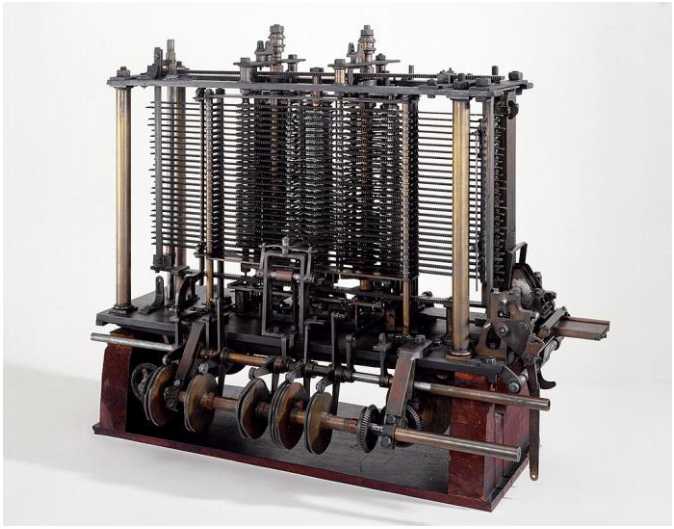
Computer Systems - Knowledge Organiser

Key Terms & Definitions

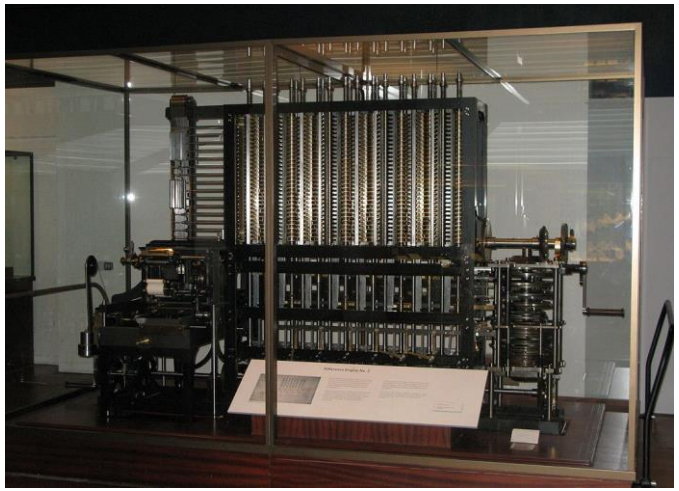
1	Computer	An electromechanical device which receives input, processes it and produces and output
2	Device	A piece of electrical or mechanical equipment made for a particular purpose
3	Program	A sequence of instructions written in a programming language that a computer can execute or interpret
4	Software	A set of programs used to operate computers and perform specific tasks
5	Hardware	The physical components of a computer
6	Data	Individual facts or statistics
7	Processor	The part of the computer that interprets and carries out instructions
8	Main memory	The part of the computer that stores data that is currently being used by the processor
9	Secondary storage	The part of the computer that stores data long term that is not currently being used by the processor
10	I/O (Input / Output)	Refers to input, any method of getting information into the computer, and output, any method of getting data out of the computer.
11	Computer architecture	The way in which the parts (components) of a computer system are organised
12	Operating system	Specialised software that communicates with computer hardware to allow other programs to run
13	Logical operator	The name of a logic circuit (AND, OR, NOT)
14	Logical expression	A text based method of describing a logic circuit
15	Truth table	A way of describing the output of a logic circuit for all possible inputs
16	Logic gate	A physical device which performs a logical operation
17	Logic circuit	Two or more logic gates connected together to solve a problem or perform a task

18	Artificial intelligence (AI)	Any machine that performs tasks that typically require intelligence in humans (suggestion - there's no agreed definition)
19	Machine learning	A type of AI in which a range of techniques are used to attempt to imitate the way that humans learn
20	Free software	Software which the user has the right to: ● use for any purpose ● study how the software works and change it however they want ● redistribute and make copies ● improve it and share their improvements with anyone
21	Open source software	Mostly the same software in practical terms as free software however the different terms exist because the people advocating the use of each have different views about what free / open source software.

