followed step-by-step to solve a problem. This does *not* need to be on a computer.

**Decomposition** is the breaking down of a complex problem into smaller more manageable problems that are easier to solve.

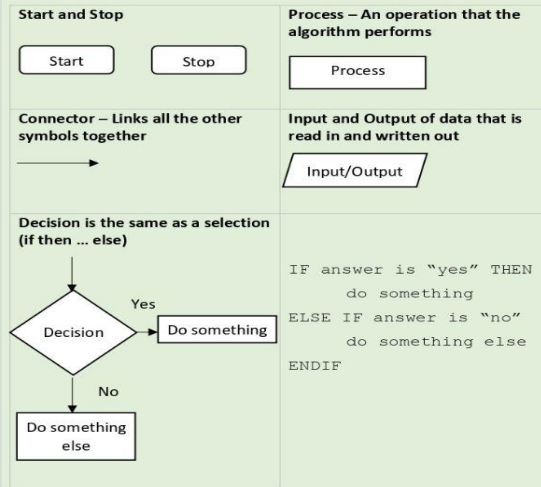
**Abstraction** allows us to remove unnecessary detail from a problem leaving us with only the relevant parts of a problem thereby making it easier to solve.

**Algorithm Efficiency** More than one algorithm can be used to solve the same problem. Normally we use the algorithm that solves the problem in the quickest time with the fewest operations or makes use of the least amount of memory.

**Dry run testing** is carried out using **trace tables**. The purpose of the trace tables is for the programmer to track the value of the variables and outputs at each step of the program and to track how they change throughout the running of the program.

### Flowchart Symbols

We can represent algorithms using flowcharts



## Pseudocode

We can represent algorithms using pseudocode

	Example	Python equivalent
<b>Variable assignment</b>	a ← 10	a = 10
<b>Constant assignment</b>	constant PI ← 3.142	PI = 3.142
<b>Input</b>	a ← USERINPUT	a = input()
<b>Output</b>	OUTPUT "Bye"	print("Bye")
<b>Arithmetic Operators</b>		
Add	+	+
Multiply	*	*
Divide	/	/
Subtract	-	-
Integer division	a ← 7 DIV 2	a = 7 // 2
Modulus (remainder)	a ← 7 MOD 2	a = 7 % 2
<b>Relational Operators</b>		
Less than	<	<
Greater than	>	>
Equal to	=	==
Not equal to	≠ or <>	!=
Less than or equal to	≤	<=
Greater than or equal to	≥	>=
<b>Boolean Operators</b>		
AND	AND	AND
OR	OR	OR
NOT	NOT	NOT
<b>Selection</b>		
if ..	IF i > 2 THEN j ← 10 ENDIF	if i > 2: j=10
if .. else ...	IF i > 2 THEN j ← 10 ELSE j ← 3 ENDIF	if i > 2: j=10 else: j=3
if ... else if ... else	IF i ==2 THEN j ← 10 ELSE IF i==3 THEN	if i ==2: j=10 elif i==3: j=3

```
j ← 3
ELSE
j ← 1
ENDIF
```

```
else:
j=1
```

### Iteration

#### While loops

```
a ← 1
WHILE a < 4
    OUTPUT a
    a ← a + 1
ENDWHILE
```

```
while a<4:
    print(a)
    a=a+1
```

#### For loops

```
FOR a ← 0 TO 3
    OUTPUT a
ENDFOR
```

```
for a in
range(3):
    print(a)
```

#### Repeat loops

```
REPEAT
    OUTPUT a
    a ← a + 1
UNTIL a=4
```

### Subroutines

#### procedure

```
SUB hello()
    OUTPUT "hello"
ENDSUB
```

```
def hello():
    print("hello")
```

#### Function (with parameters and return)

```
SUB add(n)
    a ← 0
    FOR a ← 0 TO n
        a ← a + n
    ENDFOR
    RETURN a
ENDSUB
```

```
def add(n):
    a=0
    for a in
range(n+1):
        a=a+n
    return a
```

### Built-in functions

#### Length of array

```
LEN(a)
```

```
len(a)
```

#### Random integer

```
RANDOM_INT(0, 9)
```

```
import random
random.randint(0,9)
```

## Searching Algorithms

### Linear Search Algorithm

- The purpose of the linear search algorithm is to find a target item within a list.
- Compares each list item one-by-one against the target until the match has been found and returns the position of the item in the list.
- If all items have been checked and the search item is not in the list then the program will run through to the end of the list and return a suitable message indicating that the item is not in the list.
- The algorithm runs in linear time. If  $n$  is the length of the list, then at worst the algorithm will make  $n$  comparisons. At best it will make 1 comparison and on average it will make  $(n+1)/2$  comparisons.
- The performance of the algorithm will be improved if the target item is near the start of the list.

#### Example

Find the position of letter "Z" within the following list. Assume we do not have visibility of the list

Index position	0	1	2	3	4	5	6	7
Value	V	A	S	Z	X	R	T	G

We compare it with the value in index position 0. We find that the value is "V" so we need to move on to the next index position. At index position 1 and 2 we still have not found Z. However, we get to index position 3 and we compare the target with the value and we find that they match, so the algorithm returns the index position and stops.

