Operating Systems - Lab Guide

What's an Operating System?

An operating system (OS) is a software that manages computer hardware and software resources and provides common services for computer programs.

Side note: A daemon (/ˈdiːmən/ or /ˈdeɪmən/) is a computer program that runs as a background process. Traditionally daemon names end with the letter d: for example, syslogd is the daemon that implements the system logging facility and sshd is a daemon that services incoming SSH connections. In a Unix environment, the parent process of a daemon is often, but not always, the init process (short for initialization) which is the first process started during booting of the computer system. (P.S: Init is also a daemon).

[Sources: 1, 2, 3]

Why an OS?

- Abstract the hardware for convenience and portability
- Multiplex the hardware among multiple applications
- Isolate applications to contain bugs
- Allow sharing among applications

[Source: 1]

Examples of Operating Systems

- Unix/Linux
- OSX/Mac OS
- Andriod/Symbian/iOS

What are we going to do?!

Understand some basic concepts so we can build a tiny OS ! :)

What're the components of this OS?

1. Booting (Kernel loading)
2. Memory management
3. User environments
4. Preemptive multitasking
5. File system
6. Device Drivers

What do we need?

- QEMU - Machine simulator for development and debugging
- GDB knowledge
- Makefile knowledge
- C programming knowledge
- x86 Assembly knowledge