Learning aims

- Describe GPUs and their uses
- Describe multicore and parallel systems

Key Terms

Multi-core & parallel systems
Co-processor
Workload
Supercomputers
Graphics processing unit (GPU)
Concurrent
Processing Speed:
Data
Performance
Core

Throughput - how much data a system can process in a given time.

Latency is the delay before a computer starts processing a task. CPUs aim to have low latency to respond quickly to commands.

Key term	Definition
Multi-core & parallel systems	CPUs with more than one core that can work on tasks at the same time (parallel processing), improving performance for multitasking and large workloads.
Co-processor	An extra processor designed to carry out specific tasks alongside the main CPU (e.g., GPU for graphics, FPU for maths).
Workload	The total amount of processing tasks a computer system has to handle.
Supercomputers	Extremely powerful computers with thousands of processors, designed to perform massive calculations quickly (e.g., climate modelling, scientific simulations).
Graphics Processing Unit (GPU)	A specialised processor with many cores designed for handling graphics and parallel tasks like AI or scientific computing.
Concurrent	When multiple tasks are in progress at the same time (not necessarily finished at the same time).
Processing speed	How quickly a computer can complete instructions, often measured in GHz (clock speed) or in instructions per second.
Data	The raw values (numbers, text, images) that the computer processes.
Performance	How efficiently a computer system completes tasks, measured using factors like speed, throughput, and responsiveness.
Core	An individual processing unit inside a CPU. Each core can fetch, decode, and execute instructions independently.
Throughput	The amount of data a system can process in a given time (higher throughput = more work done).
Latency	The delay before a system starts processing a task. Low latency means faster response times.

Co-processor systems

- A **co-processor** is an **additional processor** used for **specialised tasks**.

- In early computers, a single **CPU** handled all operations.

- As demands grew, co-processors were developed to work **alongside the CPU** to **boost performance**.

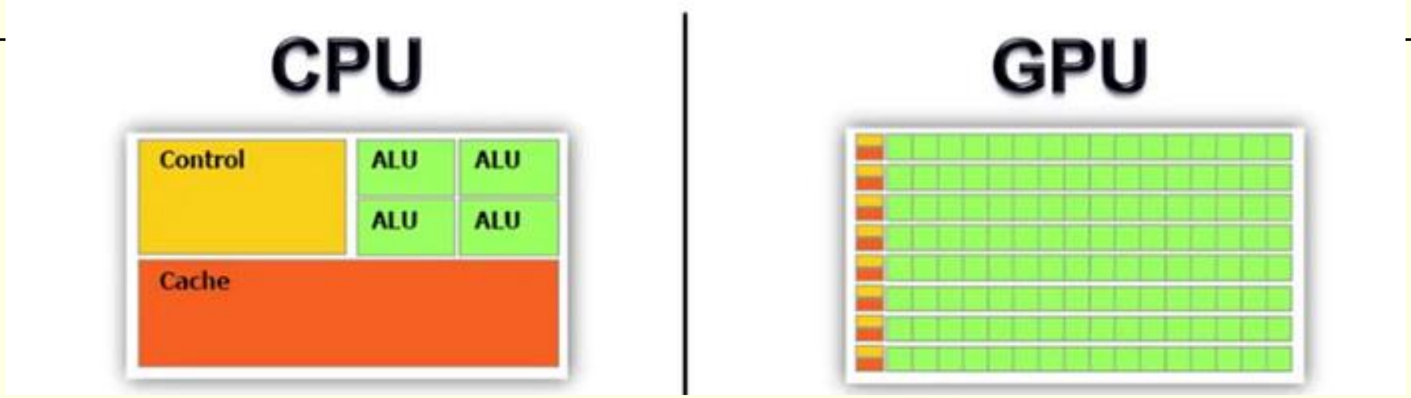
The Most Common Co-Processor:

