Central Processing Units

Control Unit (CU)

- Fetches, decodes, and manages the execution of instructions
- Issues control signals to control hardware
- Moves data around the system

Arithmetic Logic Unit (ALU)

- Performs arithmetic and logical operations.
- Where calculations are done and where decisions are made.

Registers

- Small amounts of high speed memory in the CPU.
- Used to store small amounts of data that are needed during processing.

Clock

- Used to coordinate all the computer's components.
- Sends out a regular electrical pulse to do this.
- The frequency of the pulses = clock speed, measured in hertz.
- Higher clock speed = greater number of instructions which can be performed at a time.

Buses

- High speed internal connections.
- Used to send control signals and data between the processor and other components.
- Address bus - carries memory addresses from the CPU to other components.
- Data bus - carries data between the CPU and other components.
- Control bus - carries control signals from the CPU to other components.

Stored Program Concept.

The processing architecture that all CPUs use.

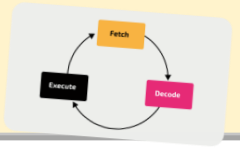
John von Neumann invented the processor architecture which stores a program in memory as instructions and executes them sequentially using the ALU, control unit and registers.

Fetch-Decode-Execute Cycle

The fetch-decode-execute cycle is followed by a **processor** to process an instruction.

The cycle consists of several steps:

1. The memory address held in the program counter is copied into the MAR.
2. The address in the program counter is then incremented - or increased - by one. The program counter now holds the address of the next instruction to be fetched.
3. The processor sends a signal containing the address of the instruction to be fetched along the address bus to the computer's memory.
4. The instruction held in that memory address is sent along the data bus to the MDR.
5. The instruction held in the MDR is copied into the CIR.
6. The instruction held in the CIR is decoded and then executed. The results of processing are stored in the ACC.
7. The cycle then returns to step one.



Depending on the type of instruction, additional steps may be taken:

If the instruction is to transfer data held in the ACC back to RAM, the intended memory address is copied into the MAR.

The data to be transferred is copied into the MDR and then transferred to the specified address using the address bus and data bus.

The executed instruction may require the program to jump to a different place in the program.

In this case, the memory address of the new next instruction to be fetched is copied into the program counter.

The process then restarts at step one.

Keywords

Control Unit

Arithmetic Logic Unit

Cache

Register

Bus

Memory

Applications

Instructions

Data

Kettlethorpe HIGH SCHOOL

Computer Science

Year 10| Hardware

Secondary Storage

Used to store programs and data for longer term when the computer is switched off. Contains non-volatile data which is retained with the computer is switched off. Not all computers require secondary storage. Embedded computers such as a watch do not need to store data when power is turned off.

Magnetic devices

- Use magnetic fields to magnetise individual sections of a spinning disk.
- Fairly cheap, high in capacity and durable.
- Can be damaged if dropped.
- Vulnerable to magnetic fields.

Optical Devices

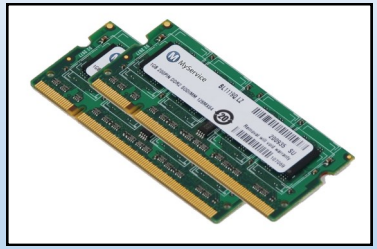
- Use a laser to scan the surface of a spinning disc.
- The disc surface is divided into tracks, with each track containing flats and hollows.
- The flat areas are known as lands and the hollows as pits.
- Lands reflects the laser light back; pits scatter the beam.

Solid State Devices

- Have faster access times than other devices
- Because they have no moving parts, are more durable.
- More expensive so tend to be smaller in capacity.
- Require little power, so used where battery life is a consideration.

RAM – Random Access Memory

- Volatile memory – data is lost when the computer is turned off.
- Called random access because data can be directly written to or read from any location.
- Used to hold data and instructions that are currently in use.
- The more RAM a computer has, the more data it can hold simultaneously



Cache

- A small amount of high speed memory in the CPU.
- Used to temporarily hold data the CPU will reuse.
- Allows for faster processing since as the CPU need not wait for data to be fetched from RAM.

Embedded Systems

A small computer which includes hardware and software, designed to control a specific device.

