

# OMSCS Classes

- [Intro to Operating System](#)
  - [Introduction to Operating System](#)

# Intro to Operating System

# Introduction to Operating System

## Monolithic os

Every hardware service that it will provide or every service that an application that will require from the operating system is part of the operating system already.

Everything is packaged altogether into the operating system, making it really large.

Code might be difficult to maintain, portability isn't that good. Just looking at the linux kernel, it is huge! Gbs

## Modular os

This is what modern linux is.

Uses modules that you can dynamically load into the kernel and run as kernel code.

You can install and uninstall the module without restarting the computer.

Pro: Easier to maintain and smaller footprint, less resources is needed because only the needed functionality is loaded, further functionality can be loaded dynamically when is needed

## Microkernel

Let services that doesn't require kernel privileges run as a normal process, and only keep services that requires privileges in operating system.

Uses inter-process communication because those services that run as normal process will require to communicate with other process in order to provide the interface for services.

But requires frequent elevation of privileges (syscalling), which can be expensive.

Pro: It is small in sizes.

# Linux architecture

On the bottom of the layer, is the hardware itself, then the linux kernel/operating system sits on top of the hardware in order to abstract away the hardware details. The linux kernel runs in privilege mode.

Then on top of the linux kernel lies the user application and standard libraries that runs in user mode. Programs such as shell, editor, compilers, ... etc.

## Syscalling

syscall is the API that allows user mode application to enter privilege mode in order to perform a certain operating system service. syscall such as opening a file, closing a file, or opening a socket.

syscall is a software interrupt and it is raised and handled accordingly by the operating system.

## Signal

Mechanism for operating system to communicate with the user processes.