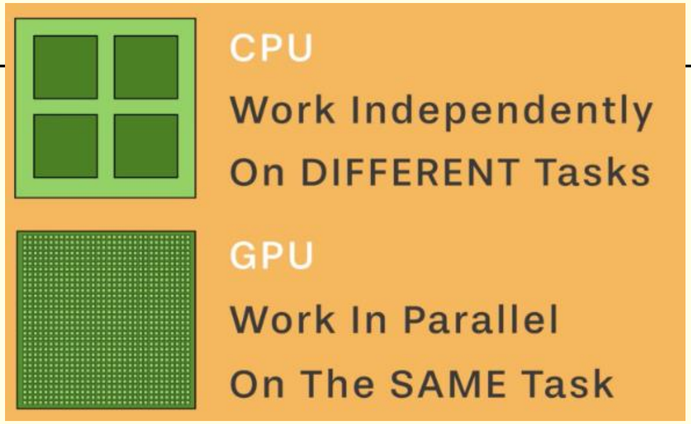
GPU = Graphical Processing Unit.

- Originally used just for **rendering graphics**.
- Now used for **parallel data processing** across many cores.
- Tasks not limited to graphics anymore (e.g. machine learning, scientific simulations).

Differences Between CPUs and GPUs

CPU	GPU
• Fewer cores, higher clock speed • Great for complex, sequential tasks • General-purpose, flexible • Good for **small**, **complex tasks**.	• Thousands of slower cores • Great for simple repetitive tasks • Specialised for parallel data tasks • Good for **large**, **repetitive calculations** on **matrices**, **vectors**, **pixels**.




CPU
Work Independently
On DIFFERENT Tasks

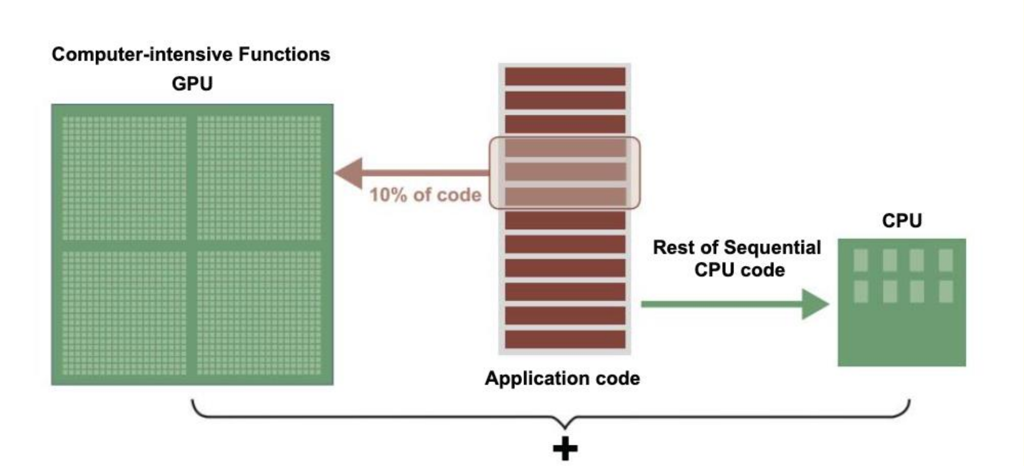
GPU
Work In Parallel
On The SAME Task

Why are GPUs Great for Graphics?

- Break graphics into:
  - **Vertices** with **XYZ coordinates**
  - **Textures**, **lighting**, and **camera angles**
  - All this requires **3D arithmetic, RGB color, and pixel-level** calculation.
  - GPUs handle this in **parallel**, speeding up rendering massively.

What is a GPU?

