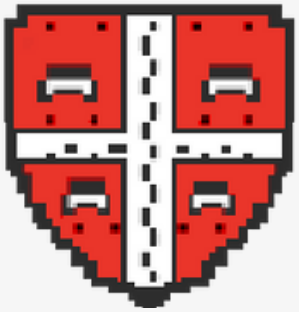


r/IGCSE Resources

Topical Notes for Cambridge IGCSE

Computer Science (0478)

Authored by
ashy and pizza

1st edition, for examination until 2026

Contents

Data Representation

Binary Number System

- It is a number system based on the number 2
- There are only two values used in this system to represent all values; 0 and 1
- The typical headings for a binary number with eight digits are:

2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
128	64	32	16	8	4	2	1

Uses of binary data:

- To process data in logic gates
- To store data in registers
- To process data on the computer

Hexadecimal Number System

- It is a number system based on the number 16
- There are only sixteen values used in this system to represent all values.
  - Denary digits 0 to 9
  - Letters A to F

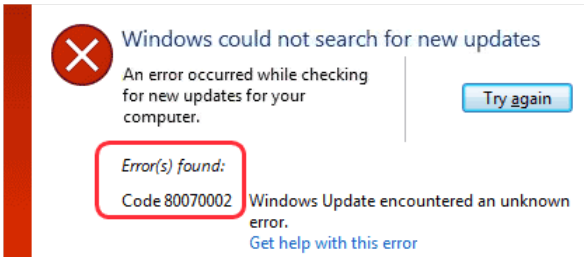
A	B	C	D	E	F
10	11	12	13	14	15

- The typical headings for a hexadecimal number with five digits are:

16^4	16^3	16^2	16^1	16^0
65536	4096	256	16	1

Uses of hexadecimal data:

- **Error Codes**
  - ★ An error message generated by the computer
  - ★ The hexadecimal values refer to the memory location of the error



- **HTML (HyperText Markup Language) colour codes**
  - ★ It is used when writing or developing web pages
  - ★ It is often used to represent text colours on the computer screen
  - ★ All colours can be made up of combinations of the three primary colours (red, green and blue)
  - ★ The different intensity of each colour is determined by its hexadecimal value (meaning that different hexadecimal values represent different colours)

- **MAC (Media Access Control) address**
  - ★ It is a number which uniquely identifies a device on a network
  - ★ Refers to the NIC (Network Interface Card) which is part of the device
  - ★ It is rarely changed
  - ★ It can be of 48/64 bits
  - ★ It is shown as 6 groups of two hexa digits

NN – NN – NN – DD – DD – DD

  - ★ **There are two types of MAC addresses:**
    - ★ **Universally Administered MAC address:**
      - Most common type
      - Set by the manufacturer at the factory
      - Rare for a user to change this MAC address

    - ★ **Locally Administered MAC address:**
      - Easy to change

- **Internet Protocol (IP) address**
  - ★ It gives a unique address to each device connected to a network identifying its location
  - ★ Assigned by the Internet Service Protocol
  - ★ There are **two** versions of IP addresses:
    - **IPv4:**
      - ★ 32 bits
      - ★ Denary or hexadecimal
      - ★ Uses a decimal point

139.103.20.92

    - **IPv6:**
      - ★ 128 bits
      - ★ Broken down into 16 chunks
      - ★ Hexadecimal

2202:f000:e:1f24:190e:323a:9377:d87b

  - ★ **The main advantages of IPv6 compared to IPv4 are:**
    - Removes the risk of IP address collisions
    - Has built-in authentication checks
    - Allows for more efficient packet routes

  - ★ **There are two types of IP addresses:**
    - **Static:**
      - ★ Remains the same each time a device logs onto the internet
      - ★ Remote servers which are hosting a website
      - ★ An online database
      - ★ A File Transfer Protocol (FTP) server

    - **Dynamic:**
      - ★ Change each time a device logs onto the internet
      - ★ Assigned by the ISP by using Dynamic Host Configuration Protocol (DHCP)

Dynamic	Static
Greater Privacy	Allows devices to be fully traceable
Less reliable addressing type	Allow for faster upload and download speeds
VoIP connection fails due to IP address change	More expensive to maintain as the device has to be running constantly for information

Why do programmers use hexadecimal to represent binary data?

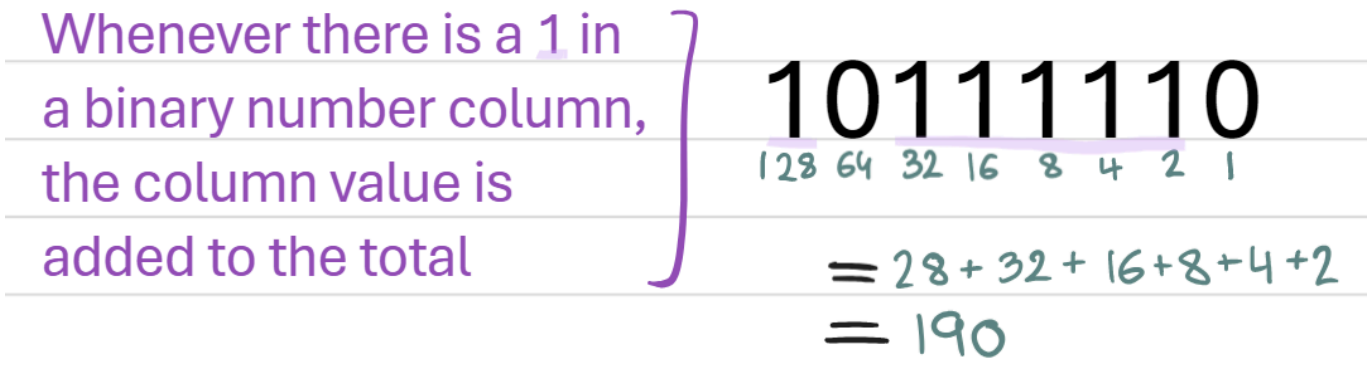
- Easier to debug
- Easier to understand
- It takes up less screen space
- Less likely to make a mistake

Denary Number System

- It is a number system based on the number 10
- 0 - 10 are used as values
- The typical headings for a binary number with five digits are:

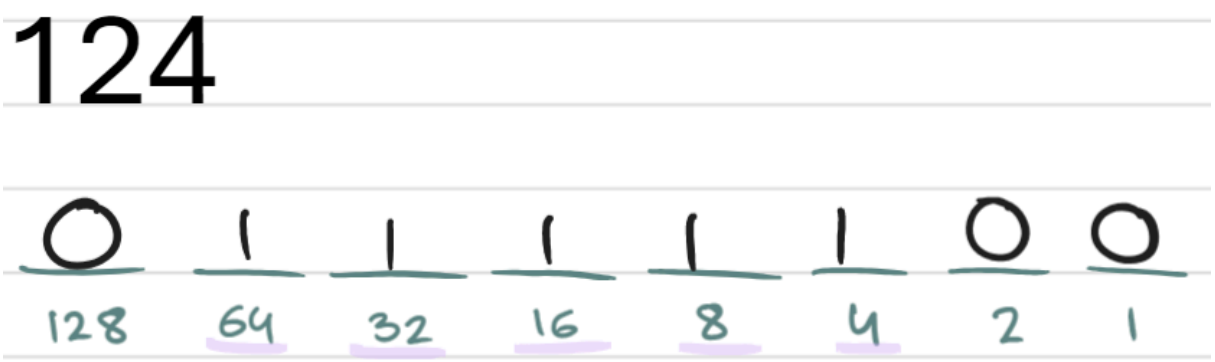
10^4	10^3	10^2	10^1	10^0
10000	1000	100	10	1

Converting Binary to Denary:



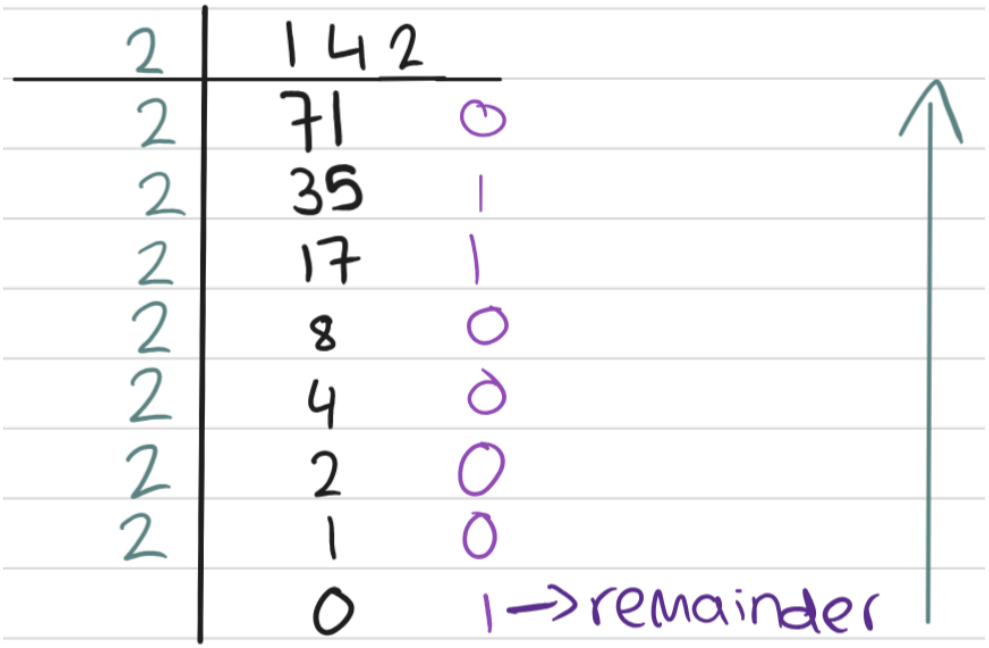
Converting Denary to Binary:

Method 1:

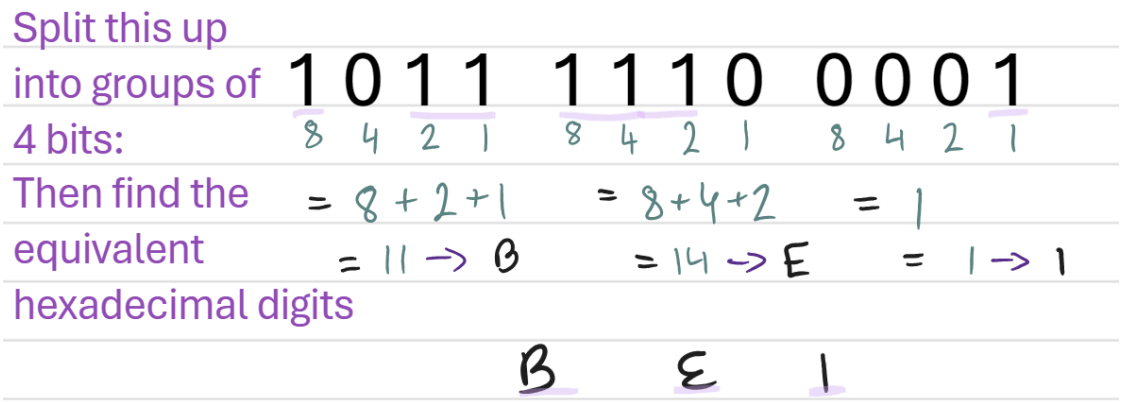


Method 2:

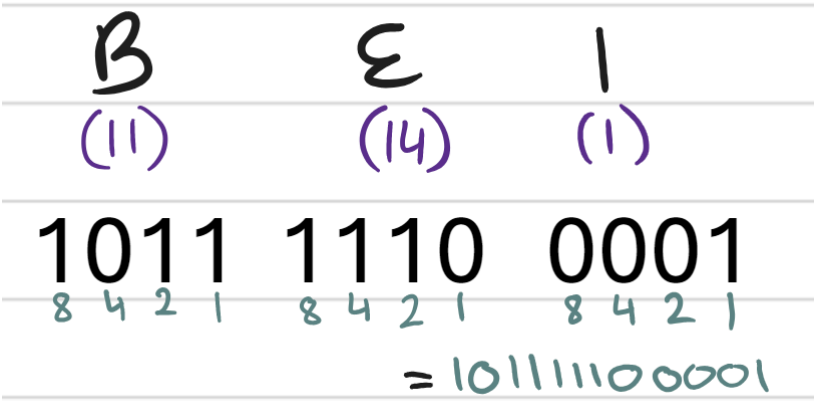
- Divide the number by 2, write the result of the division, including the remainder and keep on dividing until the result is 0.
- Lastly, write down all the numbers in reverse.