#### Pseudocode

```
i ← 0
x ← len(listOfItems)
pos ← -1
found ← False
WHILE i < x AND NOT found
  IF listOfItems[i] == itemSearch THEN
    found ← True
    pos ← i + 1
  ENDF
  i=i+1
ENDWHILE
OUTPUT pos
```

## Binary Search Algorithm

- The binary search algorithm works on a sorted list by identifying the middle value in the list and comparing it with the search item.
- If the search item is smaller the mid element becomes the new high value for the search area.
- If the search item is larger the mid element becomes the low value for the search area.
- The keeps repeating until the search item is found.
- When the search item is found the index position of the item is returned.
- At each iteration the search are halved in size consequently this is an efficient algorithm.

Example: Binary search in operation to find 81

	Low		Mid		High						
Iteration 1 L=1, H=11 mid=6	0	5	13	19	22	41	55	68	72	81	98
Iteration 2 L=6, H=11 mid=8	0	5	13	19	22	41	55	68	72	81	98
Iteration 3 L=8, H=11 mid=9	0	5	13	19	22	41	55	68	72	81	98
Iteration 4 L=9, H=11 mid=10	0	5	13	19	22	41	55	68	72	81	98

#### Pseudocode

```
low ← 1
high ← LENGTH(arr)
mid ← (low + high) DIV 2
WHILE val ≠ arr[mid]
  IF arr[mid] < val THEN
    low ← mid
  ELIF arr[mid] > val THEN
    high ← mid
  ENDF
  mid ← (low + high) DIV 2
ENDWHILE
OUTPUT mid
```

## Linear search versus binary search

	Advantages	Disadvantages
<b>Linear Search</b>	<ul style="list-style-type: none"><li>• Very simple algorithm and easy to implement</li><li>• No sorting required</li><li>• Good for short lists</li></ul>	<ul style="list-style-type: none"><li>• slow because it searches through the whole list</li><li>• very inefficient for long lists</li></ul>
<b>Binary Search</b>	<ul style="list-style-type: none"><li>• much quicker than linear search, because it halves the search zone each step</li></ul>	<ul style="list-style-type: none"><li>• The list need to be ordered</li></ul>

## Ethics - Revision notes

## Ethical, Legal and environmental impacts of digital technology on society

### The Ten Commandments of Computer Ethics (From the Computer Ethics Institute)

Thou shalt:

1. not use a computer to harm other people
2. not interfere with other people's computer work
3. not snoop around in other people's computer files
4. not use a computer to steal
5. not use a computer to bear false witness
6. not copy or use proprietary software for which you have not paid (without permission)
7. not use other people's computer resources without authorization or proper compensation
8. not appropriate other people's intellectual output
9. think about the social consequences of the program you are writing or the system you are designing
10. always use a computer in ways that ensure consideration and respect for other humans

### Environmental Impacts

- The disposal of computer waste is a big problem because they contain many toxic chemicals. Often old computing equipment is illegally shipped for disposal to developing countries.
- The growth in cloud computing means a greater need for storing data online. For this data centres are used but they require huge amounts of electricity, thereby contributing to climate change.
- Cobalt is a key element required for Lithium batteries for powering mobile devices. Much of the World's cobalt is mined in the Congo even by very young children in appalling conditions.

### Environmental benefits

- Less reliance on paper saving resources
- More opportunity for online global communication and collaboration thereby saving on travel and associated pollution
- Greater insight of environment and climate through using computer to model and analyse and process environmental data

### Legislation

#### Computer Misuse Act (CMA)

The purpose of the CMA is to prevent:

- unauthorised access to computers by hackers
- intentionally impairing the operation of computer systems through denial of service (DOS) attacks on web servers or distributing viruses
- the theft of data

### Three levels of offence:

- 1) Unauthorised access
- 2) Unauthorised access with intent to commit an offence
- 3) Unauthorised modification of data

### Copyright, Designs and Patents Act (CDPA)

**Copyright** is a law that protects the creators of original pieces of work. No one else has the right to use or copy it without permission from the owner. This ensures that people can be rewarded for their work.

**Plagiarism** To pass off some else's work as one's own work.

**Patent** An inventor has the exclusive right to create, use and sell an invention for fixed period

**Piracy** Illegally copying and distributing copyrighted material.

**Fair use** allows copyrighted work to be used legally in certain situations

- personal or educational use (not commercial use)
- use only a small amount of the work (e.g. a short quote)
- acknowledge original source of the work

**Copyleft** work can be copied, modified used even used for commercial gain as long as the derived works are also distributed under copyleft.

**Creative Common Licences (CCL)** The creator of the work has explicitly given anyone permission to use the work.

**Investigatory Powers Act** This is legislation that allows public authorities to carry out mass surveillance on electronic communications.

**Justification** - By monitoring electronic communications security services can keep us safe from terrorists and other serious criminals

**Concerns** - Can infringe on our privacy and civil liberties

In a liberal democracy there will always a need to balance security and privacy, but where we draw that line will always be a matter of debate.

### Some powers of the security services under the IPA

- can hack into computers, networks, mobile devices, servers
- internet service providers have to store which websites users visit for 12 months and allow access to authorities when requested
- carry out mass surveillance of communications; authorities can collect bulk data including data about people who are not suspected of anything.

- demand that an internet service provider provide access to a customer's communications including keys to encrypted data

### General Data Protection Regulation (GDPR)

The purpose of the GDPR is to ensure that personal information collected by businesses and other organisations are protected.

