OS Intro

Background

Von Neumann Architecture

- Based on the stored-program computer concept, where instruction data and program data are stored in the same memory
- Still used in most computers produced today
- Consists of a Control Unit, Arithmetic and Logic Unit (ALU), Memory Unit, Registers, and Inputs/Outputs.

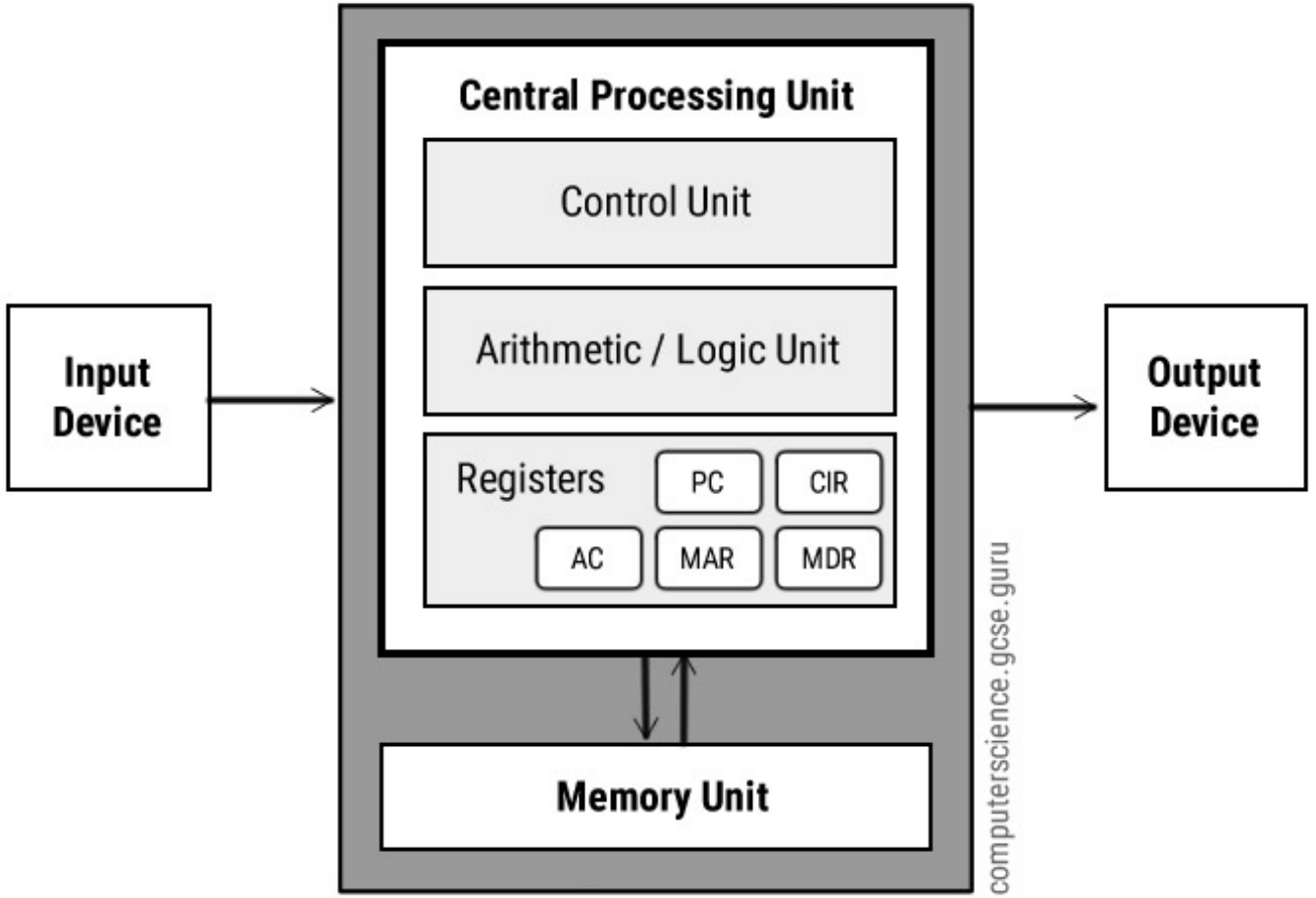

Image from: https://www.computerscience.gcse.guru/theory/von-neumann-architecture