Forms a part of a larger device such as a washing machine and can perform only a limited number of tasks.

Have several advantages:

- Cheaper to design and build.
- Require less power.
- Do not need much processing power.

Kettlethorpe HIGH SCHOOL

Computer Science

Year 10| Hardware

Operating Systems

User Management

- Individual users can be created and deleted.
- Allows more than one person to use a computer with their own files and settings.
- Access levels control user access to systems for security.

File Management

- Allows users to find and manage data stored by the computer.
- Data is stored in files, within folders, within drives.
- Assigns metadata to files including date created, date modified, last date accessed.

Process management

- Allows users to run applications such as web browsers or word processors.
- Multiprogramming enables several programs to run at the same time.
- Each program is made up of instructions. When running, they are called a process.
- Allocates use of the main memory and the CPU between processes.
- A scheduler is used to time the different processes.

Peripheral Management

- Manages input and output between peripherals and a process.
- Data is transferred between input devices, the CPU, and output devices.
- Uses device drivers to communicate with devices.

Utility Software

Defragmentation

- Files on a disk are broken down into a series of segments.
- When files are deleted, the segments where they were stored are made available for new files.
- The new file may need more segments than the old, and so the segments allocated to it are not together on the disk. This is known as fragmentation.
- A fragmented disk takes longer to read from and write to, making the computer slower.
- Defragmentation software rearranges the segments so that they are stored next to each other.

Anti-Malware

- Protects against viruses, spyware, and other unwanted software.
- Scans the system to identifies potential viruses.
- Will attempt to delete or fix potential threats once they have been identified.
- Runs either when activated or automatically at a specified date and time.

File Repair

- Corrupt files can sometimes be repaired.
- Can detect and recover physical errors on the disk and mark damaged sections as unavailable.

Utility Software

Data Compression

- Reduces the size of a file using algorithms.
- Smaller files are easier to transmit.
- Allows more files to be stored in the same space.

Backups

- A copy of data is known as a backup.
- These allow damaged or deleted data to be restored.
- Full backups include every file. This requires a lot of storage and time.
- Incremental backups include new and changed files since the last backup.

Developing Robust Software

- What threats will the code face?
- Are security features like usernames and passwords needed?
- How will patches be installed and the code updated?
- Is encryption needed?
- Does the coder need to create an audit trail?

Audit Trail - *a record of what has been done and who or what did it.*
Code Review – *a check of code by other programmers.*

Compilers v Interpreters

Compilers	Interpreters
• Translates the whole code in one go into Machine Code. • Optimise the code • Used at the end of development when code is finished • Create error reports and object code. **Compiled programs run quickly and without needing additional software. Programs are supplied as executables which cannot be modified.** **Because the source code is translated as a whole, more memory is needed.**	• Translate and execute source code • Work line by line. • Syntax is checked • If code is correct it is executed • If code is incorrect interpreting is • stopped. **Instructions are executed as soon as they are translated. Instructions are not stored for later so less memory is needed.** **The CPU must wait for each instruction to be translated so execution is slower.**

High Level Languages

Easier for humans to understand, using English like words and phrases.

- Much easier to learn, write and debug.
- Examples include Python, Java and C

Low Level Languages

Very close to computer language, hard for humans to understand.

Machine code

- CPUs understands machine code can directly execute it.
- Consists of 0s and 1s only.
- Very difficult to learn, write and debug.

Assembly Language

- Also known as Assembly Code
- Uses mnemonics (abbreviations)
- Easier for humans to understand and program but
- still difficult
- Must be translated into Machine Code for execution
- Commonly used to program device drivers

Computational Thinking

Decomposition

Breaking down large problems into a set of smaller parts.

- Smaller problems are easier to solve.
- Each part can be analysed independently.
- The parts are combined to produce the full problem.

Abstraction

Using symbols and variables to represent a real-world problem using a computer program and removing unnecessary elements.

- Allows the creation of a general idea of how to solve the problem.
- Provides focus on what actually needs to be done.
- Provides a simple view of the problem.

Sequencing

Breaking down complex tasks into simple steps.

- Each line follows the next.
- Can create simple programs very quickly.
- Easy to follow for a small program.
- Not very efficient.
- Difficult to follow with large programs.
- Hard to maintain.