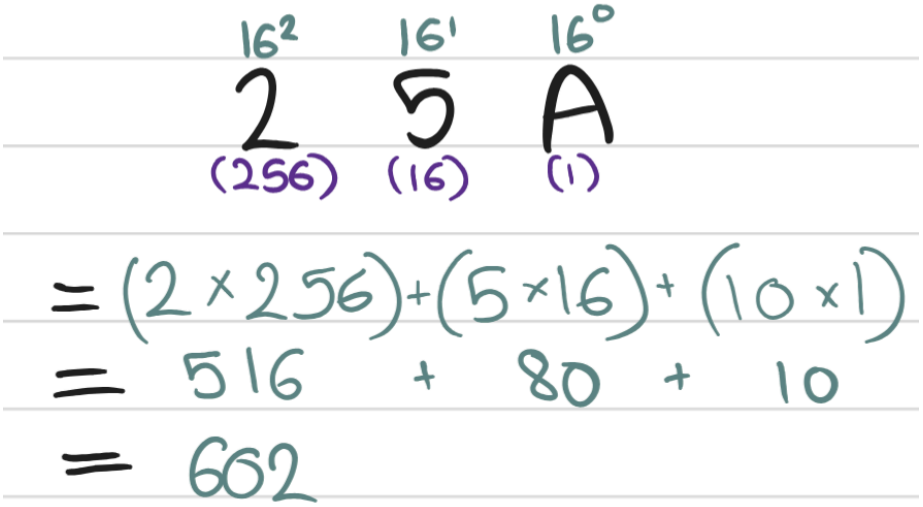
Converting Binary to Hexadecimal:



Converting Hexadecimal to Binary:



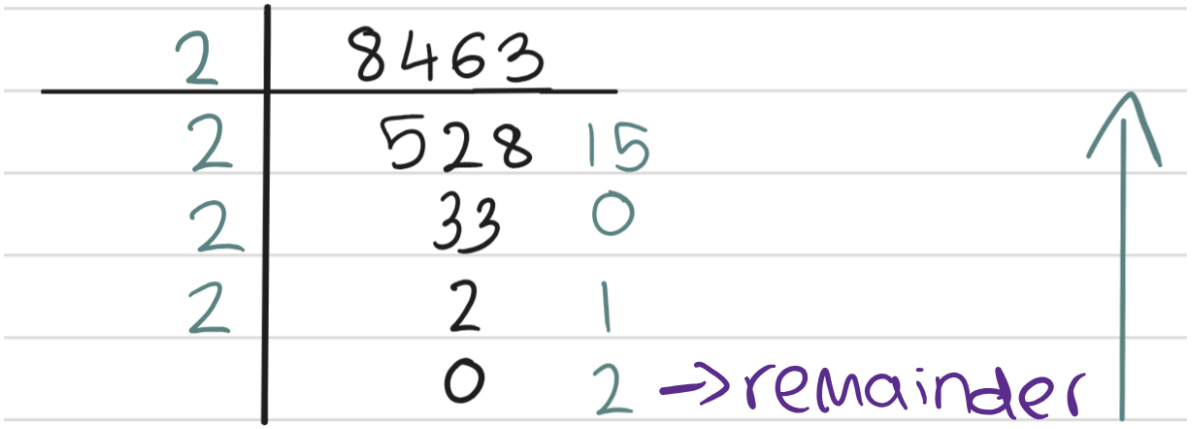
Converting Hexadecimal to Denary:



- First, have each hex digit by its heading value
- Then add all the totals together to get the denary number

Converting Denary to Hexadecimal:

- Divide the number by 16, write the result of the division, including the remainder and keep on dividing until the result is 0.
- Lastly, write down all the numbers in reverse.



Addition of binary numbers

- A table showing the addition of two binary digits

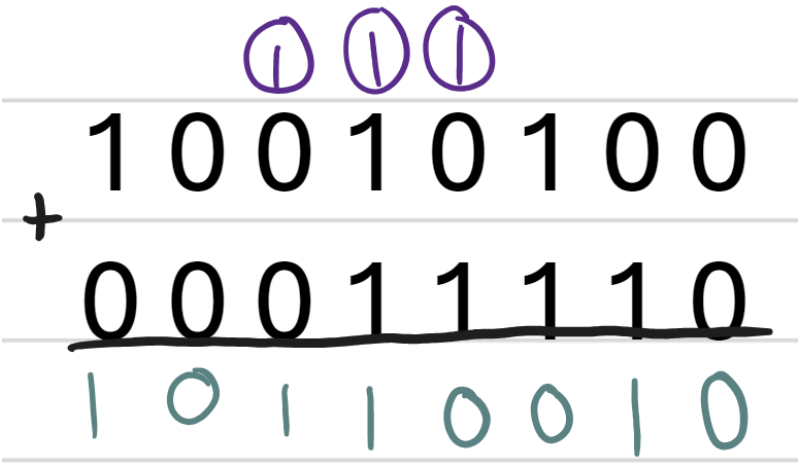
Binary addition	Carry	Sum
0+0	0	0
0+1	0	1
1+0	0	1
1+1	1	0

- A table showing the addition of three binary digits

Binary addition	Carry	Sum
0+0+0	0	0
0+0+1	0	1
0+1+0	0	1
1+0+0	1	0
0+1+1	0	1
1+0+1	1	0

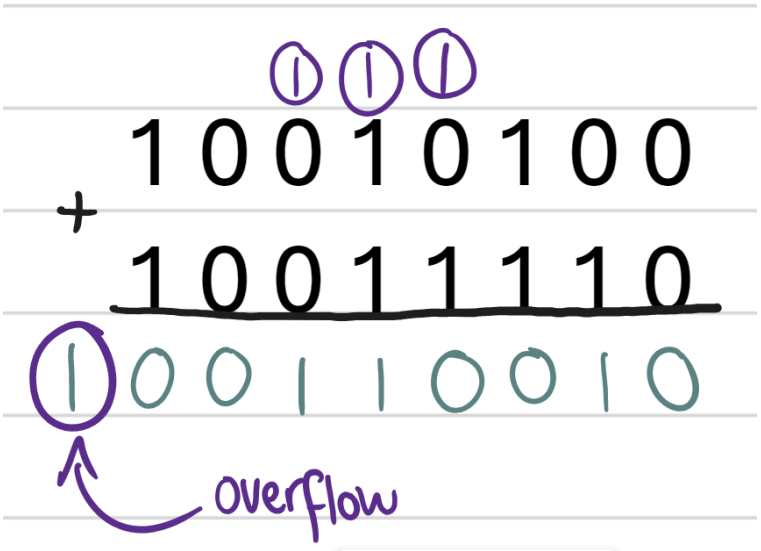
1+1+0	1	0
1+1+1	1	1

- Example of binary addition:



Overflow

- The result of carrying out a calculation that produces a value that is too large for the computer's allocated word size
- Example of overflow (on an 8-bit addition)

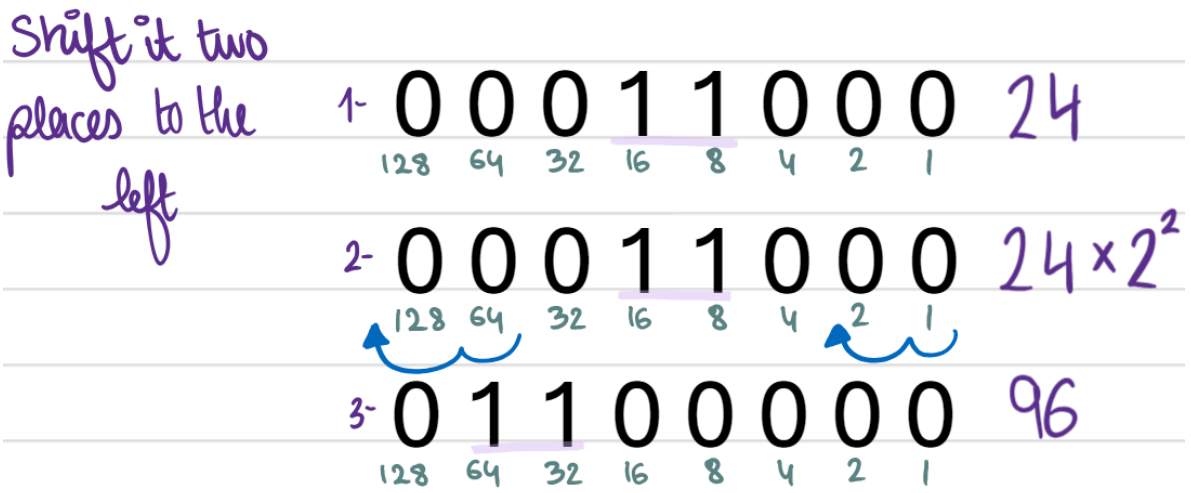


- The formation of a 9-bit value indicates that the sum has exceeded this value

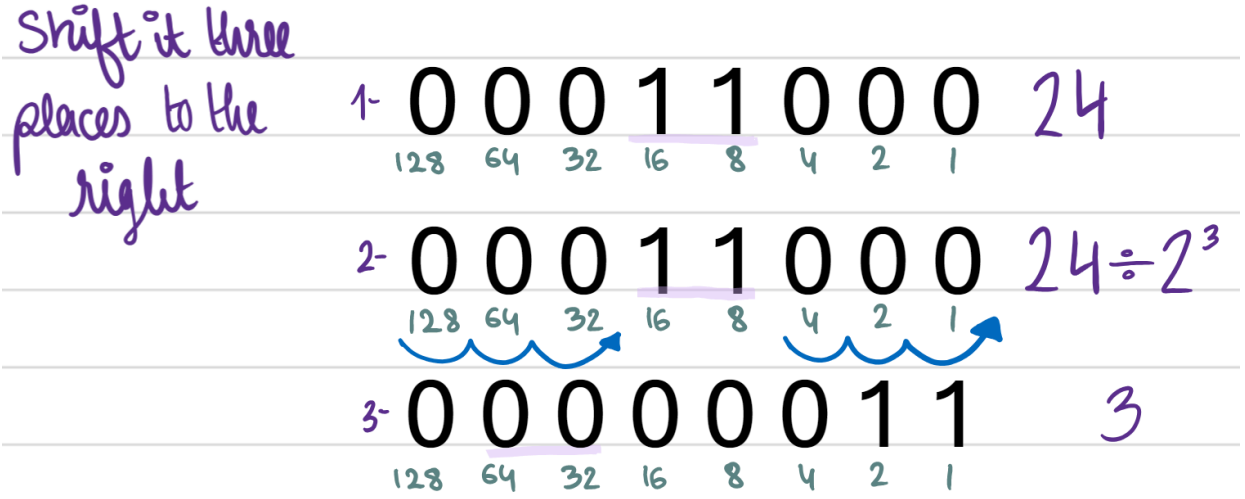
Logical Shifts

- Moving the bits to the left or right in a register
- Any bits shifted out of a register are replaced with zeros

- An example of a binary number shifting to the left:
  - ★ Each left shift is equivalent to multiplying the binary number by 2



- An example of a binary number shifting to the right:
  - ★ Each right shift is equivalent to dividing the binary number by 2