Types of Computer

<table>
<tr>
<td> 

[1]
Babbage's difference engine was the first design for a general purpose computer, one that can automate any process specified by a program.</td>
<td> 

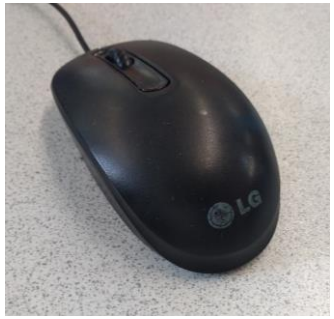
Before mechanical and electronic computers, a computer was a person who was employed to do tasks (such as performing complex calculations) that we now use modern computers to do.</td>
<td> 
[2]
Babbage also designed the difference engine. The major difference between this and the analytical engine was that the difference engine was not designed to be general purpose. It was specifically designed to do certain calculations and can therefore be considered a special purpose computer.</td>
</tr>
<tr>
<td colspan="2">General purpose computers</td>
<td>Special purpose computers</td>
</tr>
</table>

[1] Portion of the calculating machine with a printing mechanism of the Analytical Engine, built by Charles Babbage, as displayed at the Science Museum (London) by Mrjohncummings is licensed under CC BY-SA 2.0

[2] Babbage Difference Engine by geni is licensed under CC BY-SA 4.0

Hardware and Software

Hardware



Mouse - External hardware

Motherboard - Internal Hardware

Hardware can be **internal** (inside the PC/laptop/mobile phone case) or **external** (outside the case).

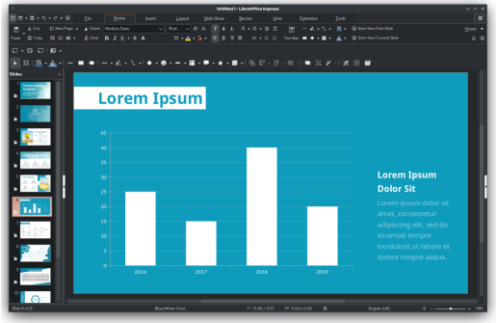
External hardware examples:

- Mouse
- Keyboard
- Monitor
- Headphones
- Speakers
- Webcam

Internal hardware examples:

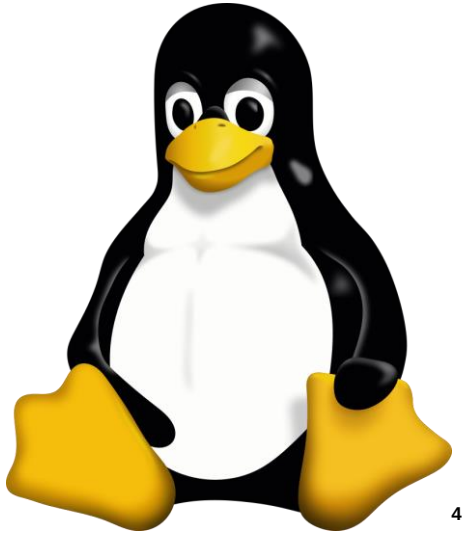
- CPU (Central processing unit) / Processor
- Motherboard
- GPU (Graphics processing unit)
- Hard drive
- RAM (Random access memory)
- Power supply (can be external in some devices)

Software



[3]

Presentation software



[4]

Tux, the mascot of the Linux operating system

Software can be placed into two categories: **system software** and **application software** based on the task(s) it performs.

Application software is designed to perform tasks that the user wants to complete. Examples include:

- Word processors
- Spreadsheet software
- Presentation software
- Web browsers
- Games

System software is designed to control the hardware of the computer. It provides an interface between the hardware and the application software.

The computer's operating system is an example of system software. It performs tasks such as memory management and processor scheduling which are necessary for the application software to run. Examples of operating systems include:

- Windows
- macOS
- iOS
- Android
- Linux

---

[3] LibreOffice 7.2.4.1 Impress screenshot by Risyad Rais is licensed under CC BY-SA 4.0

[4] lewing@isc.tamu.edu Larry Ewing and The GIMP

More on Hardware

Required hardware

A computer with required internal hardware

Modern computers come in a range of shapes and sizes for different purposes but all have the following hardware in some form:

- Processor
- Main memory
- Storage
- Communication system(s)
- I/O (Input / Output)

(Secondary) Storage

An SSD

Common examples of secondary storage devices include:

- Hard disk drives (HDDs)
- Solid state drives (SSDs)
- USB flash drives
- SD cards
- Optical disks

Architecture

John von Neumann[5]

Since all modern computers have similar basic hardware, we can say that they're similar in architecture.

Modern computers generally follow an architecture known as von Neumann architecture which was described by John von Neumann in 1945.