Selection

- Allows the program to make decisions
- Uses conditions to change the flow of the program
- IF Statements may be nested one inside another
- IF statements perform comparisons sequentially and so the order is important.

```
num = 30
if num > 10:
    print("True")
else:
    print("False")
```

Repetition

Repeating a set of steps several times.

Count Controlled:

- Repeats the same code a set number of times
- Uses a variable to track how many times the code has been run
- This variable can be used in the loop
- At the end of each iteration the variable is checked to see if the code should be run again
- FOR sets how many times the code should be repeated
- NEXT tells the code to return to the start of the loop
- STEP sets how the variable should increment

```
for i in range(3,30):
    print("Iteration:", i)
```

Condition Controlled:

- Uses a condition to determine how many times code should be repeated
- While loops will run whilst a condition is met and use the statements WHILE and ENDWHILE
- Repeat loops will run until a condition is met and use the statements REPEAT and UNTIL

```
x = 1
while x <= 5:
    print (x * 5)
    x = x + 1
print("done")
```

Sub Programs

Small programs which form part of a larger program.

Procedures are sets of instructions stored under a single name (identifier).

Functions are like procedures but will always return a value to the main program.

Parameters are values passed into a sub program. These are referred to as arguments when calling the sub program.

Kettlethorpe HIGH SCHOOL

Computer Science

Year 10| Computational Thinking

Sorting Algorithms

Bubble Sort

1) Take the first element and second element
2) Compare the two
    a) If element 1 > element 2
        i) Swap them over
    b) Otherwise
        i) Do nothing
    c) Move to the next pair in the list
    d) If there are no more elements return to step (1)
    e) Otherwise, return to step (2)
3) Repeat until you have worked through the whole list without making any changes

Merge Sort

1) Split the list into individual elements.
2) Merge the elements together in pairs, putting the smallest element
first.
3) Merge two pairs together, putting the smallest first.
4) Keep merging until all pairs are in order.

Searching Algorithms

Linear Search

1) Check the first value
2) If it is desired value
    a) Stop
3) Otherwise check the second value
4) Keep going until all elements have been checked or the value is found

Binary Search

1) Put the list in order.
2) Take the middle value.
3) Compare it to the desired value.
    a) If it is the desired value.
        i) Stop.
    b) If it is larger than the desired value.
        i) Take the list to the left of the middle value.
    c) If it is smaller than the desired value.
        i) Take the list to the right of the middle value.
4) Repeat step 3 with the new list.

Trace Tables

A method of recording the values used within an algorithm at each stage of processing to help in troubleshooting.

- Tests algorithms for logic errors which occur when the algorithm is executed.
- Simulates the steps of algorithm.
- Each stage is executed individually allowing inputs, outputs, variables, and processes to be checked for the correct value at each stage.
- A great way to spot errors

Flowcharts

- Created to represent an algorithm.
- Show the data that is input, and output.
- Show processes that take place.
- Show any decisions and repetitions that take place.
- Lines show flow through the chart.
- Shapes represent different functions

Symbol	Name	Function
	Start/end	An oval represents a start or end point
	Arrows	A line is a connector that shows relationships between the representative shapes
	Input/Output	A parallelogram represents input or output
	Process	A rectangle represents a process
	Decision	A diamond indicates a decision

Kettlethorpe HIGH SCHOOL

Computer Science

Year 10| Computational Thinking

Arithmetic Operators

- Addition +
- Subtraction -
- Multiplication *
- Division /
- MOD Modulus (the remainder, e.g. 12 MOD 5 gives 2)
- DIV Quotient (integer division, e.g. 21 DIV 5 gives 4)
- Exponentiation (to the power of, e.g. 3^3 gives 27)

Comparison Operators

- == Equal to
- != Not equal to
- < Less than
- <= Less than or equal to
- > Greater than
- >= Greater than or equal to

Boolean Operators

- AND - two conditions must be met for the statement to be true
- OR - at least one condition must be met for the statement to be true
- NOT – inverts the result, e.g. NOT(A AND B) will only be false when both A and B are true

Evaluating Fitness for Purpose and Efficiency

Fit for Purpose - meets the original purpose and requirements the code was designed for. Provides the expected outputs. Test tables help to examine the values at each stage and check code is working as expected.