- In A Level Computer Science, a GPU is responsible for **processing graphics** within the computer to **reduce the load on the CPU**
- **CPUs are general purpose processors** whereas **GPUs are designed specifically for graphics**
- GPUs are likely to have built in circuitry or instructions for common graphics operations
- GPUs can perform an instruction on **multiple pieces of data at one time**
- This is useful when processing graphics (e.g. transforming points in a polygon or shading pixels) which means it can perform transformations to on screen graphics quicker than a CPU
- The GPU can either be **part of the graphics card or embedded in the CPU**
- Modern GPUs typically contain hundreds or even thousands of smaller processing cores, allowing them to perform many operations in **parallel**



What can a GPU be used for besides graphics

Besides graphics processing, a GPU can also be used for:

- **3D modelling**
  - The GPU can be used to **render** lighting effects, textures and shadows
- **Data modelling**
  - As GPUs can handle many calculations simultaneously, they can handle large datasets and complex operations like sorting and filtering data
- **Financial modelling**
  - GPUs are used to simulate different scenarios in risk modelling, option pricing and other financial modelling types
  - Lots of simulations can be run in parallel
- **Data Mining**
  - Data mining is the process of **analysing large amounts of data to find patterns**
  - The main computational tasks are sorting, searching, pattern recognition, statistical analysis and graph algorithms
- **Performing Complex Numerical Calculations**
  - Matrix multiplication and inversion can be done in parallel
  - Numerical Simulations - Physics and engineering simulations often involve solving complex maths models, which can be done in parallel
  - Solving Differential equations
  - Solving differential equations involves computations which can be performed in parallel
- **Machine learning**
  - This involves **training a computer on a massive amount of data** which can be done in parallel. There are lots of matrix multiplications and other computations which can be performed
  - After the training, GPUs can be used to speed up the process of **making predictions** on new data
- **Calculations on multiple data at the same time**
  - There are a number of scenarios where **calculations will be needed to be carried out on multiple data at the same time** e.g. insurance pricing, modelling risk, calculating bills
  - This is done by GPUs rather than CPUs due to being set up **for parallel processing**

What types of task are GPUs suited for?

- GPUs are suited to certain tasks that utilise:
  - **Specialist instructions**
    - GPUs are designed to execute specialist instructions which are common in 3D graphics rendering such as operations on matrices, vectors and geometric transformations
    - These capabilities have been expanded over time and have been generalised which makes GPUs suitable for a wide range of complex calculations besides graphics processing
  - **Multiple cores**
    - Although a CPU can have multiple cores, these are optimised for **serial** processing
    - GPUs have smaller cores but these are optimised for parallel processing
    - GPUs can perform many calculations simultaneously - ideal for tasks that can be broken down into smaller parts
    - This is useful in machine learning and situations where large amounts of data need to be processed
  - **SIMD processing**
    - Single Instruction Multiple Data (SIMD) processing is computers that have multiple processing elements which perform the same operation on multiple data points simultaneously
    - GPUs support SIMD processing as they were originally designed to perform the same operations on multiple pixels or vertices simultaneously - this is a common requirement in image processing, simulations and machine learning

Example

Convert a colour image to grayscale

1. **Split the image** - The GPU divides the image into thousands of small blocks or pixels so each one can be worked on separately.

2. **Send tasks to many cores** -Each GPU core (tiny processor) gets a small piece of the image — like one pixel or a group of pixels — to process.

3. **Process pixels in parallel** -All cores run the same instructions (like adjusting color, brightness, or applying a filter) at the same time on different pixels.

4. **Combine the results** -Once all cores finish, the GPU combines all processed pixels back into a single, complete image.

5. **Display or save the image** -The final image is sent to the screen or stored in memory — ready for viewing or further processing.

What are the benefits of using a GPU?

- There are a number of benefits to using a GPU as well as a CPU (it isn't possible to only use a GPU as the CPU assigns tasks to the GPU)
  - **Parallel processing**
    - GPUs can handle many tasks simultaneously as they are multicore processors
  - **Speed**
    - As GPUs can use parallel processing, this speeds up tasks, particularly those involving large amounts of data or complex computations
  - **Efficiency**
    - GPUs can perform more calculations per unit of power consumed in comparison to CPUs making them more energy efficient when it comes to parallel tasks

