Introduction

The Operating System is responsible for managing devices that make up a computer system and performing the interaction between the user and these devices;

Hardware

- Processor;
- Main Memory;
- Input/Output Devices;

Software

- Application Programs;
- System Programs.

SO is a software layer between the users' hardware and applications.

Some examples of operating systems

- Mainframe operating systems;
- Server operating systems;
- Multiprocessor operating systems;
- Personal computer operating systems;

- Portable computer operating systems;
- Embedded operating systems;
- Operating systems of sensor nodes;
- Smart card operating systems (smart card).

SO Functions

Basic functions of an operating system:

1. Ease of access to system resources

- Allows access to the various devices (CD/DVD, disks, printer, etc) without requiring the user to know how communication is carried out (read/write operations);

2. Sharing resources in an organized way

- Controls the concurrent execution of all activities performed in a computer system;
- **E.g**.: A shared printer must be controlled in order to avoid printing one user and not interfering with others

Other examples of functions:

- Control machine resources
- Processors; memory space; files; network connections; external devices;
- Access control
- Establish criteria for the use of resources and access order;
- Preventing the memory space violation of concurrent processes and simultaneous access attempts to the same resource
- device management and protection.

Software

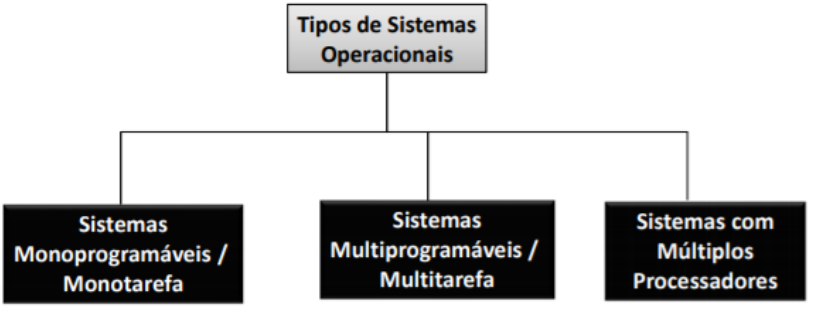
- Request services to the operating system through system calls;
- Similar to program subroutine calls (e.g., PHP, Java, C++) with a few **differences**:
- **Subroutine calls** are transfers to procedures of the program itself;
- **System calls** transfer execution to the operating system (OS);
    - Through parameters, the program informs you what you need.
- The part of the operating system responsible for implementing system calls

is often called the kernel;

The main kernel components of any operating system are:

- Processor management;
- Memory management;
- File System;
- Input and output management.

Types of Operating Systems



Operating System – Batch

Batch

- It was the first type of multiprogrammable OS to be implemented in the 1960s;

- **Feature:**
  - Does not require user interaction with the application;
  - All application data inputs and outputs are implemented by some kind of secondary memory, usually files on disk;

- **Examples of applications processed in batch**:
    - calculations, compilations, sorts, backups and all those where user interaction is not required.

Process

- A process can be understood as a running program;
- A program is a sequence of instructions;
- The process is an active element, which changes its state as it executes a program;
- Several processes can run the same program at the same time;

What is Shell?

Textual interface that interprets the commands existing in an operating system and sends them to the kernel;

A layer that interfaces between the user and the operating system

Shell script

- A programming language for Unix/Linux environments;
- Sequence of commands that are stored in a text-type file;
- Are sequentially executed by the command interpreter (Shell);

Why use Shell Script?

- Task automation;
- Automation of part of the administration;
- Creation of new "custom" commands;
- Time savings in system administration.