Central Processing Unit

- Electronic circuit responsible for executing the instructions of a computer program
- Sometimes referred to as the microprocessor or processor
- Contains the ALU, CU and a variety of registers

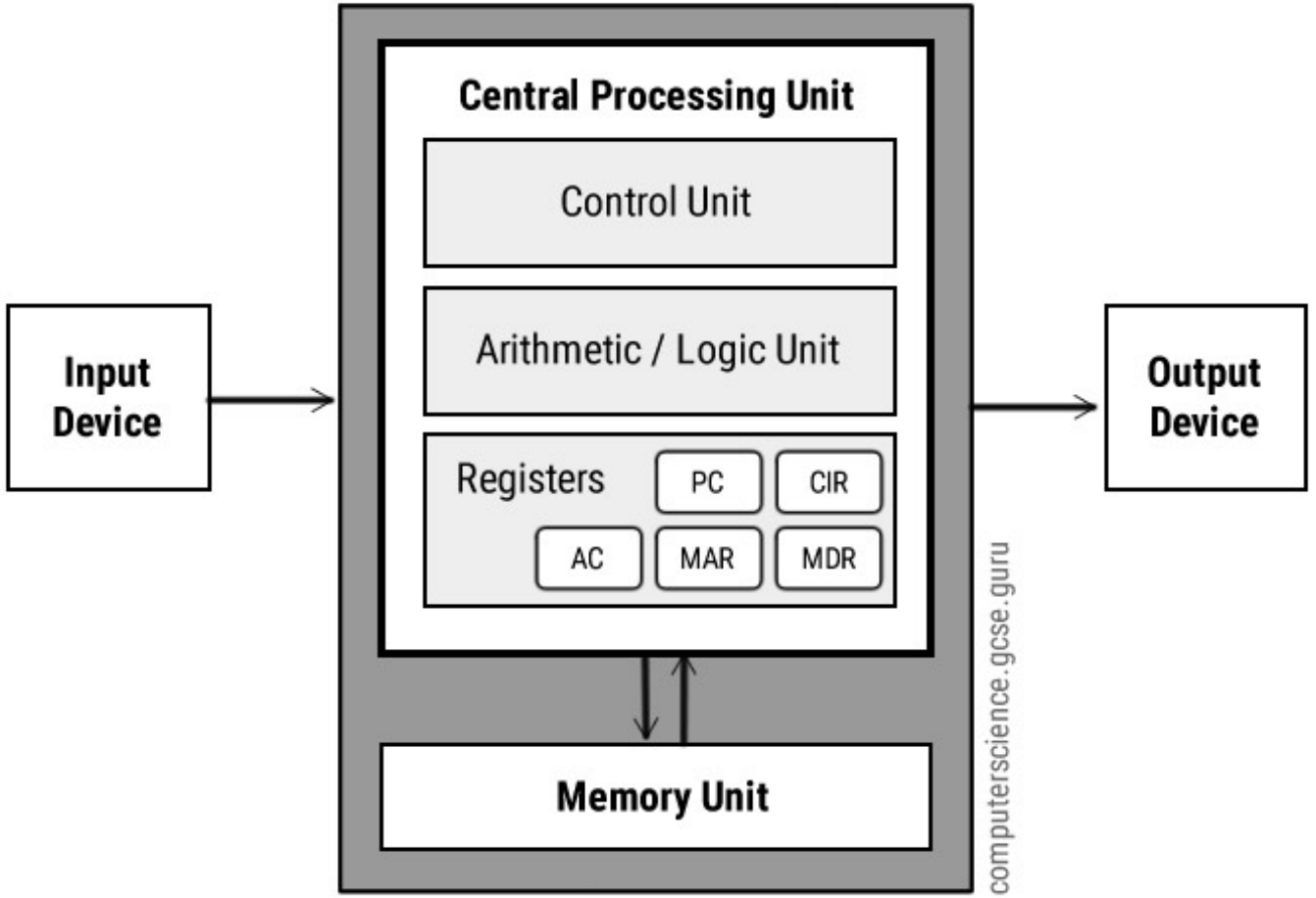

Image from: https://www.computerscience.gcse.guru/theory/von-neumann-architecture

Arithmetic and Logic Unit

- Contains hardware to perform:
  - arithmetic operations (add, subtract etc.)
  - logic (AND, OR, NOT, etc.) operations to be carried out.