**Personal data** is defined as anything that allows an individual to be identified (e.g. name, biometric data)

Six principles of the GDPR

**Personal information must:**

- be used fairly and lawfully
- be used only for specific purposes for which it was collected
- be adequate, relevant and not excessive
- be accurate and kept up to date
- be kept for longer than is necessary and deleted when it is no longer needed
- be kept secure against unauthorised access

**Other aspects of the GDPR**

- The data subject needs to be notified if their data are shared with other organisations
- Obtain consent from the data subject to their process data
- Obtain consent from parents or guardians to process children's data.
- Allow data subjects to have their data removed
- Allow data subjects to access the data held about them
- Pay big fines for a breach of the GDPR

### Other Social Impacts

**Artificial Intelligence** is replacing people in jobs. More hi-tech jobs but less need for many lower skilled jobs.

The **digital divide** refers to the unequal access to information technology between different groups of people, and the knowledge and skills needed to use the technology.

Online trolling, cyber bullying and fake news on social media sites is undermining freedom of expression



## Sound - Revision notes

## Sound

**Sample** - Measure of the analogue signal at a given point in time

**Sample rate** - number of samples taken per second and is measured in Hertz.

**Sample resolution** - number of bits used to represent each sample

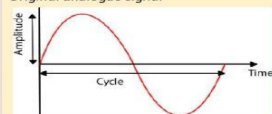
The size of sound files can be calculated using:

$$\text{size of file} = \text{length (seconds)} \times \text{sample rate} \times \text{sampling resolution}$$

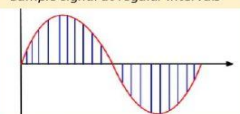
For sound to be stored digitally on a computer it needs to be converted from its continuous analogue form into a discrete binary values. The steps are:

1. Microphone detects the sound wave and converts it into an electrical (analogue) signal
2. The analogue signal is sampled at regular intervals
3. The samples are approximated to the nearest integer (quantised)
4. Each integer is encoded in binary with a fixed number of bits

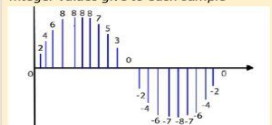
Original analogue signal



Sample signal at regular intervals



Integer values give to each sample



Encode as binary

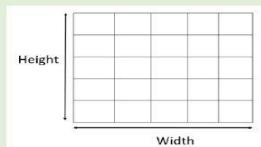
```

0 2 4 6 8 8 8 8 7 5 3 0 ->
00000 00010 00100 01000
01000 01000 01000 00111
00101 00011 ...
    
```

## Images

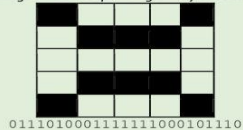
Bitmap images are made up from tiny dots called **pixels**. Each pixel will have a colour associated with it. An image can then be constructed from many of pixels which will have different colours arranged in rows and columns.

Total number of pixels in image = width in pixels x height in pixels



**Colour depth** is the number of bits used to represent each pixel in an image. If we have a black and white image it has two colours. Each pixel can be represented by a single pixel because a bit value of 0 is black and 1 is white.

Image and corresponding binary encoding



To represent more colours we can use more bits. For instance if we have 2-bits per pixel we can represent 4 colours because we know have 4 binary code combinations (00, 01, 10 11) where each code represents a different colour

**Pixilation** occurs when the image is overstretched. In these situations, the image loses quality and has a blocky and blurred appearance. This arises when the image is presented at too large a size and there are not enough pixels to reproduce the details in the image at this larger size.

Calculating the size of a bitmap image

File size in bits = width in pixels x height in pixels x colour depth

File size in bytes = width in pixels x height in pixels x colour depth / 8

## Data Compression

The purpose of data compression is to make the files smaller which means that:

- Less time / less bandwidth to transfer data
- Take up less space on the disk

Given that there are 7 bits per ASCII character, the uncompressed size of an ASCII phrase is:

size = number of characters (including spaces) x 7

**Run Length Encoding (RLE)** is a compression method where sequences of the same values are stored in pairs of the value and the number of those values. For instance, the sequence:  
0 0 0 1 1 0 1 1 1 1 0 1 1 1 1  
would be represented as:  
3 0 2 1 1 0 4 1 1 0 4 1

**Huffman coding** is a form of compression that allows us to use fewer bits for higher frequency data. More common letters are represented using fewer bits than less common letters. For instance, "a" and "e", which occur in many words would be represented with fewer bit than "z" which occurs rarely. This allows for much more effective compression than RLE.

The steps involved in Huffman encoding as are follows:

1. Do frequency table
2. Order table
3. Create the tree
4. Add 1, 0 to the branches
5. Encode letters
6. Encode message

Worked Example: How much smaller is the phrase *henry horse* encoded using Huffman encoding compared with its uncompressed size.

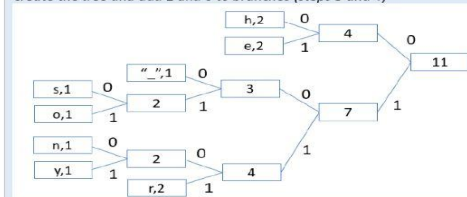
Calculate the uncompressed size

In the phrase *henry horse* there are 11 characters (including the space). Therefore the uncompressed size is  $11 \times 7 = 77$  bits

Generate ordered frequency table (steps 1 and 2)

letter	frequency
e	2
h	2
r	2
<space>	1
o	1
s	1
y	1
n	1

Create the tree and add 1 and 0 to branches (steps 3 and 4)



Encode letters

Letter	encoding
e	01
h	00
r	111
<space>	100
o	1011
s	1000
n	1100
y	1101

Encode message

00 01 1100 111 1101 100 00 1011 111 1000 01 = 33 bits

Therefore by using compression we have reduced the size from 77 bits to 33 bits a saving of 44 bits.