Efficient – the amount of time and resources needed to run a particular program. Steps which improve efficiency.

- Using repetition (loops) to reduce the amount of code
- Using arrays instead of declaring many individual variables
- Using selection statements which only make comparisons until a solution is reached

Variables

A place in memory in which data may be stored.

- Different types e.g. string, decimal, etc.
- Allows the program to store data such as an input for later use

Constants

A fixed value used by the program such as pi

Allows easy use of fixed values without having to store them in the program.

Arrays

- An ordered collection of related data.
- Each element in the array has a unique index, usually starting at 0.
- All elements must be the same type of data.
- Arrays are usually a fixed size.

```
names = ['jack', 'mary', 'robert', 'owen', 'maggie', 'timothy']
#           0       1       2        3        4         5

print(names[1])
#prints the index 1 - which is mary
```

- 1 Dimensional arrays are like a simple list, each element needs a single index number. **names[1] references element 1 in the 1D names array.**
- 2 Dimensional arrays are like tables, with each element needing two index numbers.
- 2 Dimensional arrays are usually used to store properties of objects, with objects in rows and properties in columns. **name[0,2] references element 0,2 in the names array.**

Kettlethorpe HIGH SCHOOL

Computer Science

Year 10| Python Programming

Binary

A number system made up of 0 and 1 used by computers to store and represent data such as numbers, sound, and graphics.

- Also known as Base 2.
- Computers use binary because the CPU contains transistors, which are either on or /off.

Binary States

The number of states which can be represented in a given number of bits using binary.

- To calculate the number of states use the formula 2 n where n is the length of the pattern
- Limits on the number of bits available affect how much data can be stored

Binary Maths

Addition

- 0 + 0 = 0
- 1 + 0 = 1
- 1 + 1 = 10 (binary for denary 2)
- 1 + 1 + 1 = 11 (binary for denary 3)

Multiplication (using binary shifts)

- Move the digits to the left and fill the gaps after the shift with 0.
- Move 1 place for X2, 2 places for X4 etc.

Division (using binary shifts)

- Move the digits to the right and fill the gaps after the shift with 0.
- Move 1 place for X2, 2 places for X4 etc.

Hexadecimal

A number system made up of 16 symbols, 0-9 and the letters A-F.

- Also known as Base 16
- Useful because large numbers can be represented using fewer digits.
- Easier to understand, write and check than binary.
- Commonly used for colour values and MAC addresses.

Measuring Storage	
Size	**Binary Unit**
8 bits (b)	**1 byte**
1024 bytes (B)	**1 kibibyte**
1024 kibibytes (KiB)	**1 mebibyte**
1024 mebibytes (MiB)	**1 gibibyte**
1024 gibibytes (GiB)	**1 tebibyte**
1024 tebibytes (TiB)	**1 pebibyte**

Unsigned and Two's Compliment Integers

Unsigned integers must be positive. Signed integers can be positive or negative.

- Both are as accurate
- Both can have overflow errors
- Unsigned integers store more positive values (unsigned is 0 to 255, signed is -127 to 127)

Two's complement

- Used by computers to show negative numbers.
- More effective when performing mathematical operations.
- The bit at the far left of the bit pattern - the most significant bit or MSB - is used to indicate positive or negative and the remaining bits are used to store the size of the number.

Using two's complement for negative numbers

- Find the positive binary value for the negative number.
- Add a 0 to the front of the number, to indicate that it is positive.
- Invert or find the complement of each bit in the number.
- Add 1 to this number.

Kettlethorpe HIGH SCHOOL

Computer Science

Year 10| Data

Converting Between Bases

Binary to Denary

1) Draw your conversion table.

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

2) Write the binary number in the conversion table.

3) Add together all numbers with a 1 beneath them

128
64
\+ 8
4
204

11001100 in binary is 204 in denary

Denary to Binary

1) Draw your conversion table.
2) Is the number higher than the first column in the table?
   a) If so, put a 1 in that column and work out the difference.
   b) If not, put a 0 in that column.
3) Repeat the step above with the difference.
4) Keep going until the difference is 0, put a 0 in any empty columns.
5) Read the number from the bottom row of the table.

