CO211 - Operating Systems Tutorial Sheets

Tutorial 1 - Introduction

1. The issue of resource allocation shows up in different forms in different types of operating systems. List the most important resources that must be managed by an operating system in the following settings;

   (a) Supercomputer

   Since this is most likely used for computation, processor time as well as memory should be carefully managed.

   (b) Workstations connected to servers via a network

   Network access and bandwidth.

   (c) Smartphone

   Energy, since it is a portable device, as well as access to hardware such as the camera, GPS, as well as connectivity (Bluetooth, mobile network, etc).

2. What is the kernel of an operating system?

   The kernel of the operating system is part of the operating system that remains in memory, executing in the privileged part of the CPU.

3. Why is the separation into a user mode and a kernel mode considered good operating system design? Give an example in which the execution of a user process switches from user mode to kernel mode, and then back to user mode again.

   Any bugs executing in user space should not cause the entire system to crash, since the kernel allows for recovery. If all programs were to run in kernel mode, a failure would bring down the entire system. An example of this switch would be writing to disk (or any system call in general).

4. Which of the following instructions should only be allowed in kernel mode, and why?

   (a) Disable all interrupts kernel only

   If something in user were to disable interrupts, the kernel would have no way of regaining control,

   (b) Read the time of day clock user

   All user processes should be able to access the time if needed.

   (c) Change the memory map kernel only

   Managing memory should be restricted to the kernel.

   (d) Set the time of day kernel only

   Processes running in user space should not be able to change the time, as it can cause issues for other processes.

5. A portable operating system is one that can be ported from one system architecture to another with little modification. Explain why it is infeasible to build an operating system that is portable without any modification. Describe two general parts that you can find in an operating system that has been designed to be highly portable.

   Since the operating system must interact with the hardware, it's not feasible to build an OS that can interact with every hardware configuration. Device drivers (**platform specific**) allow for the operating system to interact with hardware, and this can be provided by the hardware manufacturer. Another part would be an API the OS provides to programs (**platform independent**), allowing them to interact with hardware via this abstraction.

Tutorial 2 - Processes + Threads

1.

Tutorial 3 - Scheduling

1.

Tutorial 4 - Synchronisation

1.

Tutorial 5 - Deadlocks

1.

Tutorial 6 - Memory Management

1.

Tutorial 7 - Device Management

1.

Tutorial 8 - Disk Management

1.

Tutorial 9 - File Systems

1.

Tutorial 10 - Security

1.