## Cyber Security - Revision notes

## Cybersecurity

Cybersecurity is concerned with the protection of computer systems, computer networks and data. Its purpose is to:

- to protect computers and networks from cyberattacks
- to prevent unauthorised access to computers
- to protect computers against damage caused by malicious software
- to prevent data from being stolen
- to protect against the disruption of services running on the computer

## Cyber Security Threats

**Malware** is software that has been purposely developed to damage, disrupt or take control of computer systems.

**Social engineering** techniques manipulate people into giving away confidential and personal information.

**Weak passwords** are easy to guess. Passwords that use words are easy to crack using an algorithm that systematically goes through all the words in a dictionary until the word matches the password.

**Default passwords** Upon registration for an online account, users may be given a default password that they do not change. Often these passwords are sent out unencrypted via email so pose a major security vulnerability.

**Removable media** such as a USB pen drive can be a vector for transmitting malware.

**Unpatched/outdated software** Software needs regular updates to fix security vulnerabilities in computer systems. Software that remains unpatched is vulnerable to attack.

**Misconfigured access rights** Users should only have access to files and data that they need, but sometimes they have access that they should not.

## Penetration Testing

Penetration testing is legitimate testing of an organisation's computer system to identify whether there are any vulnerabilities that an attacker could exploit. By identifying vulnerabilities, these can be patched before the system gets attacked.

**White box testing** testers are given some information about the network, such as network architecture, source code, and IP addresses. This is designed to simulate an attack by a malicious insider.

**Black box testing** testers are given very little information about the network before the test. This is designed to simulate an outside attack or cyber warfare attack.

## Cyber Security Threats - Malware

**Computer viruses** replicate themselves and can transfer from one computer to another. They are activated by a user often as email attachments and attachment to other files and programs.

**Trojan** gains access to a computer by pretending to be legitimate software. The trojan allows unauthorised backdoor access to a computer without the user being aware.

**Spyware** records the activity on your computer such as your keystrokes, thereby logging your passwords for instance and then send the data back over the network to a hacker. Spyware can also be used to control your webcam and microphone.

**Adware** includes banners and popups that are automatically installed onto a computer. Whilst this does not cause any, adware is undesirable and can slow down the performance of a computer.

**Worms** spread like viruses but do not require human intervention. They attach themselves to network tools to spread automatically around a network very quickly.

## Methods to detect and prevent cyber security threats

**Biometric measures** such as fingerprints, facial recognition and iris scans are increasingly being used to verify a user's identity for mobile devices. These are more secure than passwords that can be guessed and forgotten. Biometric measures require a user to be present when signing into a system.

**Automatic software updates** to firewalls, operating systems, antivirus and other security software are needed so that software can be kept up-to-date against new malware and to fix recently discovered vulnerabilities.

**CAPTCHA** is a test that can distinguish between humans and bots. It uses images that machines cannot interpret but humans can.

**Password systems** Virtually all accounts require passwords to access. Some secure sites such as online banking require 2 passwords. Banks may also contact you by phone to confirm a large transaction. This is called two-factor authentication. Password systems can force users to have strong passwords that regularly need to be changed.

**Using email to confirm a person's identity** Often when you register for an online service you need to provide your email address. You are then requested to activate a link sent to you in an email. This is to confirm that the email account is actually active. Helps to ensure that the users are human and not bots.

**Anti-virus software** scans the computer intermittently to identify whether there is any malware on the computer. The software

compares each file against a database of known virus codes. If viruses are found (ie contains code that is in the database) the file is quarantined. That is the file cannot be run without explicit authorisation from the user. New malware are regularly being created and so anti-virus software needs to be updated to identify the new viruses. That is why anti-virus software is regularly updated.

## Cyber Security Threats – Social Engineering

**Blagging (Pretexting)** Fraudsters make up a scenario to con victims into revealing something they would not ordinarily do. They may have found out some personal information about you from social media sites, to pretend they already know you.

*How to prevent*

- Use biometric measures because these cannot be divulged.
- Ensure you have your privacy settings on any social media to maximum so that fraudsters cannot find information about you such as your date of birth, where you live etc.

**Phishing** Normally an email or text messaging scam where victims are conned into believing that they are being contacted by their bank for instance and can give sensitive personal details such as bank account passwords.

*How to prevent*

- Awareness and vigilance. Be particularly aware of unsolicited texts, emails and phone calls. Do not give personal confidential information away. Official organisations such as banks will never ask for this information.
- Apply email filtering to prevent dubious emails getting through.

**Pharming** Users are redirected to a fraudulent website that they believe to be genuine because it looks like the real site. For instance, you could be directed site that pretends to be an online store that asks you for your credit card information.

*How to prevent*

- Check the URL in the web address. For secure websites such as banking or e-commerce sites the HTTPS protocol should be used.
- Website filter

**Shoulder surfing** Fraudsters look over the shoulder of users to see what passwords or pin numbers that are being typed into the device. This can easily occur at computer terminals and at ATMs that are out in the street.

*How to prevent*

- Be aware of who is around you when typing in your pin into an ATM or into a chip and pin device. Make sure you cover your hands and they are shielded from prying eyes.
- Place computers in locations that makes shoulder surfing difficult

## Computer Networks - Revision notes

## Computer Networks

A network is a set of computers that are connected to one another.