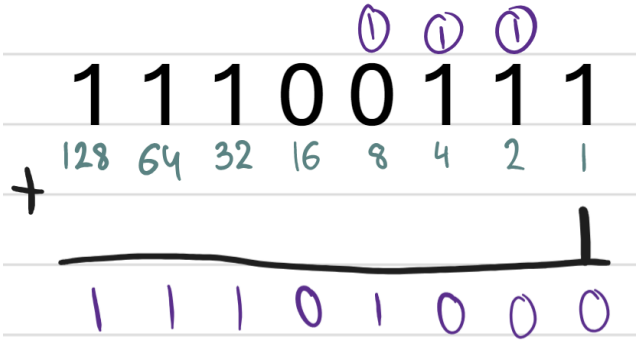
Two's Complement

- A method of representing negative numbers in binary
- Any bits shifted out of a register are replaced with zeros
- Example of Two's Complement with -24

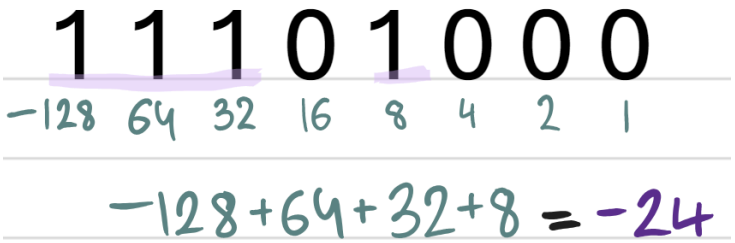
★ First, write the positive value of the given number in binary form (-24 → 24)

★ Flip the number

★ Add the number by 1



★ Add up all the 1 values



Converting a negative Two's Complement number to positive:

- First, write the binary number of the negative number
- Flip the number and add it by 1
- Add up all the 1 values

American Standard Code for Information Interchange (ASCII)

- It was set up in 1963 for use in communication and computer systems
- The extended version was released in 1986
- Only Western languages are supported

Decimal	Hex	Char	Decimal	Hex	Char	Decimal	Hex	Char	Decimal	Hex	Char
0	0	[NULL]	32	20	[SPACE]	64	40	@	96	60	`
1	1	[START OF HEADING]	33	21	!	65	41	A	97	61	a
2	2	[START OF TEXT]	34	22	"	66	42	B	98	62	b
3	3	[END OF TEXT]	35	23	#	67	43	C	99	63	c
4	4	[END OF TRANSMISSION]	36	24	$	68	44	D	100	64	d
5	5	[ENQUIRY]	37	25	%	69	45	E	101	65	e
6	6	[ACKNOWLEDGE]	38	26	&	70	46	F	102	66	f
7	7	[BELL]	39	27	'	71	47	G	103	67	g
8	8	[BACKSPACE]	40	28	(	72	48	H	104	68	h
9	9	[HORIZONTAL TAB]	41	29	)	73	49	I	105	69	i
10	A	[LINE FEED]	42	2A	*	74	4A	J	106	6A	j
11	B	[VERTICAL TAB]	43	2B	+	75	4B	K	107	6B	k
12	C	[FORM FEED]	44	2C	,	76	4C	L	108	6C	l
13	D	[CARRIAGE RETURN]	45	2D	-	77	4D	M	109	6D	m
14	E	[SHIFT OUT]	46	2E	.	78	4E	N	110	6E	n
15	F	[SHIFT IN]	47	2F	/	79	4F	O	111	6F	o
16	10	[DATA LINK ESCAPE]	48	30	0	80	50	P	112	70	p
17	11	[DEVICE CONTROL 1]	49	31	1	81	51	Q	113	71	q
18	12	[DEVICE CONTROL 2]	50	32	2	82	52	R	114	72	r
19	13	[DEVICE CONTROL 3]	51	33	3	83	53	S	115	73	s
20	14	[DEVICE CONTROL 4]	52	34	4	84	54	T	116	74	t
21	15	[NEGATIVE ACKNOWLEDGE]	53	35	5	85	55	U	117	75	u
22	16	[SYNCHRONOUS IDLE]	54	36	6	86	56	V	118	76	v
23	17	[END OF TRANS. BLOCK]	55	37	7	87	57	W	119	77	w
24	18	[CANCEL]	56	38	8	88	58	X	120	78	x
25	19	[END OF MEDIUM]	57	39	9	89	59	Y	121	79	y
26	1A	[SUBSTITUTE]	58	3A	:	90	5A	Z	122	7A	z
27	1B	[ESCAPE]	59	3B	;	91	5B	[	123	7B	{
28	1C	[FILE SEPARATOR]	60	3C	<	92	5C	\	124	7C	|
29	1D	[GROUP SEPARATOR]	61	3D	=	93	5D	]	125	7D	}
30	1E	[RECORD SEPARATOR]	62	3E	>	94	5E	^	126	7E	~
31	1F	[UNIT SEPARATOR]	63	3F	?	95	5F	_	127	7F	[DEL]

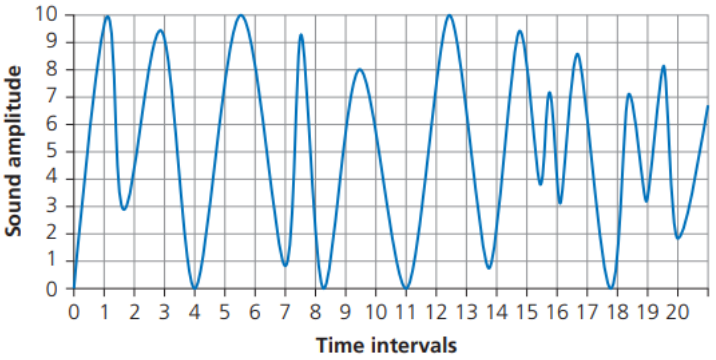
Standard ASCII	Extended ASCII
Character set consists of 7-bit codes	**Character set consists of 8-bit codes**
0-127 values	**0-255 values**
Only supports English	**Supports Latin languages**
	Includes graphical characters

Unicode

- It was set up in 1991
- It represents all languages
- Only Western languages are supported
- Its character set consists of 16-bit codes and approximately 65000 characters

Sound

- Sound waves vary continuously, hence it's analogue
- To store the waves in a computer, they're sampled via an Analogue to Digital Convertor (ADC)
- Sound waves are sampled at regular time intervals.



Keywords

Sampling:
★ Measuring the amplitude of sound waves via an ADC

Sampling resolution/bit depth:
★ The number of bits per sample

Sampling rate:
★ The number of sound samples taken per second - measured in Hertz

How is sampling used to record a sound clip?

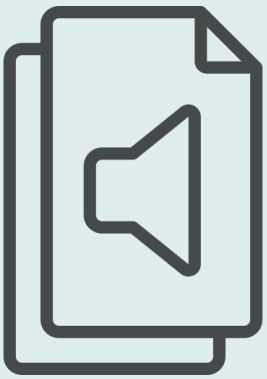
- The amplitude of the sound wave is first determined at a set time interval
- This gives an approximate representation of the sound wave
- Each sample of the sound wave is then encoded as a series of binary digits

Benefits (of a larger sampling resolution)	Drawbacks (of a larger sampling resolution)
Larger dynamic range	**Produces larger file size**
Better sound quality	**Takes longer to download music files**
Less sound distortion	**Requires greater processing power**

Formula to calculate a mono audio file size:
★ Sample rate x Sample resolution x Sample length
(Hz) (bits) (seconds)

To convert it to byte:
★ Divide it by 8

To calculate the stereo sound file:
★ Multiply the result by 2



Images

Keywords

Bitmap image:
★ Images that are made up of pixels

Image resolution:
★ The number of pixels that make up an image

Colour depth:
★ The number of colours to represent each pixel

What are the features of high-resolution images:

★ Consists of many pixels to represent images; there's a lot of detail

★ The file size increases

★ The time taken to download an image file also increases

Formula to calculate an image file size:

★ Image resolution (pixels) x Colour depth (bits)

To convert it to byte:

★ Divide it by 2

Measurement of data storage

- **Bit**:
  - It is the smallest unit of data that a computer can process and store
- **Byte**:
  - It consists of 8 bits
- **Nibble**:
  - It consists of 4 bits

- ICE memory size system

Name of memory size	Number of bytes	Equivalent denary value (bytes)
1 kibibyte (1 KiB)	2^{10}	**1 024**
1 mebibyte (1 MiB)	2^{20}	**1 048 576**
1 gibibyte (1 GiB)	2^{30}	**1 073 741 824**
1 tebibyte (1 TiB)	2^{40}	**1 099 511 627 776**
1 pebibyte (1 PiB)	2^{50}	**1 125 899 906 842 624**
1 exbibyte (1 EiB)	2^{60}	**1 152 921 504 606 846 976**

Data compression

- **It is necessary to compress data:**
  - ★ To save storage on devices (like HDD/SSD)
  - ★ It will take less time to transfer the file from one device to another
  - ★ It will take less time to stream a music or video file
  - ★ Reduces the time taken to upload or download files
  - ★ Not as much bandwidth is needed to transfer the file over the internet
  - ★ Reduced file size also reduce costs

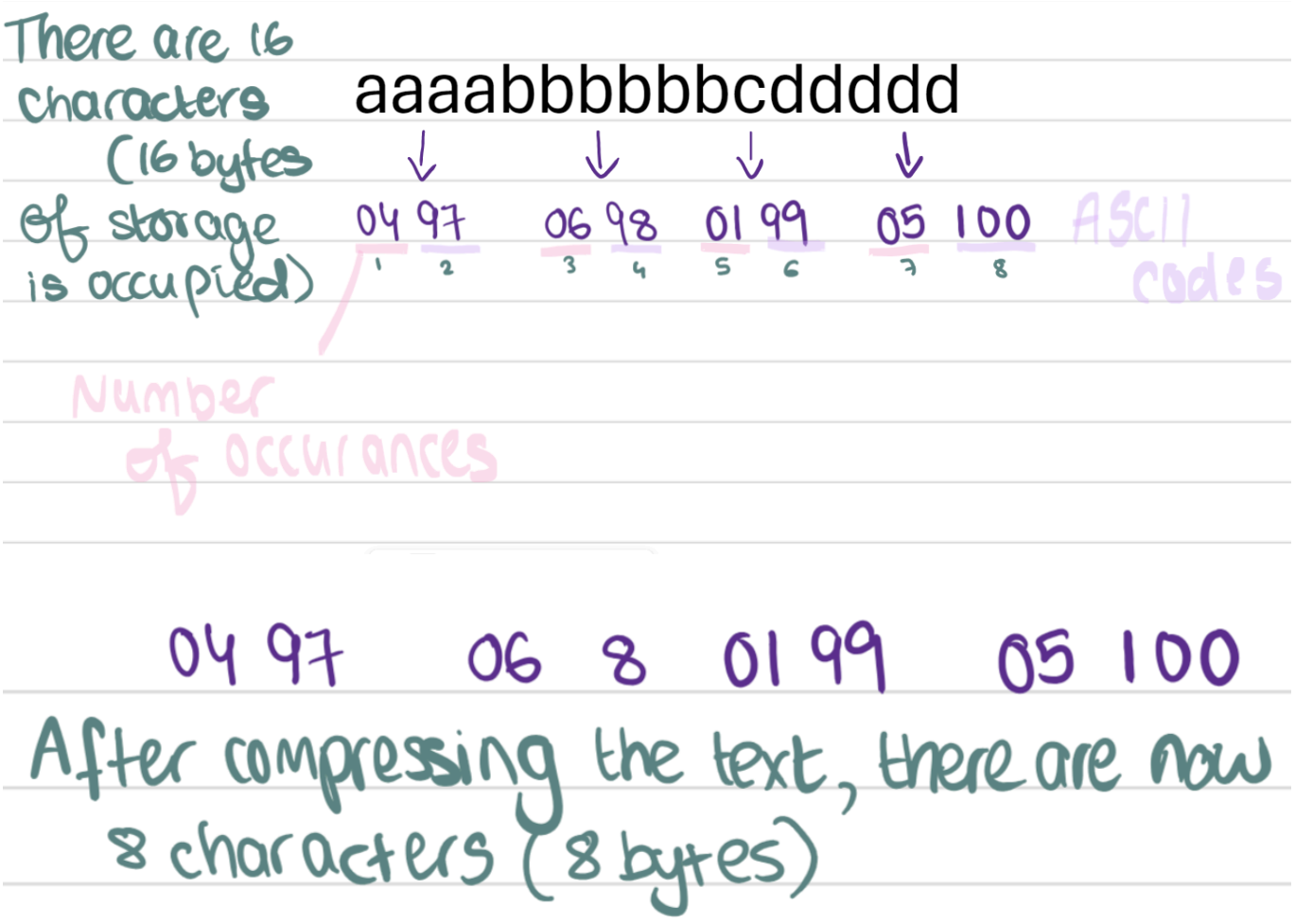
- There are **two** methods of data compression
  - ★ **Lossy:**
    - A compression method that reduces the size of a file by permanently removing data
    - Common lossy file compression algorithms are
      - MPEG-3:
        - Used for playing music
        - Reduces 90% of the audio file size
        - Quality isn't as sharp as the ones found in CDs/DVDs
      - MPEG-4:
        - Used for playing videos
        - Still retains an acceptable quality of sound and videos
      - JPEG:

- Used for bitmap image files
- It separates pixel colour from brightness

★ **Lossless:**

- A compression method that reduces the size of a file by temporarily altering the data
- Common lossy file compression algorithms are
  - Run-Length Encoding:
    - An algorithm that groups repeating patterns and indexes them.
    - Used to compress text and image files

Reduces the size of a string of adjacent, identical data



Data Transmission

A typical packet is split up into:

- **Header Packet**
  - ★ It consists of:
    - The IP address of the sender and the receiver
    - The sequence number of the packet
    - The size of the packet
- **Payload**
  - ★ It consists of:
    - The actual data in the packets (around 64KiB)
- **Trailer**
  - ★ It consists of:
    - A packet identification method
    - An error-checking method **(CRC)**

Keywords

Data packet:
★ Data that is transmitted over a network

Packet switching:
★ A method of transmission in which a message is broken into many data packets which can then be sent along pathways independently of each other

---

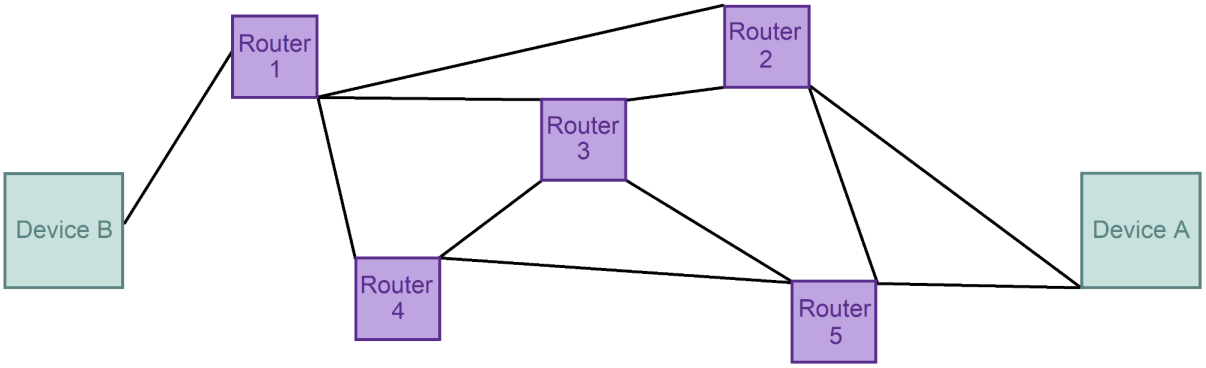
Cyclic Redundancy Check:

- The sender adds up all the 1-bits in the payload and stores this as a hex value in the trailer before it is sent
- Once the packet arrives, the receiver recalculated the number of 1-bits in the payload
- The computer then checks this value against the one sent in the trailer
- If both values match, there's no error, otherwise the packet needs to be re-sent

---

Packet Switching:

- Each packet will follow its path
- Routers will determine the path of each packet
- Routing selection depends on the number of packets waiting to be processed at each node
- The shortest possible path available is always chosen
- Packets can reach the destination in a different order to that they were sent



Different methods of data transmission:

- **Serial:**
  - ★ a transmission method where data is sent one bit at a time down a single wire.

Advantages	Disadvantages
Less chance of data being skewed	**Slower rate of data transmission**
Less chance of error in the data	**Additional data may need to be sent to indicate to the receiving device when the data transmission has started and stopped. (Start and stop bit)**
Cheaper to manufacture; only one wire is needed for serial data transmission	

- **Parallel:**
  - ★ a transmission method where data is sent multiple bits at a time down multiple wires

Advantages	Disadvantages
Faster rate of data transmission	**Higher chance of data being skewed**
Common method of data transmission among devices	**Higher chance of error in the data**
	More costly; multiple wires are used

- **Simplex:**
  - ★ a transmission method where data is transmitted in a single direction only

- **Half-Duplex:**
  - ★ a transmission method where data is transmitted in both directions, but only one direction at a time

- **Full-Duplex:**
  - ★ a transmission method where data is transmitted in both directions simultaneously

---

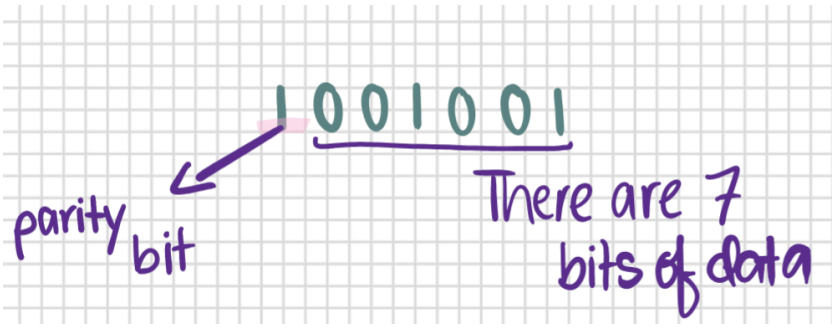
Universal Serial Bus (USB)

- It is an industry standard that is used to transmit data
- **A USB interface includes a:**
  - ★ **Port**:
    - a socket that is a part of a device or computer that enables you to insert a USB cable.
  - ★ **Cable**:
    - a type of transmission media that uses the USB method to transmit data.
  - ★ **Connection**:
    - a collective name for using a USB cable plugged into a USB port to transfer data from one device to another.
  - ★ **Device**:
    - the name of a device that plugs into a USB port on a computer.
- It uses a special type of serial data transmission connection that is used that is designed to transmit data at high speeds.

Advantages	Disadvantages
High rate of transmission	**The length of a USB cable is limited (up to 5 metres)**
It is a simple interface; the cable to device can only fit into the USB one way	**Rate of transmission isn't as high as compared to other types of connection like Ethernet**
A USB port is included in many devices as it's an industry standard	
When inserted into a USB cable, it is automatically detected by the device	
No need for another power source as it can be used to power a device	

Methods of error detection:

- **Errors can occur during data transmission due to interference, leading to:**
  - ★ data loss
  - ★ data gain
  - ★ data change
- **There are several methods of detecting errors:**
  - ★ Parity check:
    - a type of error detection method that adds an additional bit to each byte to create an odd or even sum.
    - **Before transmission:**
      - the parity check is set to be either odd or even parity.



- If an **odd** parity check is used
    - then a **1** is added as a parity bit. This is because all the ls in the byte now add up to an **odd number**
- If an **even** parity check is used
    - then a **0** is added as a parity bit. This is because all the ls in the byte now add up to an **even number**
- **After transmission:**
    - the receiving device will check each byte of data for errors.
    - if an odd parity check has been used and the device finds a byte that has an even number of 1s, then it knows that an error has occurred with this byte of data.
    - the same can be said if an even parity check has been used and the device finds a byte that has an odd number of 1s.

★ **Checksum**:

- A type of error detection method that performs a calculation on the data to create a checksum value.
- A checksum is calculated and added to the data before transmission.
- After receiving the data, the receiving device calculates its own checksum.
- If the two checksums match, the data is error-free; if they don't, an error is detected.

★ **Echocheck**:

- A type of error detection method that sends a copy of the transmitted data back to the sender to be compared with the original data sent.
- The sending device compares the data it sent to the data it has received back from the receiving device to see if they match.
- If they do, then no error has occurred.
- If they don't match, then the sending device knows the data was received with error

---

Automatic Repeat Request (ARQ):

- A type of error detection method that uses acknowledgement and timeout to see if data has arrived correctly after transmission.
- **In a positive acknowledgement method:**
    - ★ The sending device transmits the first data packet.
    - ★ The receiving device receives the data and checks it for errors.
    - ★ Once the receiving device knows it has received the data error free, it sends a positive acknowledgement back to the sending device.

- ★ When the sending device receives this positive acknowledgement, it knows the receiving device has received the data packet error free and it sends the next data packet.
- ★ If the sending device does not receive a positive acknowledgement within a set timeframe, a timeout occurs.
- ★ When a timeout occurs, the sending device will resend the data packet. It will keep doing this when a timeout occurs, until it receives a positive acknowledgement
- ★ Sometimes a limit (such as 20 times) is set and when this limit is reached it will stop resending the data.

- **In a negative acknowledgement method:**
  - ★ The sending device transmits the first data packet.
  - ★ The receiving device receives the data packet and checks it for errors .
  - ★ If the receiving device detects no errors, no further action is taken .
  - ★ If the receiving device does detect errors, it will send a negative acknowledgement back to the sender.
  - ★ If the sender receives a negative acknowledgement, it knows this means the data was received incorrectly, so it can resend the data packet.
  - ★ A timeout is set by the sending device when it sends the data. This is just so that the sending device knows that if it doesn't receive a negative acknowledgement back within that set time period, it doesn't need to be still waiting for it and can send the next data packet.

Check digit:

- A type of error detection method that is used for data entry.
- A calculation is performed on the data entered to create a value.
- Check digit values are compared to see if the data entered is correct.
- **This is how a check digit operates:**
  - ★ A check digit is pre-calculated and stored with the data (e.g., barcode or ISBN).
  - ★ When the data is entered or scanned, the check digit is recalculated.
  - ★ If the recalculated check digit matches the stored one, the data is correct.
  - ★ If the recalculated check digit does not match the stored one, the data is incorrect.

Encryption:

- A method of securing data for storage or transmission that scrambles it and makes it meaningless.
- **There are two methods of encryption:**
  - ★ Symmetric encryption
  - ★ Asymmetric encryption

Keywords

Plain text:
★ The name given to data before encryption.

Cipher text:
★ The name given to data after transmission.

Encryption key:
★ A type of algorithm that is used to encrypt data.

Symmetric Encryption:

- A type of encryption that uses the same key to encrypt and decrypt data.
- **The process is:**
  - ★ Plain text is encrypted into cipher text using an encryption key.
  - ★ The cipher text and the encryption key are sent separately to the receiving device.
  - ★ The same key is then used to decrypt the ciphertext back into its plain text form.

Asymmetric Encryption:

- A type of encryption that uses two different keys (public and private keys) to encrypt and decrypt data.
- **The process is:**
  - ★ Plain text is encrypted into cipher text using a public key. (This is also a type of encryption algorithm.)
  - ★ The cipher text is transmitted to the receiving device.
  - ★ The cipher text cannot be decrypted using the public key, it is decrypted using a private key.

Hardware

The CPU: Central Processing Unit

CPUs are used for processing and executing instructions and data that are input into the computer so that the result can be output.

A microprocessor is a type of integrated circuit on a single chip that usually performs less functions as compared to a regular CPU.