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

200 – 128 = **72**

72 – 64 = **8**

8 – 8 = **0**

200 in denary is 11001000 in binary

Denary to Hex

1) Divide the denary number by 16 and write down both the answer and the remainder.
2) Divide the answer by 16 again. Write down both the answer and the remainder.
3) Keep going until you reach an answer of 0.
4) Read the remainders from bottom to top.
5) Convert each remainder to hex.

62 ÷ 16 = 3 R 14

3 ÷ 16 = 0 R 3

3 14

3E

62 in binary is 3E in hexadecimal

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

Binary to Hex

1) Draw two separate conversion tables.
2) Write the binary number across both tables.
3) For each table, add up the numbers which have a 1 beneath them.
4) Convert each number to hexadecimal.

8	4	2	1
0	1	1	0

8	4	2	1
1	1	0	1

4 + 2 = 6

8 + 4 +1 = 13

6

D

01101101 in binary is 6D in hexadecimal

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

Compression

Lossy Compression

- Some data is removed, reducing the amount of data and the size of the file.
- There is a balance to be struck between reducing file size and maintaining quality.
- Images can be compressed by reducing colour depth.
- Audio files can be compressed by reducing the bit depth of the samples.
- Lossy standards include:
  - JPEG file format for images
  - MPEG file format for video and audio
  - MP3 file format for audio

Lossless compression

- Files are reduced in size without the loss of data.
- Lossless compression results in a smaller reduction in file size than lossy compression.
- Lossless compression gives a better file quality that lossy compression
- Lossless standards include:
  - PDF file format for documents
  - GIF file format for image

ASCII – American Standard Code for Information Interchange

- A 7 bit method to represent characters using binary
- 7 bits allows 128 characters.
- The characters are represented in a table, called the ASCII table.
- ASCII characters include control codes, punctuation, upper and lower case letters, numbers.
- Upper and lower case characters have different codes.

Kettlethorpe HIGH SCHOOL

Computer Science

Year 10| Data

Advantages and Disadvantages of Networks

Advantages	Disadvantages
Software and files can be shared. Hardware such as printers can be shared Users can communicate via email, chat, etc. Centralised maintenance and updates. Centralised security. User monitoring. Different users can be given different access rights or permissions.	Cost, additional equipment is needed. Additional management by specialist staff. Spread of malware. Potential for hacking.

Wired v Wireless Networks

Wired Networks	Wireless Networks
Using fibre or copper cable to connect devices in the network together. Fibre cable provides a faster connection and can cover longer distances. **Advantages:** • Faster data transfer • Less likely to suffer from interference • More difficult for unauthorised users to intercept data **Disadvantages:** • Expensive to install or reconfigure • Harder to move devices so less flexible	Using radio signals or infrared light to connect devices in a network together. **Advantages** • Devices can easily be added • Users can move around freely and stay connected **Disadvantages:** • Signals have a limited range. • Can suffer from electromagnetic interference from other devices. • Signals can also be blocked by walls or other objects. • Each wireless access point (WAP) only has so much bandwidth.

Types of Network

LAN - Local Area Network

Confined to a single location.

Owned and maintained by a single organisation.

Used by organisation such as schools and small businesses.

Connected by cables or wireless.

WAN – Wide Area Network

Covers a wide geographical area.

Used by organisations with several different sites such as banks or universities.

Allows all the sites to communicate and share data.

Uses national or international long distance media.

The Internet

A vast WAN covering the entire world.

An Internet Service Provider (ISP) provides access to the Internet.

Routers provide an interface between the Internet and the customer via the ISP.

Network Speeds

Measured in bits per second.