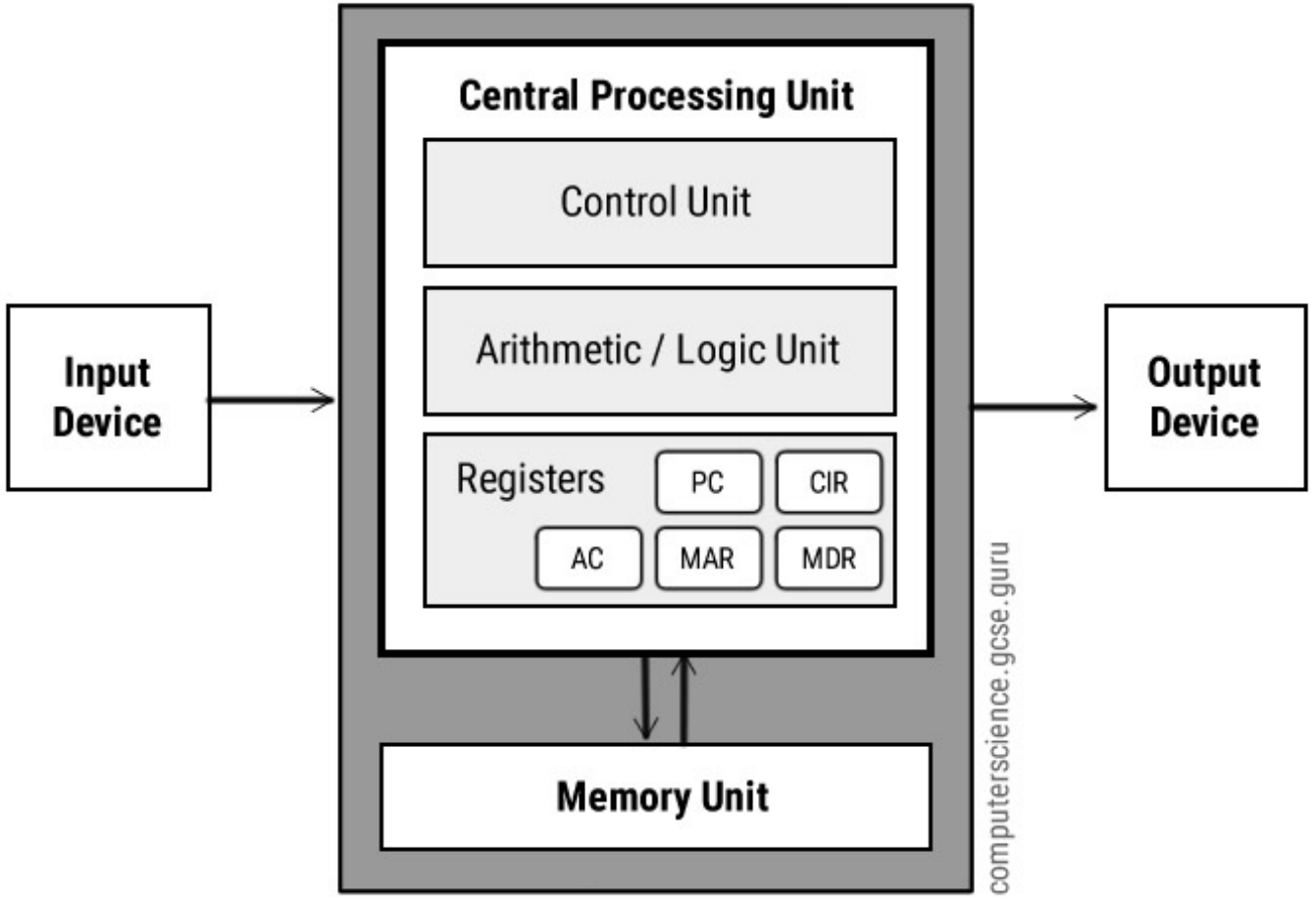

Image from: https://www.computerscience.gcse.guru/theory/von-neumann-architecture

Registers

- High speed storage areas in the CPU
- All data must be stored in a register before it can be processed
- For our purposes, the most important register is the Program Counter (PC) Register
  - contains the address of the next instruction to be executed