**Standalone** computers are isolated from other devices.

### Advantages of a network

- ✓ Share resources, such as software applications, files and hardware (eg printers).
- ✓ Allows communication (eg email) and can transfer files easily.
- ✓ Easier network management (eg can backup data onto a central files server; updates can be sent to all computers; users on a network can login to any computer)

### Disadvantages of a network

- ✓ Greater security risk as computers can be hacked if they are connected to the internet.
- ✓ Worms can spread from one computer to another
- ✓ A problem with any shared resource, (eg file server goes down) can impact the whole network.

## Types of Computer Networks

**Personal Area Network (PAN)** set up around an individual person. Many people have multiple devices such as tablets, phones and computers that can be interconnected using a PAN. A Bluetooth PAN uses radio waves to communicate wirelessly between devices over a range of a few metres.

**Local Area Network (LAN)** covers a relatively small geographical area typically extends over the range of a single organisation such as a university campus, school site. LANs are usually managed by a single organisation.

**Wide Area Network (WAN)** made up of many local area networks and covers a much wider geographical area. The internet the ultimate WAN. It is a network of networks with billions of interconnected devices. No single person or organisation has control over a WAN.

## Network Topology

A network topology describes how a set of computers are arranged within a network.

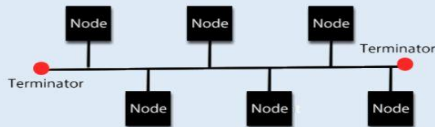
**Bus network topology** All devices including clients, servers, printers and so on are connected to a cable called a bus. All communication is via the shared bus. At either ends of the bus is a terminator.

### Advantages

- ✓ Easy and cheap to install and does not require much cable
- ✓ Easy to add more computers

### Disadvantages

- ✓ If the main cable fails then the whole network fails.
- ✓ Less secure as data are broadcast to all devices on the network.
- ✓ Can be slow as there are collisions between data along the shared bus.
- ✓ Will get slower as more computers are added.



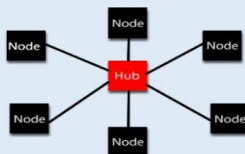
**Star network topology** all devices including clients, servers, printers and so on are connected to a central hub or switch. All communication is via the hub

### Advantages

- ✓ Greater security as data are only sent to the intended recipient.
- ✓ If any of the connections fail only a single node will be affected.
- ✓ Fewer collisions between data packets

### Disadvantages

- ✓ If the central hub fails then every computer on the network is affected.
- ✓ Expensive as extra cable and hardware (hubs) are needed.



## Wired and Wireless

Computers can be connected using wired or wireless methods

**Wired** transmission methods use cables to communicate

**Wireless** transmission use radio waves communicate (eg Wi-Fi).

### Advantages of wireless

- ✓ Can use computer anywhere and not constrained by cables

### Disadvantages of wireless

- ✓ Packets can be intercepted more easily than wired connections
- ✓ Security is a much more difficult challenge, as the network can be accessed from outside the confines of a building.

- ✓ Slower than wired methods
- ✓ Signal can be interfered with by other electronic devices.

### Advantages of wired

- ✓ Allows more control, security and reliability. Can restrict who has access to the network.
- ✓ Wired methods have greater speeds than wireless methods.

### Disadvantages of wired

- ✓ Cables can be difficult to maintain in big organisations

**Wired networks** use a variety of cables, including copper and fibre optic.

**Copper** cables use electrical signals to transmit data. Three main types:

- ✓ **Coaxial cable** – the signal loses strength over long distances
- ✓ **Unshielded twisted pair** – A pair of copper cables are twisted together and allows data to be transmitted over longer distances
- ✓ **Shielded twisted pair** – Shielding around the twisted cables means the signal is less susceptible to interference.

**Fibre optic** cables are glass or plastic and use pulses of light to transmit data

### Advantages of copper cables

- ✓ Cheaper than fibre optic
- ✓ Reliable because a telephone is powered from the copper cable and does not rely on a separate electrical power supply

### Advantages of fibre optic

- ✓ Slow
- ✓ Low capacity
- ✓ Can only be used over short distances
- ✓ Interference can occur

### Advantages of fibre optic

- ✓ Higher bandwidth than copper so can transmit more data
- ✓ Less attenuation (degrading) of the signal so fibre optic is more suitable over long distances
- ✓ Less “cross talk” interference between fibres compared with copper so the quality of the signal is better

### Disadvantages of fibre optic

- ✓ Expensive
- ✓ Difficult to install

## Network Security and Protocols

Why do we need network security?

- ✓ To prevent unauthorised access to our electronic devices
- ✓ To protect our data eg to prevent sensitive data being stolen
- ✓ Prevent cyberattacks

### Methods of Network Security

**Authentication** allows us to confirm the identity an individual.

There are lots of ways of confirming the identity of an individual that come under one of three factors:

- ✓ Knowledge factor: Something the user knows, eg a password
- ✓ Possession factor: Something the user owns eg a mobile phone
- ✓ Biometric factor: eg Fingerprint, iris scan

**Encryption** The message is garbled so if it gets intercepted during transmission it will be almost impossible for anyone without the key to read the original message.

**Firewall** prevents packets containing malware getting on to the computer

**MAC address filtering** A MAC (Media Access Control) address is a unique identifier for any device that is connected to a network. Each network interface card has a unique MAC address that is a 12 digit hexadecimal code (e.g. 12-F3-EE-56-44-A1).