Components of a CPU

A CPU consists of:

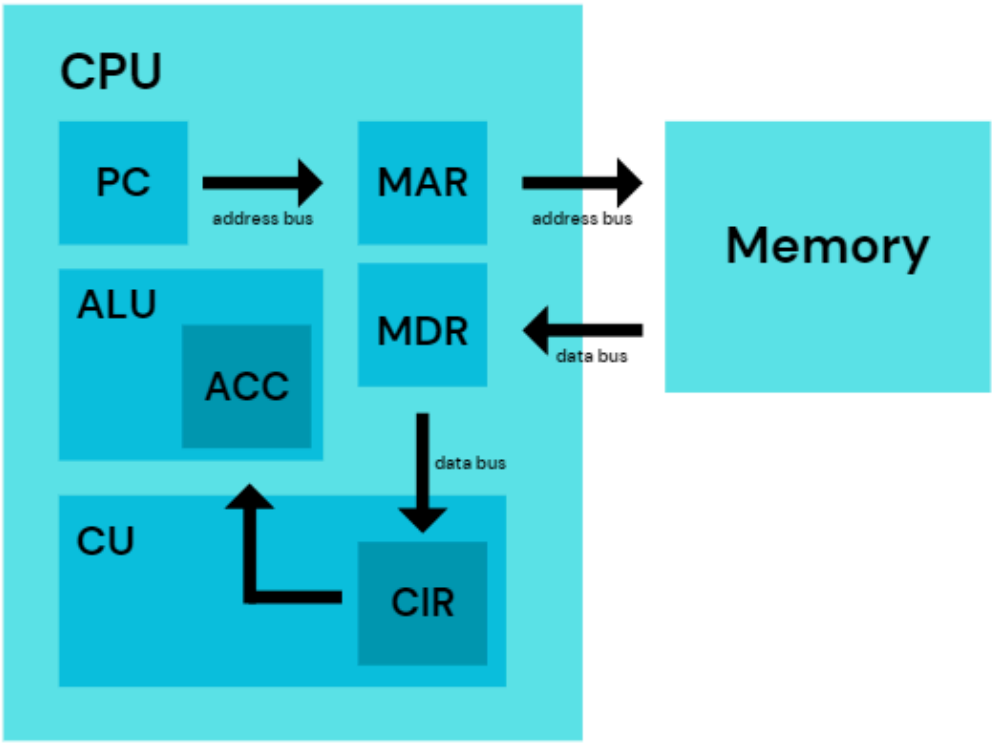
- Units:
  - Examples of units:
    - Arithmetic Logic Unit (ALU): The ALU deals with all arithmetic operations such as addition, subtraction, etc. It also handles logical operations that need to be performed such as "AND", "OR", "NOT", etc.
    - Control Unit (CU): The CU has the job of managing and synchronising data and instructions transfer in the CPU. It does this by sending control signals to other components. Additionally, it also decodes instructions sent to it.

- Registers:
  - Registers are high-speed memory components in the CPU that temporarily store important information.
  - Examples of registers:
    - Program Counter (PC): Stores the memory address of the next instruction that will be processed
    - Memory Address Register (MAR): Stores the memory address of the instruction/data that is about to be fetched from the RAM
    - Memory Data Register (MDR): Stores data or instructions that have just been fetched from the RAM
    - Current Instruction Register (CIR): Stores the instruction that is currently being processed or decoded by the CPU
    - Accumulator (ACC): Stores any interim values or final values created during arithmetic or logical operations done by the ALU

- Buses:

- Buses are communication systems within the computer that transfer data and instructions between components
- Examples of buses:
  - Address bus: Carries the memory address of data/instructions in
  - Data bus: Carries actual data/instructions
  - Control bus: Carries control signals to and from the CU

The Fetch-Decode-Execute Cycle



Fetch:

- This stage is for fetching the actual data or instructions from the computer's memory to the CPU.
- Steps:
  1. The Program Counter stores the address of the next instruction to be processed. This is copied to the Memory Address Register via the address bus.
  2. The Memory Address Register copies this address to the RAM via the address bus.
  3. The RAM uses the address sent to it to locate the data or instruction that must be processed.
  4. The RAM copies the data/instruction to the Memory Data Register via the data bus.

5. The Memory Data Register copies the data/instruction to the Current Instruction Register, via the data bus, where it will be processed.
6. The Program Counter is incremented so that the address of the next instruction is contained in it.

Decode:

- This stage is used to decode the data/instruction, which is carried out by the Control Unit using an instruction set
- An instruction set is a set of instructions written in low-level machine code that tells the computer how to carry out specific functions. They are the only instructions understood by the CPU.

Execute:

- This stage is used to carry out any final arithmetic or logical operations and then finish executing the instruction.
- The Arithmetic and Logical Unit is responsible for carrying out any such operations. It used the Accumulator to store any interim values created during execution.
- After all calculations are finished, the command is executed.

The entire process repeats for each piece of data or instruction that needs to be carried out by the CPU.

Increasing the CPU's Performance

Cache:

Cache is internal, temporary storage space inside the CPU itself. It stores data and instructions which are used regularly by the CPU.

The CPU does not need to fetch regularly used data or instructions from the RAM, instead, it simply uses the cache. This takes less time and increases the CPU's performance.

The more cache a computer has, the more data/instructions it can store nearby, which means less time is spent retrieving information. So, a larger cache size can mean better CPU performance.

However, an extremely large cache would take more time to look through while fetching information, causing significant delays which would render the cache useless. So, there must be a limit for the cache size.

Clock:

The System Clock is located inside the CPU. Its job is to control the processing speed and time taken to execute instructions, as well as control the time intervals between each task and synchronise the entire FDE cycle. In short, it controls the number of FDE cycles completed each second.

Clock speed is measured in Hertz (instructions / second). The average clock speed found in most computers is 2 Gigahertz, which is 2 billion instructions executed per second.

Higher clock speed means that more instructions are processed each second, increasing the computer's performance.

However, increasing the clock speed too much can cause the computer to overheat.

Cores:

Cores are the components in the computer that run the FDE cycle. Each core can run one cycle, i.e process one instruction, at a time.

A higher number of cores in a computer means that more instructions can be processed at once. This allows a computer to multitask, which increases the computer's performance.

Embedded Systems

An embedded system is a combination of software and hardware that is built into a device, which runs on firmware and performs a dedicated function.

They differ from general purpose computers as they only perform a specific task, instead of being multipurpose.

Embedded systems are used in:

- domestic appliances
- Cars
- security systems
- lighting systems
- vending machines

Advantages	**Disadvantages**

• They are small • They are cheap • They can be remote-controlled • No operating system is needed • They use very less power	• They are difficult to update • They are susceptible to hacking • Specialists are needed for repairs • There are environmental concerns regarding their disposal • They may have a confusing interface

Input Devices

Input devices allow data to be entered into the system.

Examples of input devices:

- Barcode scanner
- Digital camera
- Keyboard
- Microphone
- Optical mouse
- QR code scanner
- Touch screen
    - Resistive
    - Capacitive
    - Infra-red
- Two-dimensional (2D) and three-dimensional (3D) scanners

Output Devices

Output devices are used to tell user information.

Examples of output devices:

- Actuator
- Digital light processing (DLP) projector
- Inkjet printer
- Laser printer
- Light emitting diode (LED) screen
- Liquid crystal display (LCD) screen
- Liquid crystal display (LCD) projector
- Speaker
- 3D printer

Sensors

Sensors are input devices that capture analogue data from their immediate environment at set time intervals.

Sensor	Data Recorded	Uses
Acoustic	Sound levels of an environment	Security systems - to detect intruders
Accelerometer	Any forces of acceleration acting on the environment	Earthquake warnings
Flow	The amount of liquid/gas/steam flowing/moving through an environment	Pipes or plumbing
Gas	The amount and concentration of a gas	Carbon monoxide detectors in houses
Humidity	The humidity levels in an environment	Greenhouses - to ensure the internal humidity is not too much or too less
Infra-red	Any infra-red rays present	Security systems
Level	The level or volume a liquid/gas substance is at	Amount of fuel in vehicles
Light	The light levels of an environment	Automatic lighting
Magnetic field	Any magnetic fields present in the environment	Detecting cars in parking lots or bridges
Moisture	The moisture present in the environment/any substances	Detecting moisture levels in soil (agriculture-related)
pH	The pH level of a substance	Water bodies to check level of pollution, **or** detecting soil pH levels to ensure plant health
Pressure	The pressure which is being exerted on the sensor	Security systems to detect intruders
Proximity	How close an object is to the sensor	Used in robots or automatic doors
Temperature	The temperature of the environment	Automatic cooling systems

Data Storage

Primary Storage:

- Primary storage contains data that is directly accessible by the CPU.
- Random access memory (RAM) and read-only memory (ROM) are examples of primary storage.

Secondary Storage:

- Secondary storage contains data that is not directly accessible by the CPU and is not currently required by the CPU.
- It is non-volatile, which means that the data contained in it is not lost if power is switched off – i.e, it stores data permanently until deleted by the user.
- It stores data that can be transferred to other computers if required.
- Pen drives and CDs are examples of secondary storage devices.

Primary Storage: RAM and ROM

RAM:

- It stores data temporarily, making it a volatile method of data storage. It stores the data and instructions that are currently being used by the CPU. For example, program files for an application that is currently open on the user's computer.

ROM:

- It stores data permanently, making it a non-volatile method of data storage. It stores the programs that are required for the computer to boot initially. For example, BIOS (Basic input/output system)

RAM	**ROM**
• Temporary storage (volatile) • Stores the data currently in use by the CPU	• Permanent storage (non-volatile) • Stores programs that initially boot the computer

• Its contents are constantly changing, based on what the CPU requires • It is easy to increase the RAM's size	• Its contents are fixed and are not typically changed • It's difficult to increase or decrease the size of the ROM

Secondary Storage: Magnetic, Optical and Solid-State

Secondary storage needs to be fetched into the primary memory first, to be accessed by the CPU. It can be of two types: internal, such as solid-state drives or hard-disk drives, or external, such as CDs or DVDs.

Magnetic Storage:

- Magnetic storage uses magnetised and demagnetised dots to store data.
- A circle disk, known as a platter, spins very fast, with a read-write head over it. The read/write head magnetises and demagnetises dots on the platter as required.
- The platter is divided into tracks and sectors.
- The current state of a dot (i.e. whether magnetised or not) decides its stored value. For example, all magnetised dots store the value "1", and all demagnetised dots store the value "0".
- An example of magnetic storage is the hard disk drive.

Optical Storage:

- Optical storage uses circular discs and lasers to store data.
- The disc is rotated or spun with a read/write head over it, which uses a laser beam.
- The laser beam makes indentations on the surface of the disc by burning data onto it. An indentation is known as a pit while the space between indentations are known as lands
- The pits and lands represent binary values, for example, a pit represents a "1" while a land represents a "0".
- The data is written in a spiral or circular pattern around the disc.
- Examples of optical storage are CDs, DVDs, and Blu-Rays.

Solid-State Storage

- Solid-state storage is also known as flash storage. It uses transistors and cells to store data, onto which the data is 'flashed'. These transistors and cells are arranged on a semiconductor chip.
- Transistors are laid out to form a grid. They are used to form a control gate, which is above a floating gate, at each intersection in the grid.
- The transistors control the flow of electrons in the semiconductor chip. They flow through the control gate to the floating gate, where they are stored.