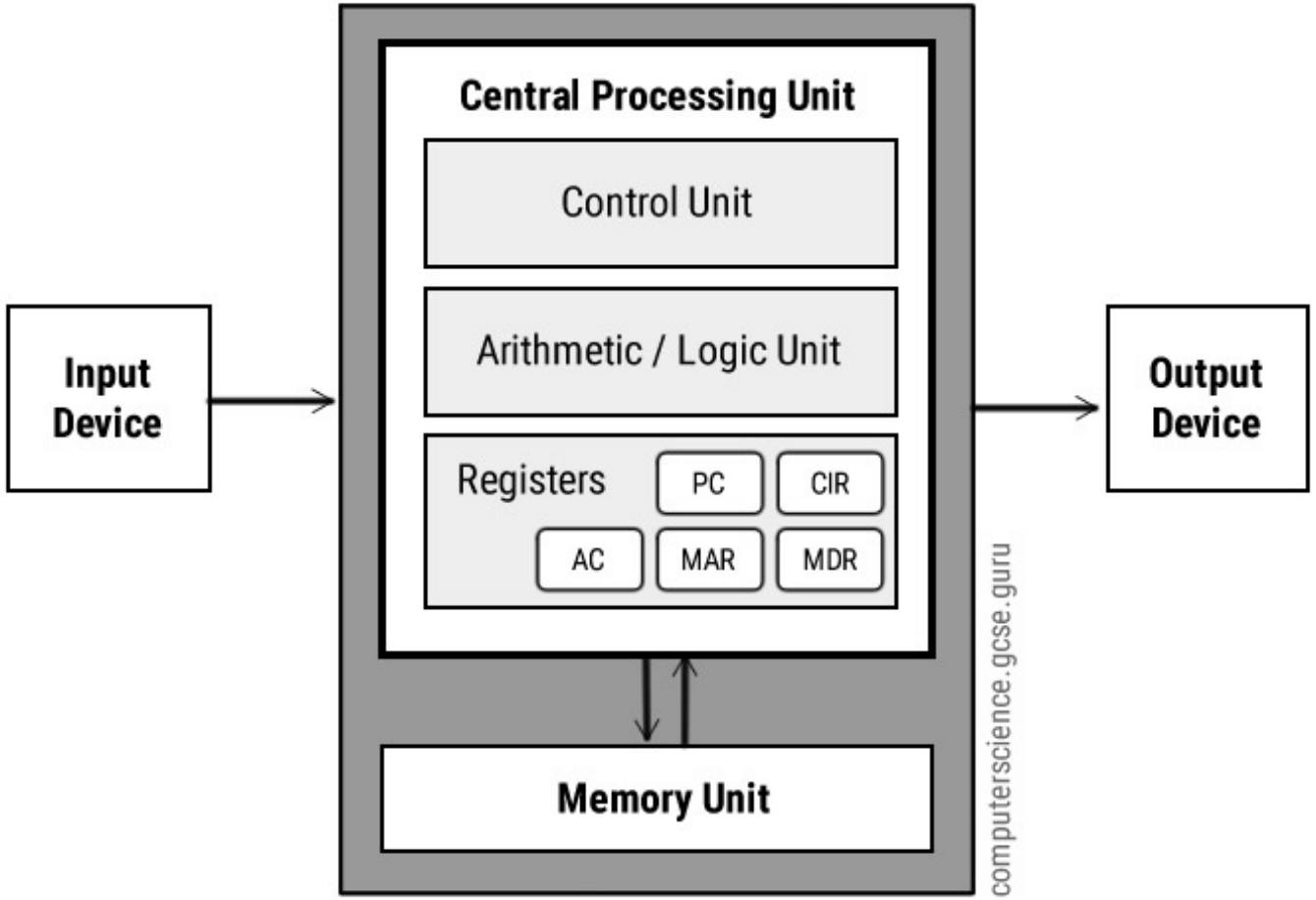

Image from: https://www.computerscience.gcse.guru/theory/von-neumann-architecture

Memory Unit

- RAM or primary or main memory
- Unlike the HDD/SSD (secondary memory), this memory is fast and directly accessible by the CPU
- Data can be loaded from secondary memory or offloaded from registers temporarily when not needed