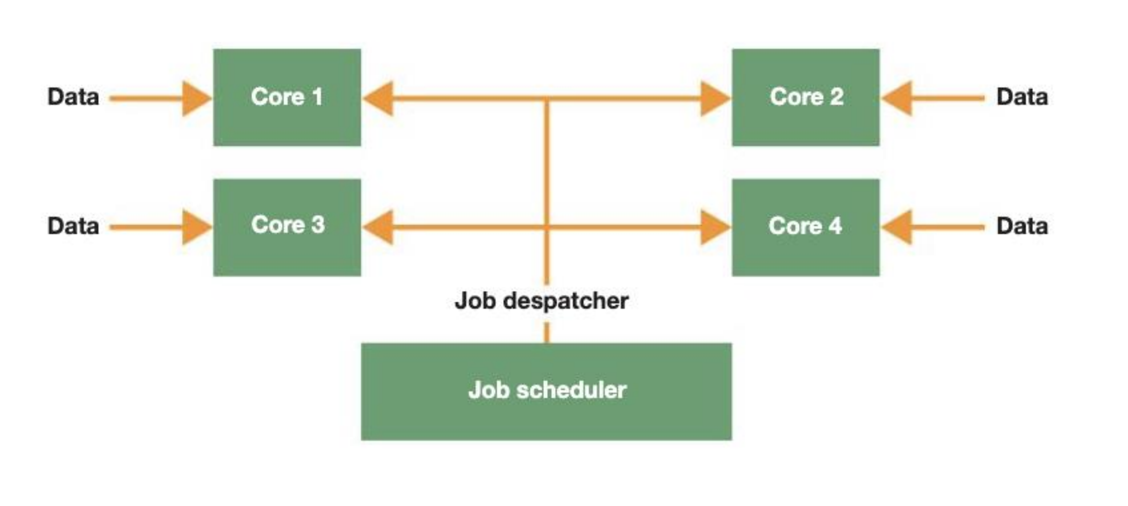
Multicore & Parallel Processors

What is parallel processing?

- In A Level Computer Science, parallel processing is when a computer has **multiple cores**
- Each core can work on the **same task**, to complete it **more quickly**, or each core can work on **separate tasks** at the **same time**

What is multicore processing?

- A multicore system has **more than one processing unit** in a single processor which can **independently process instructions at the same time**
- Parallel processing can also be achieved by utilising more than one processor (a CPU and a GPU)



Benefits and Limitations of Multicore Processors

Benefits	Limitations
Speed: If a task can be divided into subtasks that can be executed simultaneously, the total execution time can be reduced	**Limit on maximum speed**: Even with an infinite number of processors, there is a limit to the maximum speed improvement that can be made using parallel processing if a part of the program can't be parallelised
Improved performance: Simultaneous computation can take place on different data subsets (this would be used in machine learning, data mining and scientific computing)	**Complex programming**: It is harder to write code for parallel processing than serial processing. Tasks have to be synchronised and data shared correctly
Better resource utilisation: Parallel processing allows for better use of computer resources as multi-core or multiple processors can be used more effectively	**Debugging difficulty**: It is more difficult to debug a parallel program than a serial program due to the precise timing of specific events
Problem solving: Problems which are large and complex (which lend themselves to parallel processing) can be solved more easily	**Communication between processors**: Communication between processors can take significant time and resources, potentially outweighing the benefits of using parallel processing
Real-Time applications: Real-time applications including graphics rendering are more feasible and will benefit significantly	**Limited applicability**: Not all tasks can be run in parallel and must be executed serially

What are the benefits of using multicore processors?