- After storing the electrons, the binary value of the transistor changes – i.e, from "1" to "0".
- Typically, a NAND or NOR structure is used for solid-state storage,
- Examples of solid state devices are pen-drives and solid-state drives (SSDs)

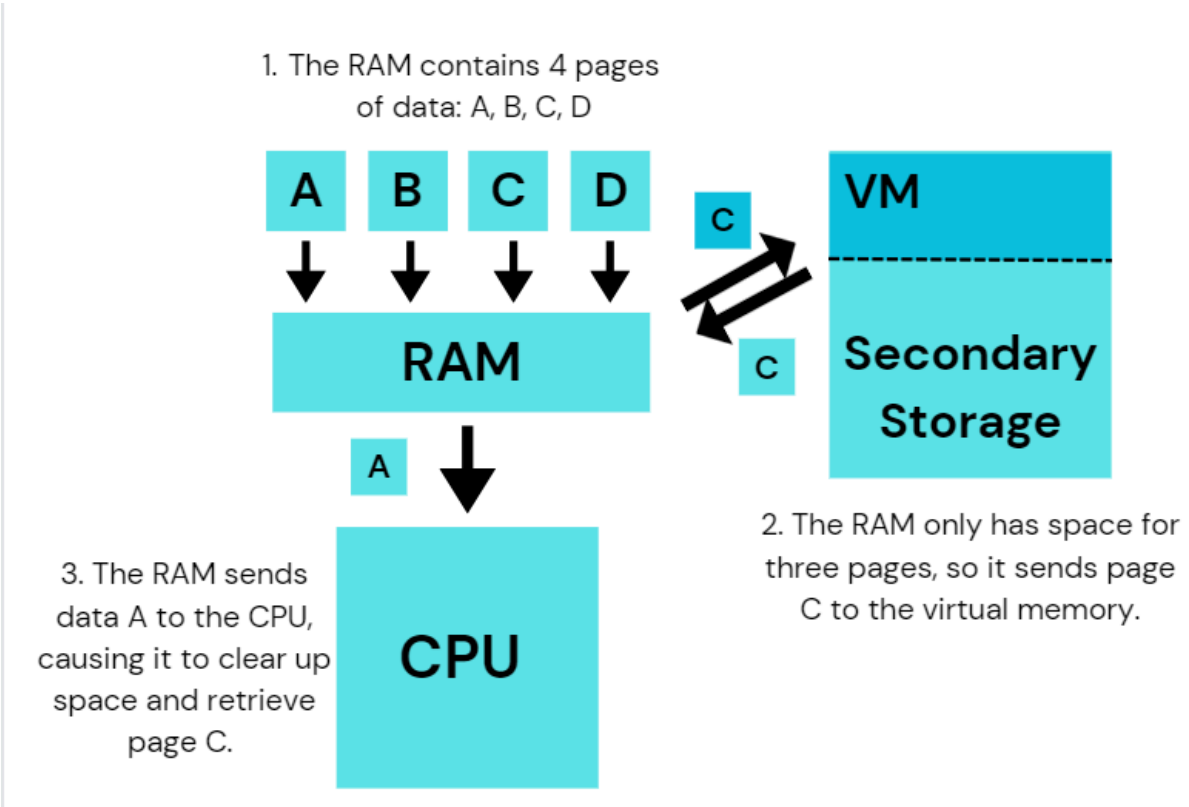
Virtual Memory

Virtual memory is a sectioned-off segment of the hard disk drive, which acts as an extension to the primary memory (RAM). It is required for when the RAM is full but more data transfer needs to take place.

This is how it operates whenever required:

1. Data in the RAM is split into pages
2. The pages which are not currently being used by the CPU or which are not immediately required are sent to the virtual memory. This clears up space in the RAM.
3. Any new data that the RAM now requires is imported from the secondary storage.
4. Once the data in the RAM is no longer needed, it is sent back to secondary storage. The pages the RAM initially sent to the virtual memory are fetched back.

Virtual memory is important as the RAM is at risk of overloading from excess data and crashing. The virtual memory helps ensure that this does not happen.



Cloud Storage

Cloud storage is data stored by a third-party remote server instead of in local storage. It is accessible via the internet.

Cloud Storage	**Local Storage**
• Accessible everywhere, but only if you have an internet connection • There is no hardware cost, and you do not have to manage your own hardware. • Back-up of your data is easily available. • It is easy to increase and decrease the amount of storage space you have • Trusting a third party with personal data may be unsafe, as they could use it for malicious reasons.	• Only accessible as long as you have your storage with you, but doesn't require the internet • You have to pay for your own hardware, as well as keep it secure • Changing the storage size may be difficult. • There is no need to rely on a third-party – you are fully in control of your own data, which means there are less privacy issues

Network Interface Card

The Network Interface Card (NIC) is a hardware component that is required by the computer to access any networks by connecting to routers. It also contains the MAC address.

Media Access Control Address

A Media Access Control (MAC) address uniquely identifies each computer on a network, which is provided by the manufacturer when the computer is created. It is located on the NIC.

Structure:

- It is written in hexadecimal
- NN – NN – NN – DD – DD – DD; the first six digits comprise the manufacturer's code, while the last six digits comprise the device's serial number.
- It is 48 bits in total; 24 bits for the manufacturer's code, and 24 bits for the device's serial number.
- It is usually separated by a dash.

Internet Protocol Address

An Internet Protocol (IP) address is assigned by the internet service provider and it uniquely identifies each computer on a network.

There are two types of IP addresses: static and dynamic.

- Static: The IP address never changes unless specifically requested. All devices with a static IP address are always traceable. A static IP address may be comparatively less safe.
- Dynamic: The IP addresses change when a computer connects to a network. It may change every time the computer connects to a network, or once every few times a computer reconnects. As it is always changing, it is more difficult to trace, thus ensuring better privacy.

Structure:

There are two types of IP addresses – IPv4 and IPv6. They both have the characteristics of a typical IP address, however, their structures differ.

IPv4	**IPv6**
• Usually written in denary digits (base-10) • Each number is separated using dots • There are 4 groups of digits • It is 32 bits – each group is 8 bits • Each number is between 0-255 • There are fewer available unique addresses for IPv4 • Example: 24.2.89.232	• Usually written in hexadecimal • Each number is separated using a colon • There are 8 groups of digits • It is 128 bits – each group is 16 bits • The values are between 0000 - FFFF • There are more available unique addresses for IPv6 • Example: 1234:0A34:1F20:0000:FFFF:998B:900C:5ABC

Software

Types Of Software

Software: A series of instructions written in a programming language that performs a function or a task.

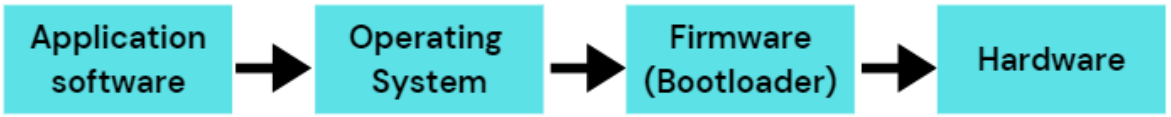
System Software:

- It provides the services that the computer requires, including operating system and utility software (such as anti-virus)
- For example: manages hardware and software, provides a user interface, etc

Application Software:

- It provides the services that the user requires, and runs on the OS.
- For example: word processors, databases, spreadsheet applications, and web browsers.

All application software runs on the operating system. The operating system runs on the firmware, also known as the bootloader. The bootloader runs on the hardware.



The Operating System

The operating system plays many diverse roles in a computer system. It does the following tasks:

- Managing files:
    - Allows the creation and deletion of folders and directories to store files
    - Allows users to create, copy, open, close, move, delete, rename, save and sort files.
- Handles interrupts:
    - Decides the appropriate time to run interrupts in order of requirement
- Provides a user interface:
    - Allows data to be entered by and displayed to the user

    - User interfaces can be: Graphical User Interface (GUI), which uses icons, pointers, etc or Command Line interface, which requires data to be entered as commands in a programming language.
- Managing peripherals and drivers:
    - Manages and detects any peripherals, i.e. any connected input or output devices. It additionally instals appropriate drivers when a device is connected.
    - Driver: Software that is required for communication between two devices externally connected, which translates code from one device to another.
- Managing memory:
    - Manages data transfer to and from the memory (RAM)
    - Makes sure all currently running processes have enough memory space to continue running properly
    - Keeps track of each memory location and records its status (i.e is it occupied, in use, etc)
    - Ensures that two processes don't try and access the same memory location at once
    - Manages transfer of pages between the virtual memory and the RAM
- Managing multitasking:
    - Decides the order in which processes are meant to run, and how long each instruction should spend getting processed
    - Organises a priority queue, to schedule instructions to be processed in order of importance
- Providing a platform for running applications
- Providing system security:
    - Oversees software updates and ensure all software are always up to date
    - Runs anti-virus checks and sets up the firewall
    - Manages data backup and system restoration in case of data loss
    - Manage each user accounts access
- Managing user accounts:
    - Helps each user have a separate username, password
    - Restricts access for each user as required
    - Keeps each users data separate

Interrupts

Including: – how an interrupt is generated – how it is handled using an interrupt service routine – what happens as a result of the interrupts

Interrupt: A signal that is sent to the CPU asking for its attention. These signals allow the CPU to multitask.

An interrupt is generated when something demands the CPU's urgent attention. Hardware and software can both send interrupts. The CPU checks for these interrupts at the end of each FDE cycle.

Examples of hardware interrupts	**Examples of software interrupts**
• Pressing a key on the keyboard • Moving the mouse • Hardware failure/errors	• Division by zero • Two processes trying to access the same memory location • Input required

Interrupts can be of different priority levels. For example, an interrupt sent due to hardware failure would be of greater importance than an interrupt sent when the user requires input. The interrupts are organised into a priority queue based on their level of importance. A CPU's interrupt handler organises the interrupts.

To handle interrupts, the CPU calls an interrupt service routine. An interrupt service routine's job is to retrieve the interrupt and perform the actions required for it to be solved.

When the CPU receives an interrupt, it follows the following steps:

1. The current FDE cycle is finished and the CPU checks the interrupt queue.
2. If an interrupt with higher priority than the next process is found:
    a. The next process is stored
    b. The interrupt's source is found
    c. An appropriate interrupt service routine is called, which handles the interrupt.
3. After the interrupt is handled, the CPU once again checks for high-priority interrupts. If any are found, they are serviced.
4. The next process is retrieved and another FDE cycle takes place.

This process takes place continuously.

Programming Languages

There are two types of programming languages: high-level language and low-level language.

High-Level languages use English-style words, which are easier to understand for humans. They need to be converted into low-level language to be understood by computers. Python, Java, and C# are examples of high-level languages.

Low-level languages have two types: assembly language and machine code.

- Assembly language uses mnemonics in place of words. For example, a high-level language may store a variable using “var num1 ← 100” may be written as “STR 100” in assembly language. It can directly access a computer’s hardware using commands, such as store variables in specific memory locations. It must also be translated into machine code before it can be understood by a computer, however, conversion is less time consuming.
- Machine code uses 0s and 1s to represent data in the form of binary numbers. It does not need any conversion.

High-level language	**Low-level language**
• They are machine independent, which means that the code can run on any computer regardless of its model. • The code is much easier to read and debug since it is written in English words • The code cannot directly manipulate a computer’s hardware in any way. • The code must always be translated to low-level language first, which can be time consuming.	• They are machine dependent, which means that the code is for a specific computer only and may not run on other computers. • Both assembly language and machine code are difficult to read as well as debug. • The code can manipulate computer hardware if required. • Only assembly language requires translation, which is less time taking. Machine code does not require any translation.

Translators

Translators are required to convert high-level language or assembly language into machine code, which is the only language which can be understood by the computer. There are three types of translators: an assembler, an interpreter, and a compiler.

Assemblers translate assembly language into machine code. It translates the entire code at once and produces a file. If it finds any errors, it will create an error report and display it once the entire code is read, without producing a file.

Interpreters and compilers translate high level languages into machine code. Their methods of reading the code and finding errors are different.

An interpreter reads the code line-by-line. If it finds an error, it will stop immediately and display it to the user. It will not continue to read the code until the error has been corrected. Otherwise, it runs the code. They are better to use while in the process of writing code, as it lets you correct errors as they are made.

A compiler reads the entire code at once. If it finds any errors, it will compile an error report of all the errors and display it to the user. Otherwise, it will run the code and create an executable file. They are better for when the code is complete, as it creates an executable file that can be run easily anywhere.

interpreter	**Compiler**
• Translates and executes the code line-by-line • Stops execution when an error is found • Are better to use while in the process of writing code, as it lets you correct errors as they are made • Needs to be re-translated each time it needs to be fun • The source code is always required for it to be run • The interpreter software is always needed for code to run	• Translates the whole code at once before executing it, producing an executable file • Provides an error report for the whole code if errors are detected • Better for when the code is complete, as it creates an executable file. • Doesn't need to be re-translated each time it needs to be fun, as an exe file is produced • The source code is no longer required after the executable file is produced • The compiler software is no longer needed after the exe file is produced

Integrated Development Environment

An Integrated Development Environment is an interface where users can write, edit and run programs easily. An IDE always contains:

- An editor: where code can be written and edited.
- A runtime environment: where the code's outputs are shown, and user input is allowed.