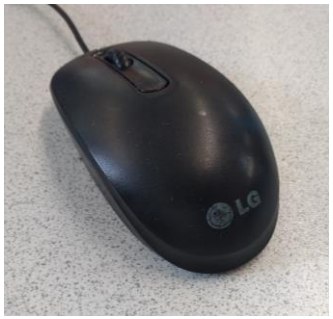
Main memory (RAM)

A 4GB stick of RAM

When a program is to be executed, it is loaded from secondary storage into main memory along with any data required. Instructions and data are then fetched from memory by the processor as required during the execution of the program.

Any new data created as a result of running the program must be saved from main memory to secondary storage otherwise it will be lost when the computer is switched off.

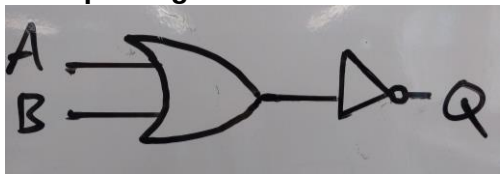
[5] JohnvonNeumann-LosAlamos by LANL

Processor (CPU)  **Two CPUs side by side**	When executing (running) a program, the CPU fetches instructions and data from main memory as required. It then decodes each instruction to understand what it is asking the CPU to do. It then performs the task that the instruction is asking it to do. Examples of operations the CPU may be instructed to do: ● I/O operations ● Arithmetic or logical operations on data ● Control the flow of a program (which instruction is to be executed and when)	**Communication components**  **A wireless NIC (network interface card)**	Used for transferring programs and data between computer systems. You will have used at least one wireless communication component such as WiFi, Bluetooth or mobile data (3G, 4G, 5G). Communication components can allow: ● Input to be obtained remotely ● Data to be stored on remote systems ● Programs to be executed remotely
Input devices  **Mouse - an input device**	Captures data from an external source. This could be someone typing on a keyboard, clicking with a mouse or even using a games controller. Examples of input devices include: ● Keyboard ● Mouse ● Games controller ● Camera ● Microphone ● Touchscreen	**Output devices**  **Speaker - an output device**	Communicates data stored on the computer to the user in some way. This could be a monitor displaying an image, a speaker playing a song or even a games controller with haptic feedback. Examples of output devices include: ● Speaker ● Headphone ● Monitor ● Touchscreen ● Games controller with haptic feedback

Boolean logic

Logical operations

Example Logic Circuit



OR gate followed by NOT gate to make a NOR gate

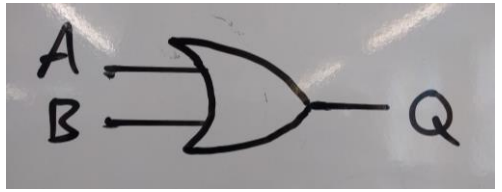
The fundamental logical operations are:

- not (inversion)
- and (conjunction)
- or (disjunction)

Hardware components are built from **logic gates** which have been connected together into **logic circuits**.

OR

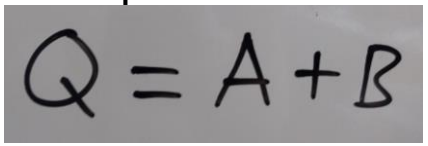
Gate



Truth table

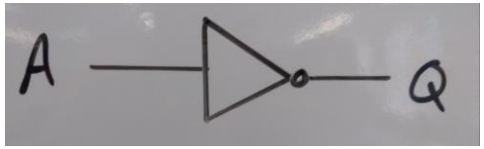
A	B	Q
0	0	0
1	0	1
0	1	1
1	1	1

Boolean expression



NOT

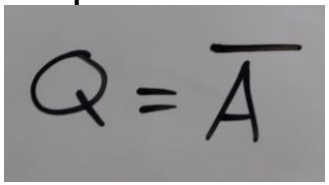
Gate



Truth table

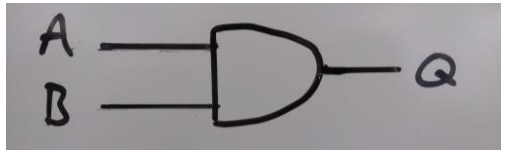
Input (A)	Output (Q)
0	1
1	0

Boolean expression



AND

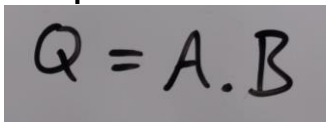
Gate



Truth table

A	B	Q
0	0	0
1	0	0
0	1	0
1	1	1

Boolean expression



AI and Machine learning

Artificial intelligence (AI) today

 [6]

A picture "painted" by an AI

AI today mostly focuses on individual aspects of intelligent behaviour.