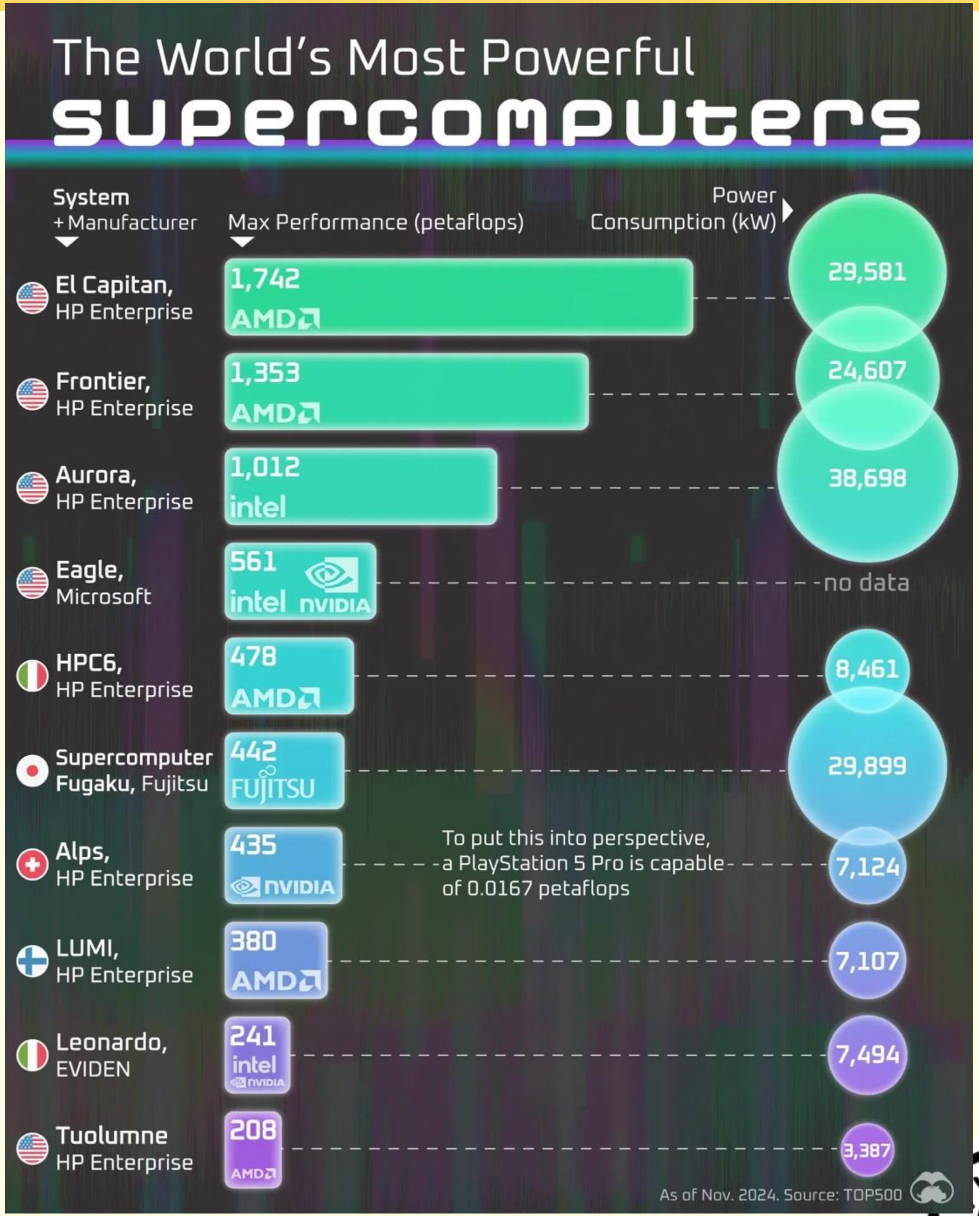
- **Multitasking**
  - Each core can work on a different task - this is particularly effective when the user has multiple applications open at the same time
- **Background tasks**
  - When using a single core processor, a background task like anti-malware scans can slow down the user's other task. A multi-core processor can assign the background task to one core, to reduce the impact on the other task
- **Improved responsiveness**
  - If a program becomes unresponsive, it won't slow the user's computer down as much if they're using multi-core as other cores will continue running their task

Supercomputers

Supercomputers are used on problems such as weather forecasting, running climate change models, processing Big Data or sequencing DNA.

The world's largest and most powerful supercomputer is El Capitan, located at the [Lawrence Livermore National Laboratory in California]() and dedicated in January 2025.

It utilises AMD processors and has **11,039,616 cores**, making it the most powerful computer in the world



To put this into perspective, a PlayStation 5 Pro is capable of 0.0167 petaflops

As of Nov. 2024. Source: TOP500