An IDE typically also contains other useful features, such as:

- A compiler or interpreter
- Error diagnostics
- Auto-completion of code
- Auto-correction of code
- Prettyprint, which colour-codes keywords

The Internet And Its Uses

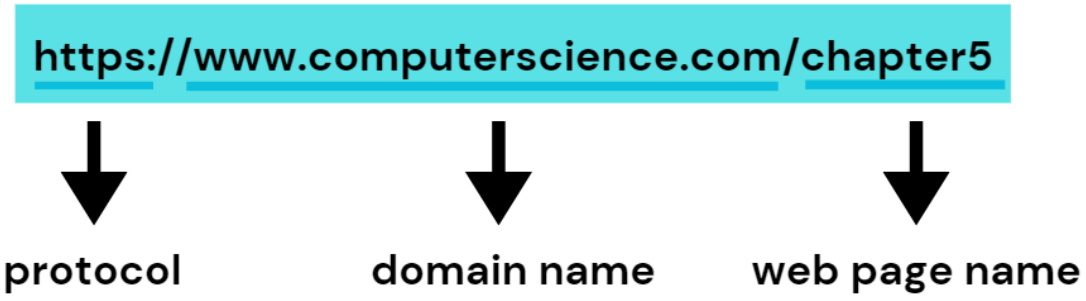
The Internet and the World Wide Web

The internet is a network of computers joined together through infrastructure such as wires and cables. It is a type of network that spans the entire world, connecting computers and enabling them to use the world wide web.

The world wide web is the collection of websites and web pages accessed *using* the internet's infrastructure. It is primarily software-based, as opposed to hardware infrastructure.

Uniform Resource Locator

Websites and webpages on the world wide web must be accessible easily. Uniform resource locators exist for this purpose. A URL is a text-based address to uniquely identify a web page. It contains three sections: the protocol, the domain name and the web page/file name.



Web Browsers

A web browser is used mainly for the purpose of rendering hypertext markup language (HTML) and displaying web pages to the user. It typically has other functions as well, such as:

- Storing bookmarks and favourites
- Recording user history
- Allowing the user to open multiple tabs
- Storing cookies

- Providing navigation tools such as a button to reload, go back to the previous/next page
- Providing an address bar

Retrieval of Web Pages

Web pages are stored in web servers. For a web browser to access their HTML, retrieve it from the server and render it, it first requires the web server's IP address. This address is stored in a domain name server, to which a request must be sent first.

The steps to retrieving web pages are:

1. The user types the URL of the website into their web browser's address bar.
2. The web browser sends the URL to a domain name server. The DNS stores IP addresses of each web server.
3. The DNS searches in its database for the IP address corresponding to the URL sent by the user. If the IP address is not found in one DNS, it is sent to another DNS until it is found. If the URL's corresponding IP address is not found at all, an error message is displayed to the user.
4. When the DNS finds the corresponding IP address, it sends it back to the web browser.

After this, the HyperText Transfer Protocol - HTTP- is executed by the browser to retrieve web pages from their web server and then render them. The protocol consists of the following steps:

1. The web browser sends a request to the web server asking for the web pages by using its IP address to locate it.
2. The web server sends back all HTML, CSS and active script (such as JavaScript).
3. The browser renders the code it receives.

HyperText Transfer Protocol Secure - HTTPS

During the transmission of data to and from the web server, there is a risk of it being hacked and stolen. Personal information such as passwords may be intercepted and used for malicious purposes. The standard HTTP protocol does not protect data being transmitted. So, HyperText Transfer Protocol Secure – HTTPS – is used instead, as it uses digital certificates to ensure security.

A digital certificate is a certificate that is awarded to a certain web server as proof of their authenticity. It shows that the website encrypts all its transmitted data. It is awarded by the certificate authority, which is an organisation which checks the authenticity of a website.

Typically, a website ensures encryption using the Secure Socket Layer(SSL) protocol.

The HTTPS protocol secures its data using the following steps:

1. Before the web browser asks for the web pages, it sends a request to the web server asking for its digital certificate.
2. If the web server has one, it sends it to the web browser.
3. The browser confirms if the certificate is authentic. If it is, web page transmission takes place in an encrypted manner. Otherwise, a message displaying that the website is unsafe is displayed to the user.

Users can check if a website is secure by:

- Ensuring the URL contains “HTTPS” and not “HTTP”
- Checking if there is a padlock symbol near the URL

Cookies

When a user visits a website, they may enter specific information that needs to be entered again at later dates. For example, passwords and usernames. However, it is an inconvenience to the user if this information must be entered again each time. The web server uses cookies to store such information.

A cookie is a small text file that stores information about the user. There are two types of cookies: session and permanent.

They are created by:

1. The web server sending an empty cookie file to the browser
2. The user’s details are then stored in the encrypted file, which is then stored in the secondary storage or primary storage.

They are retrieved by the web server by:

1. Whenever the website is opened or revisited, the web serving sending a request for the cookie file
2. The browser sends the cookie file so that details are automatically entered.

Session cookies are temporarily stored in the primary memory. They are only used as long as the browser is still open. As soon as the browser window is closed, the cookie is deleted.

Persistent cookies are permanently stored in secondary storage. They are stored until they are deleted by the user, or until they reach a specific expiration date. They're retrieved each time the browser window is opened.

Cookies can be used for:

- Saving personal details such as passwords and usernames
- Tracking user preferences
- Holding items in an online shopping cart
- Storing login details

Cookies can have certain privacy issues, as they store sensitive information. They are:

- Cookies may store data without the user's knowledge or permission
- Hackers can intercept the data and build a false user profile, leading to identity theft
- Sensitive information in the cookie files may be intercepted during transmission by a 3rd party
- Other websites may gain unauthorised access to certain cookies
- The computer may be hacked and the cookie file's information may be visible to a 3rd party.

Digital Currencies

Digital currency is a form of payment that is completely electronic and does not require any tangible currency, such as coins or notes.

Most digital currencies and tangible currencies both use a centralised system.

A centralised system uses a central authority that oversees all transactions made by users, such as a bank. It can be susceptible to security breaches, and there are confidentiality risks in using such a system. This is often why crypto currencies are used.

Crypto currencies are a subtype of digital currencies which use a decentralised system known as the blockchain. There is no central authority involved, thus making it safer and more confidential.

Blockchains:

A blockchain is a digital ledger or database of all transactions made on its network, which is publicly available for all devices on the network.
Each transaction made on a blockchain is stored in a block, which contains:

- A unique identifier for the transaction
- The sender, receiver, and amount of money sent during the transaction

- The previous block's unique identifier.

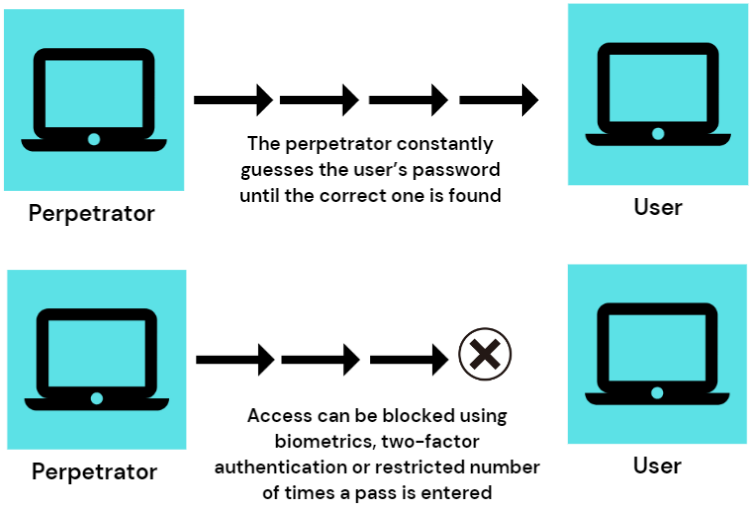
All blocks in a blockchain are interconnected via their unique identifiers. Each block contains the previous block's identifier. Once added, a block is completely unchangeable by anyone on the network, and can only be viewed.

Cyber Attacks

There are multiple ways malicious third parties can hack into our data and use it for their own purposes. The most common ones are listed below.

Brute-force attacks:

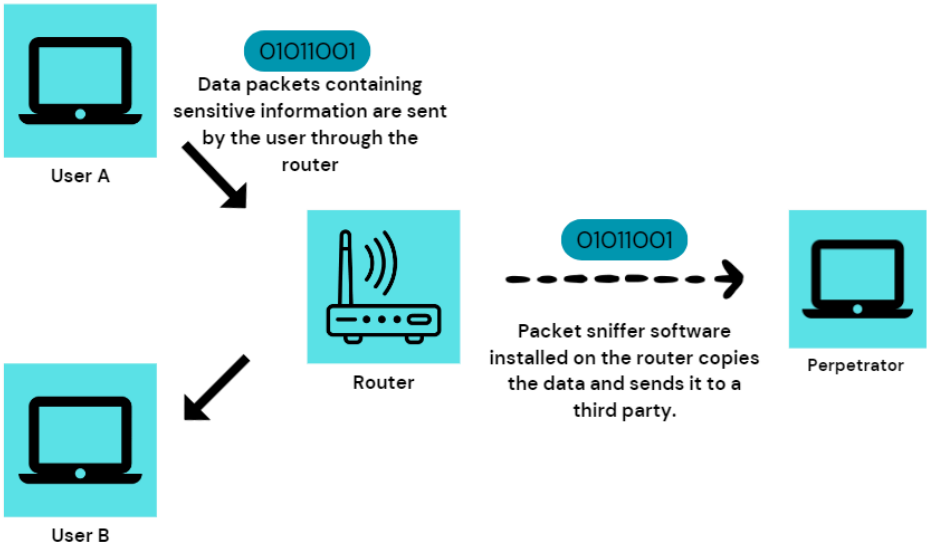
- The perpetrator tries multiple combinations of passwords repeatedly in order to guess it and hack into the user's account. The passwords can be guessed manually or by using software.
- Preventions:
    - Using biometrics such as facial recognition or fingerprints to unlock devices
    - Using two-factor authentication
    - Using passwords which are difficult to guess, such as ones containing symbols and numbers.



Data Interception

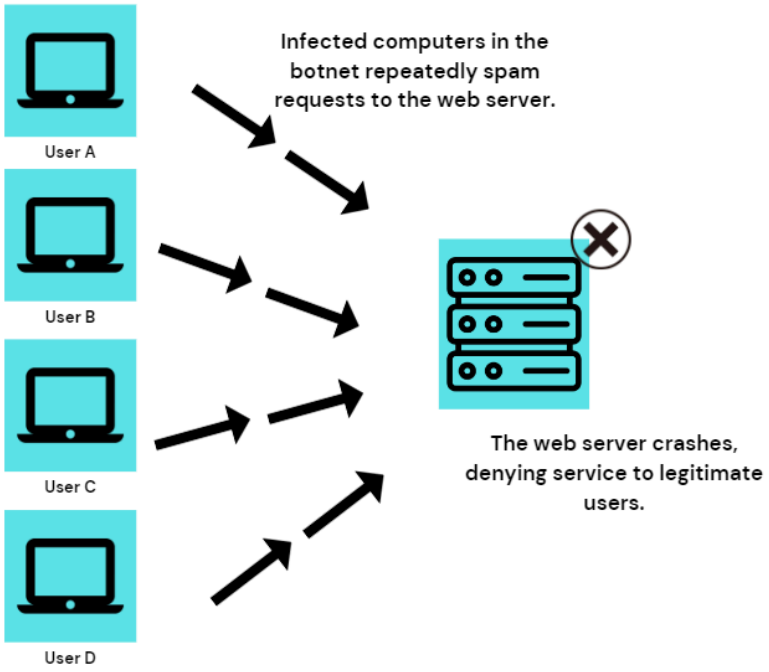
- Data interception is done by using a packet sniffer software which the perpetrator has previously installed on transmission hardware. For example, it can be installed on a router.

- The packet sniffer looks through all data packets being transmitted through that router or piece of hardware. If it finds any useful information, such as passwords, it copies and sends it to the perpetrator.
- Preventions:
    - Encrypting the data so it becomes meaningless to anyone who sees it.



