Processes

Reading 3.1 - 3.5

Do processes just run right away?

How does a process get dispatched (scheduled)?

What if the process is not ready to get dispatched?

When an I/O interrupt occurs, the OS needs to look for the corresponding process.

A better model

A quick look at scheduling

How long would that take?
An even better model

OS control structures

- Memory tables:
  - Allocation of memory to processes
  - Protection attributes for access to shared memory regions
  - Information needed to manage virtual memory

- I/O tables:
  - Manage I/O devices and channels

- File tables:
  - Manage status of files
  - Mostly, this information is maintained and used by a file-management system

Exercise time!
Importance of process block

- The most important data structure in an OS
  - Contains all of the information about a process that is needed by the OS
  - Blocks are read and/or modified by virtually every module in the OS
  - Defines the state of the OS

- Difficulty is not access, but protection
  - A bug in a single routine could damage process control blocks, which could destroy the system's ability to manage the affected processes
  - A design change in the structure or semantics of the process control block could affect a number of modules in the OS

Modes of execution

User mode
- Less-privileged mode
- User programs typically execute in this mode

Kernel / system mode
- More-privileged mode
- Also referred to as control mode or kernel mode
- Kernel of the operating system
Process Creation

- Once the OS decides to create a new process it:
  - Assigns a unique process identifier to the new process
  - Allocates space for the process
  - Initializes the process control block
  - Sets the appropriate linkages
  - Creates or expands other data structures

System Interrupts

**Interrupt**
- Due to some sort of event that is external to and independent of the currently running process
  - Clock interrupt
  - I/O interrupt
  - Memory fault
  - Time slice
- The maximum amount of time that a process can execute before being interrupted

**Trap**
- An error or exception condition generated within the currently running process
- OS determines if the condition is fatal
  - Moved to the Exit state and a process switch occurs
  - Action will depend on the nature of the error the design of the OS

Mode Switching

**If no interrupts are pending the processor:**
- Proceeds to the fetch stage and fetches the next instruction of the current program in the current process

**If an interrupt is pending the processor:**
- Sets the program counter to the starting address of an interrupt handler program
- Switches from user mode to kernel mode so that the interrupt processing code may include privileged instructions

Process switching

Exercise time!