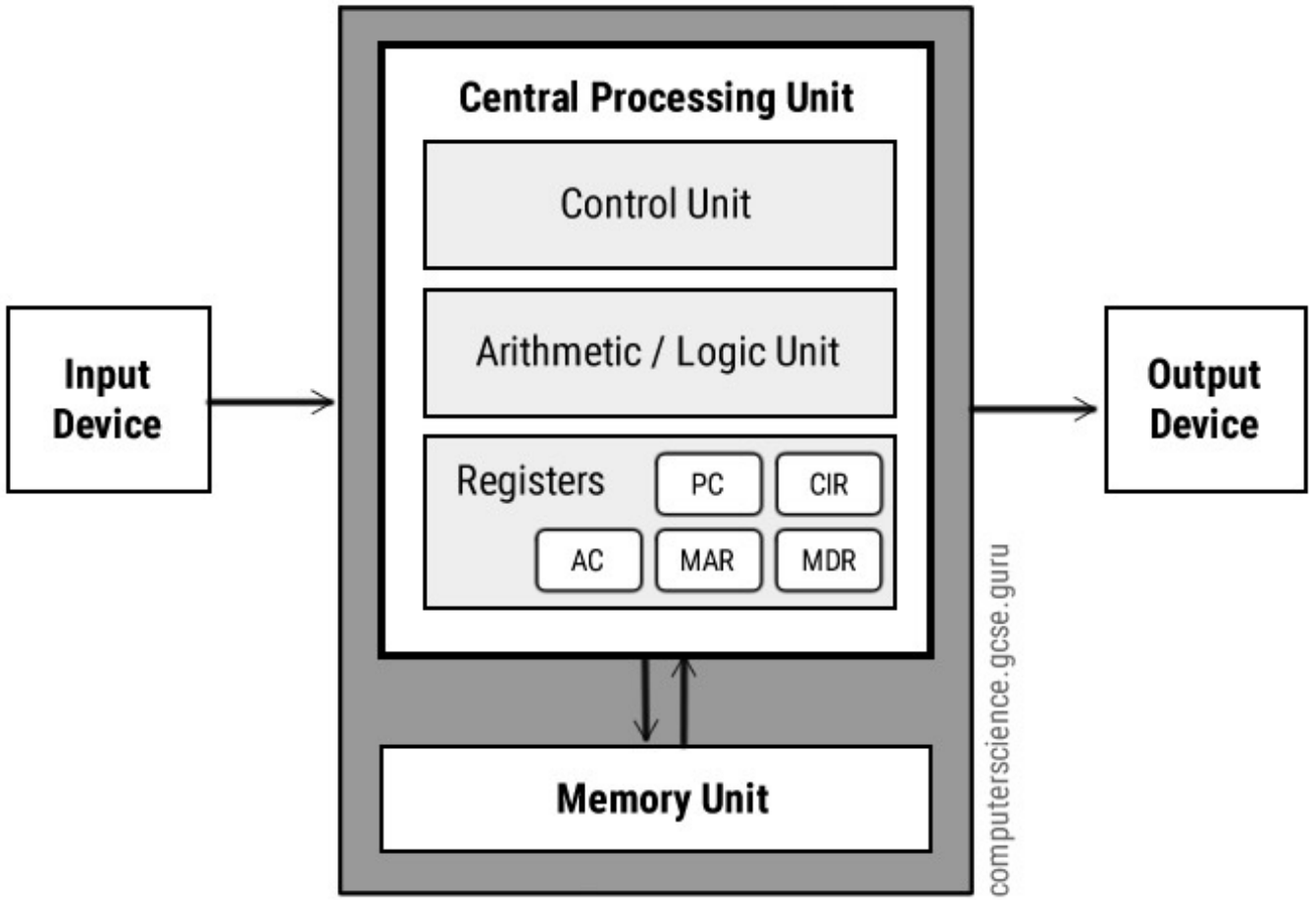

Image from: https://www.computerscience.gcse.guru/theory/von-neumann-architecture

Control Unit

- Coordinates the the function of the ALU, memory and input/output devices, based on program instructions
- Also provides the timing and control signals required by other components
- Fetch, Decode, Execute Cycle