- ✓ *White list filtering* only allows devices on a list to connect to the network.
- ✓ *Black list filtering* devices in a black list blocked from accessing the network.

### Network Protocols

A **network protocol** is a set of rules that allow computers to communicate and exchange information over a network. There are many types of protocols depending on the application.

**HTTP (Hypertext transfer protocol)** is the protocol used for the World Wide Web. An exchange begins with a request for a web page from a client web browser to a web server. The server then sends the web page to the client.

**HTTPS (Secure Hypertext transfer protocol)** is a secure way of transferring data between a web browser and a server because the data are encrypted during transfer. Used for e-commerce and online banking.

**FTP (File Transfer Protocol)** is usually used to download or upload large files from a server to a client.

**Ethernet** is not a single protocol but a collection of related protocols. LANs most commonly use ethernet. The following is a simplified procedure:

- 1) Check whether there is any traffic on the ethernet
- 2) If so wait for traffic to clear
- 3) Send the packet
- 4) If collision detected, go to step 1 to resend.

**Wi-Fi** is a collection of protocol that use radio waves to transmit data between devices. Wi-Fi is a trademark and WLAN (Wireless LAN) is the generic term. Data are transmitted when the medium is clear, and an acknowledgement is received if the transmission was successful. If no acknowledgement is received, then the data are resent as it is assumed that a collision occurred, and the packets did not reach their destination.

### Email protocols

**SMTP (simple mail transfer protocol)** Sends the mail from the user onto the mail server.

**IMAP (Internet Message Access Protocol)** Retrieves the mail from the mail server to the client (user) and allows access from anywhere on any device because the email remains on the server.

**TCP (Transport Control Protocol)** When files are sent over the internet they are broken up into small chunks called packets. When they arrive at the destination computer they are reassembled back into the original format. TCP handles and controls all this. TCP waits for acknowledgements to verify whether the packets have reached their destination. TCP will also retransmit packets of they have not arrived at the destination or become corrupted.

**IP (Internet Protocol)** The internet protocol is a set of rules that govern the transmission of data across the internet.

**UDP (User Datagram Protocol)** is used as an alternative to TCP. It is used in video conferencing and online gaming when speed is necessary as huge volumes of data are transferred in real time. It improves speed by not checking for lost packets so they do not get re-sent.

### TCP/IP

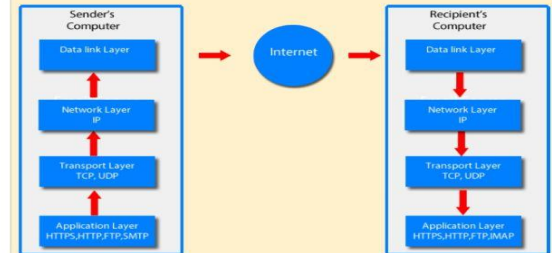
The TCP and IP protocol work closely together and are referred to as TCP/IP. The TCP/IP model consists of four layers that pass data between each layer.

**Application layer** contains protocols related to the application such as HTTP, HTTPS for web browsers, FTP for file transfer and SMTP and IMAP for email. The application layer interacts with the user via appropriate application software (eg web browser / ftp client).

The **transport layer** establishes the end to end connection. When files are sent over the internet, they are broken up into small chunks called packets. When they arrive at the destination computer they are reassembled back into the original format. It is the role of the transport layer to split the data into packets and pass the data onto the network layer. On the recipient's computer the transport layer reassembles the packets into the original form. The packets are numbered by this layer to allow them to be reassembled. The transport layer chooses the port number for sender and receiver. TCP and UDP are the main protocols used in this layer.

The **network layer** adds the source and destination IP address and route the packets over the network. At the destination the network layer strips out the IP addresses. The IP operates on this layer.

The **data link layer** has a network card and deals with the physical connection and adds the physical addresses (MAC address) of the hardware to the packets that it receives from the network layer. For each step the sender and receiver MAC address is removed then a new sender and receiver MAC address is added. The receiver MAC address becomes the sender MAC address.



## Computer Systems - Revision notes



## Computer Systems

A computer system has both hardware and software.

**Hardware** are the physical components that make up a device or computer system. These include both the internal components (eg motherboard, CPU, RAM) and peripheral devices such as printers.

**Software** is the computer code, programs and algorithms that give instructions to the hardware to make it perform the desired task. Without the software the hardware will not get any instructions and it will not do anything.

### Software Classification

Software is split into two types: application software and system software

**Application software** is a program designed to perform a specific task that the user interacts directly with (eg spreadsheets, web browser and word processor, disk defragmentation).

**System software** is concerned with the running of the computer. Its purpose is the control the computer hardware and manage the application software. (eg operating system, antivirus, backup tools, firewall)

The **operating system (OS)** is the most important piece of system software. The OS handles management of the processor, memory, input/output devices, applications and security.

- **Application management** - Application software does not need to concern itself with interaction and complexities of managing the hardware because this is dealt with by the operating system. Application software runs on top of operating system which is an intermediary and takes care of interaction with the hardware.
- **Processor resources** – Allows multiple applications to be run simultaneously by manages the processing time between applications and cores and switching processing between applications very quickly. Multiple applications will access the processor resources via a schedule that alternates process between applications. High priority applications will have more CPU time, but it means that lower priority applications will take longer to run.
- **Memory management** – Distributes memory resources between programs and manages transfer of data and instruction code in and out of memory. Ensures that each application does not use excessive memory.

