**Operating Systems I**

**Module 1: Introduction to OS, \*nix, and C**

*Types of System Calls*

* Process control
* File management
* Device management
* Information Maintenance
* Communication
* Protection

*What is an OS and what are the services it provides?*

An operating system is the interface between applications programmers and the

hardware

*What are the different modes of interacting with an OS?*

Interaction by **user** (CLI or GUI) ­and interaction by **program**

*Where did the Unix and Linux OS come from? What is POSIX?*

The roots of unix are in a project called **Multics** (MULTiplexed Information and

Computer Services) developed by AT&T Bell Labs, MIT, and GE (1964).

POSIX (Portable OS Interface) was developed by the IEEE Computer Society to define

standards for software compatibility across Unix variants.

*What is a shell? What are some common shell commands? What are shell scripts and how you can program using shell scripts?*

Interface to the OS. A shell script is a text file containing shell commands that is run as if

the user entered commands into the terminal.

*What is a system call? How you can use system calls in C programs?*

System calls are the mechanism by which a user program asks the OS to perform

services for it. They can be used through library functions or called directly using the C

interface to Linux.

*What is the main entry point of C programs?*

The main() function

*What are the basic data types in C?*

Int, float, double, char

*How does a C program get transformed to machine code for execution?*

Pre-processor -> Compiler -> Assembler -> Linker

*What are compiler warnings and why you should not ignore them?*

Warnings inidicate possible problems with the program that could occur during runtime.