- 1 Kbps = 1,000 bits per second
- 1 Mbps = 1,000,000 bits per second
- 1 Gbps = 1,000,000,000 bits per second

Working out file transmission speeds

time = size of file (in bits) / network speed (in bits)

Kettlethorpe HIGH SCHOOL

Computer Science

Year 10| Networks

Network Protocols

- **Ethernet** - used in wired LANs, covers many standards such as cable types and data transmission speeds.
- **Wi-Fi** - used in wireless LANs.
- **TCP/IP** - Transmission Control Protocol/Internet Protocol. Allows data to be appropriately addressed when transmitting and ensures the integrity of data.
- **HTTP** – Hypertext Transfer Protocol – Web pages
- **HTTPS** - Hypertext Transfer Protocol (Secure) – Secure web pages
- **FTP** – File Transfer Protocol - transmission of files across a network and the internet.
- **SMTP** – Simple Mail Transfer Protocol – Send emails
- **IMAP** – Internet Message Access Protocol –Receive emails
- **POP3** – Post Office Protocol version 3 –Receive emails

The Four Layer TCP/IP Model

- Breaks up the process for sending of messages into separate components.
- Each component handles a different part of the communication. Helps to understand the transmission process.
- Provides a basis to begin troubleshooting when something goes wrong.
- Application Layer – encodes and decodes message using protocols like HTTP or FTP.
- Transport layer - breaks down message into pieces called packets. Packets have a packet number. The recipient uses the number to reassemble the packets in the correct order and to see if there are any missing packets.
- Network layer - adds the sender and recipient IP address and transmits the message.
- Data link layer – provides physical transfer of packets over the network.

Bus Network

All devices are connected to a single cable (called the bus) with a terminator is at each end of the cable.

Advantages:

Easy to install extra devices.

Cheap to install as it doesn't require much cable.

Disadvantages

If the cable fails or is damaged the whole network will fail.

Performance becomes slower as additional devices are connected due to data collisions.

Star Network

All nodes are connected to one or more central switches. Often used with wireless networks.

Advantages:

Every device has its own connection so failure of one node will not affect others.

New devices can be added by simply connecting them to the switch.

Usually have higher performance as a message is passed only to its intended recipient.

Disadvantages

If the switch fails it takes out the whole network.

Mesh Network

No central connection point with each device connecting directly to others. Full mesh networks have every device connected to every other device. Partial mesh networks have each device connected to several others but not necessarily every other device.

Advantages:

Messages can be received more quickly.

Messages have many possible routes they can take.

Multiple connections mean that no device should be isolated

Each device can talk to more than one node at the same time.

Devices can be added without interruption.

Disadvantages:

Can be impractical and expensive to setup.

Require a lot of maintenance

Kettlethorpe HIGH SCHOOL

Computer Science

Year 10| Networks

Types of Error

A program with a syntax error will not run. A program with a logic error will run but it will not perform as expected.

Syntax Errors

When the code does not follow the syntax rules of the programming language used. This stops the program from running.

Examples:

- Misspellings or typos
- Using a variable before it has been declared
- Missing or incorrect use of brackets

Runtime Errors

Takes place during the running of a program causing it to crash.

- Trying to divide by zero
- Trying to access item 6 in an array of 5 items

Logic Errors

The program runs but does not do what it should.

Examples:

- Incorrectly using logical or Boolean operators
- Creating infinite loops
- Incorrectly using brackets in calculations
- Using the same variable name at different points for different purposes

Network Security

Access Control – *determines which files, software and systems users have access to.*

- Users should be restricted to access only the facilities they need for their jobs.
- Restrictions limit the actions a user can take, reducing the potential of threats.

Firewall – *a tool which monitors traffic going into and out of the network, and either allows or blocks it.*

- This decision is based on rules, known as the firewall policy.
- Can be hardware based or software based.
- Hardware firewalls are expensive, but more effective and powerful.

Physical Security - *restricting the physical access to important systems and parts of the network.*

- Important equipment such as servers should be kept in a locked secure room.
- Access should only be available to authorised people.
- Someone could remove or access the hard disks containing private information or damage equipment.

Identifying Network Vulnerabilities

It is important to identify and fix vulnerabilities before they can be taken advantage of by hackers.

Penetration Testing – *determines how resilient a network is against an attack.*

- Authorised users, sometimes an external company will probe the network for potential weaknesses and attempt to exploit them.
- Often carried out using specialist, automated software.

Ethical hacking - *attempt to access a network in the same way as a hacker.*

- They are looking for weaknesses in the security of the network.
- Weaknesses can then be fixed.
- Might be employed by the business that owns the network being tested or they might work for a security company.