- **Security** – Tools such as anti-virus software and firewalls help protect the computer from attack. In addition requirement for passwords and control of access rights
- **Input / Output devices** – OS controls interaction with input (eg keyboard) outputs (eg. Monitor) and storage (eg hard disk) using hardware drivers. Allows users to save files to the hard disk and print documents for instance.

### Cloud Computing

- Can store data and files on a server elsewhere that can be accessed via the internet.
- Can use applications over the internet
- Can sync files so that all your devices see the same files
- Can share documents with others
- Can access your files anywhere if you have a good internet connection

#### Advantages of cloud computing

- Only pay for storage that you use
- Data and files available from anywhere in the world where there is an internet connection
- Data automatically backed up

#### Disadvantages of cloud computing

- Need a reliable network connection
- Files are hosted elsewhere so a security concern
- The most recent versions of software is often not available
- Transfer of data over the internet will slow down performance.

#### Advantages of local storage

- Files can be accessed even when there is no internet connection
- More secure as files do not need to be transferred over the network and the user has more control

#### Disadvantages of local storage

- Users need to organise their backup solutions
- Not so easy to share documents
- Can only access the files locally

### Memory

**Volatile memory (main memory)** When the computer is turned off the contents of volatile memory is lost. When there is no power, volatile memory is erased.

**Non-volatile memory (secondary storage)** Even when there is no power, the data remain unchanged and can be accessed once again once power has been resumed. This allows you to store files for the long term.

**ROM (Read Only Memory)** Data can only be read from the device, and cannot the memory cannot be edited or deleted. ROM is only used for situations where you can be sure that updates will not be needed. The computer's BIOS (basic input output system) which controls the boot up sequence is stored on a ROM chip.

**RAM (Random Access Memory)** - When applications are executed they are loaded into RAM first. RAM is volatile.

### Embedded Systems

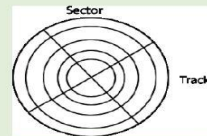
An embedded system is a computer system that is designed for a specific function, in contrast to a general-purpose computer that can carry out many tasks. Embedded systems typically have a minimal or no user interface. Thus, they can be optimised for size and power consumption, for instance. Examples of embedded systems include digital watches, MP3 players, washing machines, cars and mobile phones.

### Secondary Storage

Secondary storage is necessary for saving files long and software including the operating system. Even when the computer is turned off, the data remain unchanged, and can be accessed again once the power supply has been turned on.

#### Magnetic Hard Disk

- Tracks on the disk platters contain tiny magnets, each holding 1 bit of data.
- The polarity (negative or positive) of the magnets determines whether the bits are 0 or 1.
- The write head modifies the polarity of the magnet as appropriate.
- The read head identifies whether each magnet is negative or positive.
- The tracks are laid out as a series of concentric rings.



#### Advantages

- Cheap form of storage

#### Disadvantages

- Less reliable because it contains moving parts that can break
- Electromagnetic surge can corrupt the data held
- Slow speed of read/write access

### Optical Disks

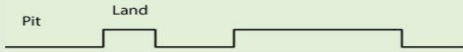
- Tracks on the disk contain pits and lands.
- The track is a spiral.
- A laser is emitted and the laser light is reflected when it hits the lands, but is scattered when it hits the pits.
- Depending on whether the light is scattered light is encoded as a binary value of 0 and reflected light is encoded as a 1.
- The sensor is able to detect light reflected, but not scattered.
- Example: Blue-Ray (25 Gb) DVD (4.7 Gb), CD (700 Mb).

### Advantages

- Can transfer easily between computers

### Disadvantages

- Can scratch easily
- Not much storage compared with other methods.
- No unlimited writes to the hard disk



### Solid state Drive

- Use millions of switches called floating gate transistors on microchips to store data.
- Electrons are stored in gates and this is encoded as 0 when there is an electron present and encoded a 1 when there is no electron present.
- The electrons remain trapped even when there is no flow of electricity.
- Contain no moving parts and are therefore more robust than optical and magnetic storage.

### Advantages

- Much faster than other means of storage
- More reliable than other means if you are only reading
- Quiet

### Disadvantages

- More expensive per volume of storage
- Reliability might be an issue if you do a lot of writing

## Boolean Logic

**NOT gate** - The output is the opposite of the input

$$Q = \bar{A}$$
$$Q = NOT A$$



### NOT truth table

Input	Output
0	1
1	0

**AND gate** - has two inputs and will have a true output if the two inputs are true otherwise the output will be false

$$Q = A \cdot B$$
$$Q = A AND B$$



### AND truth table

Input - A	Input - B	Output
0	0	0
1	0	0
0	1	0
1	1	1

**OR gate** - has two inputs and will have a true output if either or both the inputs are true

$$Q = A + B$$
$$Q = A OR B$$



### OR truth table

Input - A	Input - B	Output
0	0	0
1	0	1
0	1	1
1	1	1

**XOR gate** - has two inputs and will have a true output if either the inputs are true but not both

$$Q = A \oplus B$$
$$Q = A XOR B$$



### OR truth table

Input A	Input B	Output
0	0	0
1	0	1
0	1	1
1	1	0

### Converting a truth table to a logic circuit

There is a general approach to converting a truth table into a logic circuit.

We consider only the lines with an output of 1.  
We take in the input of each and then AND.

We then OR between each statement such that  
(NOT A AND B) OR (A AND NOT B). We can then draw the logic circuit.