Examples include:

- Game playing (Chess, Go)
- Deep Fakes
- Image recognition
- Facial recognition
- Natural language processing
- Targeted advertising
- Spam filtering

Machine learning (ML)

 [7]

Tesla engineers have made use of machine learning in the development of their self driving cars

There are several different approaches that can be taken to machine learning.

One is "supervised learning". This is where the machine is provided with examples to learn from.

Another approach is "reinforcement learning". This is where the machine is provided with feedback on its attempts to perform a task and it then uses this feedback to improve.

Case Study: AlphaGo

[8]

The AlphaGo program playing Go against a human opponent

In 2017, a computer program called AlphaGo which was created by engineers from Google successfully beat the top ranked Go player in the world at the time.

This is considered to be a major step forward in the capabilities of AI since Go is much more complex than other games that computer programs had beaten humans at (such as Chess) before this point.

Implications

It is important to consider how new technologies such as those that use AI and ML techniques will impact society.

In the case of self-driving cars, who is responsible for an accident?
In the case of medical applications, can the decisions of an AI doctor be explained to the patient?
How can we guarantee that AI and ML technologies don't lead to descrimintation? Will the benefits of AI be fairly distributed? How will humans handle less demand for labour?

---

[6] By Lasemainecomtoise (wikipedia name), in real life the artist Joseph Ayerle - Own work, CC BY-SA 4.0, https://commons.wikimedia.org/w/index.php?curid=94186985

[7] Tesla_Autopilot_Engaged_in_Model_X by Ian Maddox is licensed under CC BY-SA 4.0

[8] 8:36pm Match 3 of AlphaGo vs Lee Sedol by Buster Benson is licensed under CC BY-SA 2.0

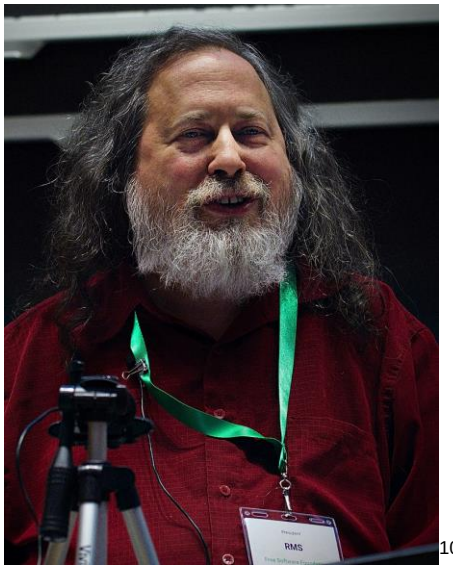
		These questions are being debated today. They have no simple answers.

Free and open source software

Free software  9 **A mobile phone running free software**	Free software is any software where the developer has granted the user the following four freedoms: • to use for any purpose • to study how the software works and change it however they want • to redistribute and make copies • to improve it and share their improvements with anyone	**Open source software**  **The logo of the open source initiative**	Open source has a more complex definition than just “having access to the source code”. It is fully defined by the Open Source initiative here.

Case Study: Richard Stallman	Stallman is commonly considered to be one of the leading figures in the free software movement and still holds influence today. He has dedicated a large proportion of his life to advancing the cause of free software with work such as the GNU project. He founded the Free Software Foundation to employ free software programmers and provide legal infrastructure for the movement.
	
	Despite his major contributions to free software, Stallman remains a controversial figure due to public comments he has made and there is disagreement as to whether he should be allowed to continue to be involved in the free software
		Examples
		`def add5(x):`
		`  return x+5`
		
		`def dotwrite(ast):`
		`  nodename = getNodename()`
		`  label=symbol.sym_name.get(ir`
		`  print '  %s [label="%s' %`
		`  if isinstance(ast[1], str):`
		`  if ast[1].strip():`
		`  print '= %s"];' % ast[`
		`  else:`
		`  print '"]'`
		`  else:`
		`  print '"];'`
		`  children = []`
		`  for  in n, childenumerat`
		`  children.append(dotwr`
		`  print ,'  %s -> {' % n`
		`  for  in :namechildren`
		`  print '%s' % name,`
		Python is FOSS software
			Common examples of FOSS (free and open source software) include:
			• Firefox Web Browser
			• Scratch Programming
			• Python Programming
			• Android OS
			• Linux OS
			• LibreOffice

[9] Replicant 4.0 on NexusS by GNUtoo is licensed under CC BY-SA 3.0

 [10]	foundation. Find out more about Stallman here.		

[10] Richard Stallman at LibrePlanet 2019 by Ruben Rodriguez is licensed under CC BY 4.0