Distributed Denial of Service (DDoS):

- DDoS is an attack towards web servers typically done as revenge or to demand ransom.
- The following steps take place:
    1. The perpetrator sends out malware publicly so that as many computers as possible download it by accident. Once downloaded, it turns the computers into bots. The collection of all computers is known as a botnet. Inactive computers are called zombies.
    2. To carry out a DDoS attack, the perpetrator makes the malware in the bots repeatedly send requests to a certain web server.
    3. The web server crashes due to overload of requests at once, denying service to legitimate users.
- Preventions:
    - Using a proxy server.
    - Using anti-malware software.



Hacking

- Hacking is the act of gaining unauthorised access to other's data. This can be done through vulnerabilities in networks, such as open ports or software bugs.
- Preventions:
    - Two factor authentication, biometrics, and strong passwords.
    - Regularly updating software to ensure it has no vulnerabilities
    - Using anti-malware softwares and regularly checking for malware
    - Using a firewall.

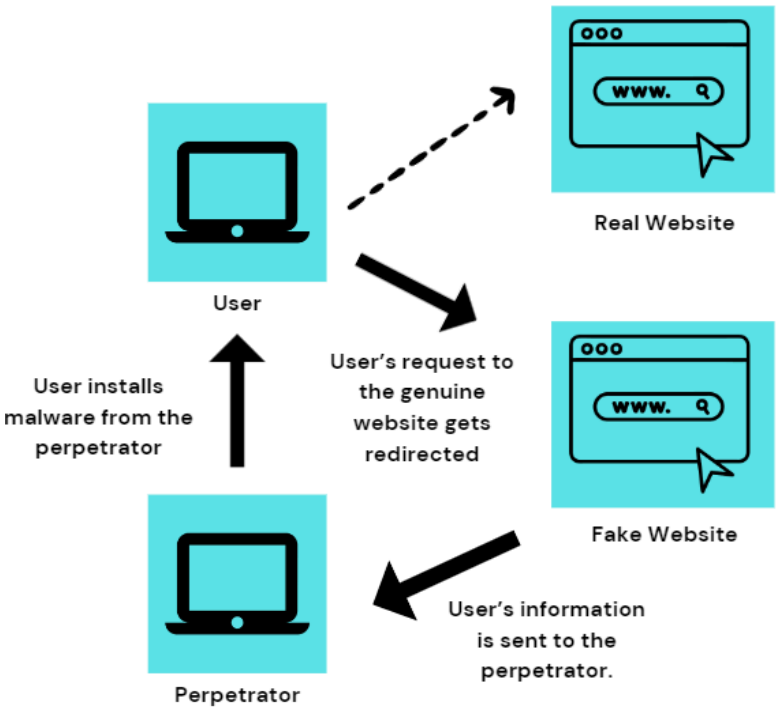
Malware

Name	Description	Preventions
Virus	It gets installed onto the secondary memory, where it proceeds to replicate itself until there is no more storage left, causing the PC to crash. It also may corrupt data.	Anti-malware software Firewalls Mindfulness while clicking unknown links Have data backup
Worm	Once installed, it finds network vulnerabilities and uses them to reduce bandwidth.	Anti-malware software Firewalls Mindfulness while clicking unknown links

Spyware	Once installed, it can record physical actions taken by the user, such as keyboard inputs. It sends this data to a third party, which can cause sensitive typed information to be leaked.	Anti-malware software Firewalls Using an on-screen keyboard as its clicks cannot be detected Using auto-filled passwords instead of typing them out each time
Adware	Spams ads and unwanted pop-ups on websites, applications or externally.	Anti-malware software Firewalls Mindfulness while clicking unknown links
Ransomware	Encrypts the user's data so that it is inaccessible, and demands money in exchange for the data to be unencrypted.	Anti-malware software Firewalls Have data backup
Trojan Horse	Malware that is disguised as a genuine application that may trick uses into downloading it.	Anti-malware software Firewalls Mindfulness while clicking unknown links

Pharming:

- A method used by perpetrators to gain access to sensitive information such as passwords.
- The steps taken are as follows:
    1. The user will be tricked into clicking a link which will start an automatic malware download in their secondary storage.
    2. The next time the user enters a certain URL, such as the link to a bank's site, the malware will redirect them to a fake website instead. The fake website will imitate the real one in order to trick the user into believing that it is real.
    3. The user will enter personal information into the website, which will be sent to the perpetrator.
- Preventions:
    - Refrain from clicking unknown links or download suspicious software
    - Ensure the website is genuine by checking its URL
    - Use anti-malware



Phishing:

- Phishing is a method similar to pharming used for obtaining sensitive information from users for perpetrators to use.
- The perpetrator will send an email to the user, which will entice them into clicking a fake website's link and entering personal information there. For example, an email could inform the user that their bank details need to be updated, and then prompt them to click on the fake website's link.
- Preventions:
    - Refrain from clicking unknown links or download suspicious software
    - Ensure the website is genuine by checking its URL

Social Engineering:

- Social engineering is the act of manipulating or deceiving people to obtain their personal information. For example, scam calls.
- Preventions:
    - Don't trust unknown people asking for sensitive information
    - Use access levels

Preventing Cyber Attacks

Firewalls:

- Typically used for a personal computer to prevent suspicious programs or data to enter or leave the computer.
- It checks incoming and outgoing signals
- It filters data packets to prevent traffic and blocks any packets if required
- It can whitelist or blacklist sites or data allowed to enter based on criteria set by the user
- It can be hardware or software

Proxy Servers:

- Typically used for protection of larger serves, to divert attacks on them.
- It checks, logs and has the ability to block incoming and outgoing signals.
- It can whitelist or blacklist certain IP addresses and signals
- It processes client-side requests primarily.
- It allows faster access to websites using cache.
- It can hide a server's internal network.

Access Levels:

- Typically used in companies
- It provides each user/employee with different permissions to data
- It limits access to data that can be viewed, and limits access to data that can be changed or deleted.
- It is usually linked to usernames or user IDs.

Anti-Malware:

- It is a software that scans the computer it is installed on for malware
- It keeps a record of known malware
- It removes/quarantines any suspicious software or malware
- It checks data before it is downloaded, and stops the download or warns the user if it detects malware.

Passwords:

- Strong passwords, which are changed often, are good defences against data breaches.

Biometrics:

- The required data for biometrics, such as fingerprints, is unique for everyone. So, it is very difficult to replicate.

Two-step Verification:

- In two step verification, extra data is sent to the user's device, which has been pre-set by the user. It is difficult to obtain.

- The data must be entered into one system, which is difficult to do from remote locations.

Automated and Emerging Technologies

Automated system:

- A device that is operated without human interaction

Features of an automated system:

- **Sensors:**
  - A type of input device that is used to capture data from its immediate environment
- **Microprocessors:**
  - An integrated circuit that is able to perform that is able to perform many of the functions of a CPU
- **Actuators:**
  - A mechanical part that causes another device or part to move.

Industry:

Advantages	Disadvantages
Faster than humans to take action	**Expensive to set up**
Safer	**Enhanced maintenance (expensive)**
Low running cost	**Computerised system; subject to cyberattacks**

Agriculture:

Advantages	Disadvantages
Reduced labour costs	**Expensive to set up**
Safer	**Enhanced maintenance (expensive)**
Better control of the irrigation process and of precious resources like water	

Transport:

Advantages	Disadvantages
Allows the same number of cars to use fewer parking spaces	**Requires additional maintenance to ensure it correctly functions**

Avoids traffic disruption in cities	
Cars can fit into smaller spaces	

Weather (stations):

Advantages	Disadvantages
Reduced labour costs	**Expensive to set up**
Safer	**Enhanced maintenance (expensive)**
Better control of the irrigation process and of precious resources like water	

Lighting:

Advantages	Disadvantages
Possible to control light sources automatically	**Expensive to set up**
A reduced energy consumption	**Enhanced maintenance (expensive)**
Longer bulb life	

Science:

Advantages	Disadvantages
Consistent and faster results	**Expensive to set up**
Safer	**Security risks**
Fewer highly trained staff needed for each experiment	**Less flexible than using human technicians**

Robotics:

- A branch of computer science that looks at the creation and use of robots.
- **Robots have a range of features:**
  - ★ A mechanical structure or framework. This is the body of the robot.
  - ★ Electrical components such as:
    - Sensors to record its environment, for example, the position of the component it is building.

- Microprocessors, to take the reading from the sensor and decide the action to perform.
- Actuators, to make the robot move.

★ They are programmable

★ They are used in factory equipment, domestic robots and drones

Robots in medicine

Advantages	Disadvantages
Saves time for doctors and nurses.	**Network failures can interrupt remote procedures, necessitating backup systems**
Enables remote surgeries, reducing patient wait times.	**Programming errors and security risks can cause harm.**
Allows resource sharing among hospitals.	

Robots in agriculture

Advantages	Disadvantages
Saves time	**High initial costs and ongoing maintenance**
Increases efficiency with 24/7 operation	**Potential job replacement for human workers**
Ensures precision through satellite-guided automation	

Robots in transport

Advantages	Disadvantages
Creates new jobs in system management	**High initial costs**
Enhances efficiency and security	**Security risks from potential tampering**
Reduces risks in hazardous environments	**Job displacement for some workers**

Robots in industry

Advantages	Disadvantages

Continuous operation 24/7	**High initial costs**
Reduce human error and enhance accuracy	**Potential job replacement for human workers**
Efficiency as automation speeds up production processes	**Requires new skill sets for fewer workers to manage and maintain robots.**

- **Robots can also handle specific tasks like:**
  - ★ Packing items into boxes
  - ★ Testing products (e.g., checking temperatures)
  - ★ Performing precise tasks (e.g., building circuits)

Robots in entertainment

- **There are different types of robots used for entertainment purposes:**
  - ★ Interactive toys
  - ★ Teaching kids on building robots and program them to perform specific tasks
  - ★ Help people learn about robotics and coding

Robots used for domestic purposes

- **There are different types of robots used for domestic purposes:**
  - ★ Vacuum cleaners and lawnmowers:
    - This saves people time
    - Still requires monitoring
    - Cannot go up and down steps

Artificial Intelligence:

- A branch of computer science that looks at creating machines that can stimulate human behaviour
- **There is a broad range of AI:**
  - ★ Image Recognition: Identifies objects or people in images.
  - ★ Speech Recognition: Recognizes and stores spoken words.
  - ★ Natural Language Processing: Understands and responds to unstructured commands.
  - ★ Computer Games: Moves elements or characters based on the environment.
  - ★ Diagnosis Systems: Assists in medical diagnosis.

Features of AI:

- Data collection: Programs gather data from users or sensors.
- A set of programmed rules: Rules are used for decision-making (e.g., turning when an object is detected).
- The ability to reason: Logic-based reasoning develops facts from rules (e.g., deducing that Fred eats meat because he is a dog).
- The ability to learn and adapt: Machine learning allows systems to improve by training on data (e.g., distinguishing between images of dogs and horses).

There are two areas of AI

- **Machine Learning:**
  - A computer program that can adapt its stored rules or processes.
  - It can be supervised or unsupervised
  - Supervised: the user tells the machine what the data means
  - Unsupervised: the data is input, then the data learns from the program and identifies which items of the data resemble one another
- **Expert system:**
  - A system that attempts to replicate the knowledge of an expert.
  - **It is made up of:**
    - A Knowledge base
    - A Rules base
    - An inference engine
    - User interface

Keywords

Knowledge base:
★ A list of facts stored in the expert system

Rules base:
★ It stores the rules upon the knowledge in the expert system

Inference engine:
★ Part of the expert system that makes decisions

User interface:
★ It outputs questions and statements to the user, and allows the user to input data

r/IGCSE Resources

r/IGCSE Resources repository | r/IGCSE subreddit | Our Discord server

igcse.reddit.com

discord.gg/igcse

Acknowledgments and Information:

The information on this guide has been generously prepared by current students/alumni; the authors have been acknowledged where possible. Links to external sites embedded in this document are not affiliated with or operated by r/IGCSE.

This guide is meant to be for educational purposes only and is to remain free of cost for the benefit of all students.

For any suggested changes or corrections to this document, please contact the r/IGCSE staff team via email or on the r/IGCSE Discord server.