*Worked example:* What is the logic circuit for the following truth table

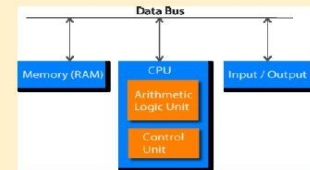
Input - A	Input - B	Output
0	0	0
1	0	1
0	1	0
1	1	1

(A AND NOT B) OR (A AND A)

## System Architecture

**CPU (Computer Processing Unit) or processor** Fetches, decodes and executes instructions and performs logical and arithmetic operations.

**Von Neumann architecture** is the stored program concept, where program instructions and the data to be processed can be stored in the same memory.



### Components of a CPU

**Bus Wires** through which data and instructions are transferred between computer components

**Clock** keeps all the CPU components synchronised

**Arithmetic Logic Unit (ALU)** Every operation takes place here. This is where the arithmetic (eg adding two binary numbers) and logic operations (eg checking to see if one number is bigger than another) take place.

**Control Unit** Decode the machine code instruction so that the ALU knows what to do with the instruction. Controls and monitors data transfer between different input and output hardware components

#### Factors affecting CPU performance

**Clock speed** is the number of cycles that a processor carries out per second. Each cycle of the CPU allows a single action (instruction) to be carried out. The greater the clock speed, the greater the number of operations and the faster the computer will run.

**Number of processor cores** A core is CPU in its own right. Nowadays most CPUs have multiple cores. Having multiple cores allows instructions to be carried out concurrently (at the same time), whereas a single core will only allow carry out instructions in serial (one at a time).

**Latency** Delay in transfer of data between components

**Cache size** Cache is a volatile memory store on the processor. Cache is much faster but smaller than RAM. Frequently used data and instructions within an application can be stored in cache instead of fetching from RAM which is quite slow. The bigger the cache the greater the volume of data and instructions that can be stored thereby reducing latency and improving performance of the CPU.

**Cache type** There are three levels of cache. Cache Level is a trade off between size and speed

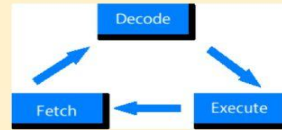
- **Level 1 Cache** closest to the CPU and is the fastest cache (lowest latency), but does not have much capacity
- **Level 2 Cache** – is slower and further away from the CPU than L1 cache so latency is greater, but has more storage capacity.
- **Level 3 Cache** is the slower than L1 and L2 cache; much faster than RAM; has greater capacity than L1 and L2 cache.



#### Fetch execute cycle

1. Instructions are loaded into memory
2. Processor fetches the instruction from the main memory

3. Instruction is decoded so the CPU knows what to do with the instruction
4. Processor then executes the instruction
5. Result of the instruction can be stored in memory
6. Next instruction is then fetched from main memory and the cycle repeats itself.



#### Classification of programming languages

**High level programming languages** are closer to human language and is therefore easier to understand. A translator is used to convert the instructions into code that the computer understand. High level languages allow programs to be written that is independent of the type of computer. High level programming languages allow code to be written that is independent of the type of computer system. It is up to the compiler to translate the code into the right machine code for a particular code. There is a huge variety of high level programming languages, and the choice depends on the application.

**Low level programming languages** refer to machine code and assembly language. The Low level refers to low level of abstraction. The low level language is close to the language understood by the computer where operations map to the instruction in the processor instruction set. However it is difficult for humans to understand. Low level languages are appropriate for developing new operating systems, embedded systems and hardware device drivers

**Machine code** is expressed in binary values 0 and 1. This is the language that computers understand. All codes whether assembler or high level programming languages need to be translated into machine code. Machine code is specific to a processor. Machine code instructions are made up of two parts the operator and the operand. The processor decodes the operator to identify the task that is to be carried out (eg. Add, load). The operand is the value or memory address that that instruction is to be operated on

Machine code instruction	
Operator	Operand
0011	10010100

**Assembly language** provides basic computer instructions for programs to run. There is a one to one relationship between machine code and assembly code instructions. One assembly language instruction maps to one machine code instruction, thus the

structure of assembly language and machine code is the same, but where machine codes uses 0 and 1 which are very difficult for programmer to understand, assembly language uses mnemonics which is easier for the programmer.

#### Assembly language sample instruction set

```
LOAD #23 # Load from RAM to processor
MOV a 23 # Transfer in number 23 into the variable a
ADD 2 3 # Add 2 values
STORE # store data in RAM
```

Each type of processor has its own instruction set and therefore its own assembly language and machine code. So Assembly code written for one type of processor will not run on another.

#### Low level language versus high level languages

	Advantages	Disadvantages
<b>Low level</b>	Produce code that is faster and better optimised than high level languages.  Appropriate for developing new operating systems, embedded systems and hardware device drivers	Difficult to understand and modify  Assembly code is written for a specific processor architecture, and so is not portable to other computer architectures
<b>High level</b>	High level programming languages allow code to be written that is more portable. Thus code can be run on different of the types of computer system with different processor architecture.  Easier to understand  Easier to modify	Needs a translator  run slower because of the layers of abstraction and there is inefficiency in translator.

**Program translators** allow programs to be translated into machine code so the than programs can be run on a computer.

**Interpreter** converts high level languages into machine code one instruction at a time on-the-fly while the program is running. Each instruction is converted to machine code once the previous instruction has been executed. Interpreters are good for debugging code because the program stops as soon as the error has been found. However running code this way is much slower running compiled code. The machine code is not saved.

**Compiler** A program that converts high level languages into machine code before the program is run. A compiler saves the machine code,