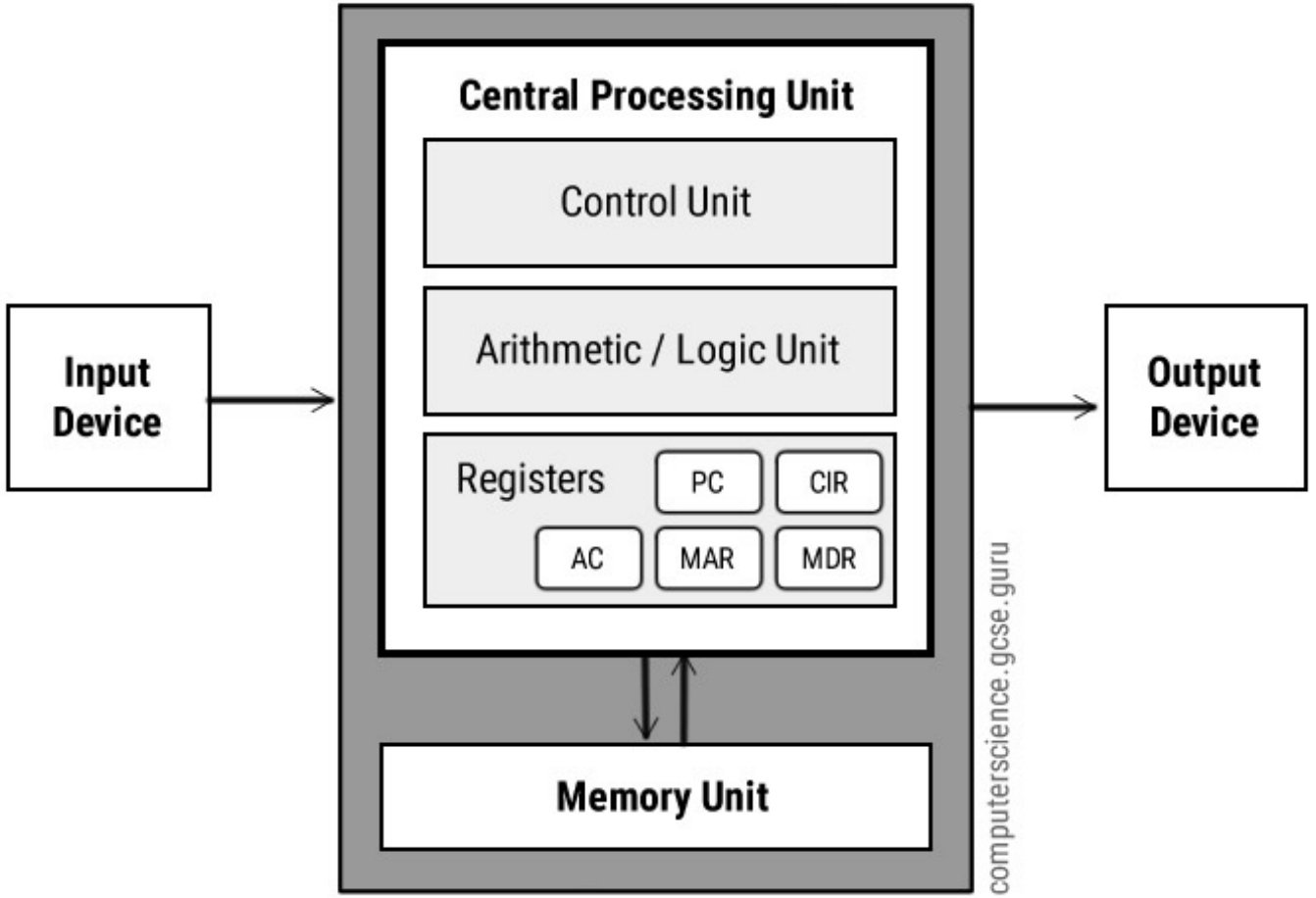

Image from: https://www.computerscience.gcse.guru/theory/von-neumann-architecture

Enter the OS

OS Responsibilities

- Take physical resources and virtualize them
- Manage concurrency to prevent non-deterministic behavior
- Storing data persistently
- Provide abstractions to the hardware for convenient use and development

OS Development Goals

- High performance with minimal overhead
- Secure
  - Minimally, protection between application and between the OS and applications through process isolation
- High reliability
- Energy efficient
- Mobility
  - Devices are getting smaller and more ubiquitous