**Chapter 3 Processes**

1. Process Elements
   1. Program code
      1. May be shared with other processes that are executing the same program
   2. Set of data associated with that code
      1. When the processor begins to execute program code, we call the entity a process
   3. Elements
      1. Identifier
      2. State
      3. Priority
      4. Program counter
      5. Memory pointers
      6. Context data
      7. I/O status information
         1. I/O requests, I/O devices, etc.
      8. Accounting information
         1. Processor time and clock time, time limits, etc.
   4. Process Control Block
      1. Contains the process elements
      2. It is possible to interrupt a running process and later resume execution as if the interrupt had not occurred
      3. Created and managed by the OS
      4. Key tool that allows support for multiple processes
2. Process States
   1. Basic:
      1. Trace
         1. Behavior of an indie process by listing the sequence of instructions for that process
         2. Behaviour of the processor can be characterized by showing how the traces of the various processes are interleaved
      2. Dispatcher
         1. Small program that switches the processor from one process to another
3. Process States
   1. Creation
      1. Process spawning
         1. When the OS creates a process at the explicit request of another process
      2. Parent Process
         1. The original, creating process
      3. Child process
         1. New child process
   2. Termination
      1. There must be a means for a process to indicate its completion
      2. A batch job should include a HALT instruction or an explicit OS service call for termination
      3. For an interactive application, the action of the user will indicate when the process is completed (e.g. a log off, quit, etc)
   3. Swapping
      1. Involves moving part or all of a process from main memory to disk
      2. When no processes in main memory are in Ready, the OS swaps one of the blocked processes out on to disk into a suspend queue
   4. Suspend
      1. Process is not immediately available for execution
      2. Process may or may not be waiting on an event
      3. The process was placed in a suspended state by itself, a parent, or the OS
      4. Process may not be removed until the agent orders its removal
4. Tables
   1. Memory Tables
      1. Basic:
         1. Used to keep track of real and virtual memory
         2. Processes are maintained on secondary memory using some sort of virtual memory or simple swapping
      2. Must include:
         1. Allocation of main memory to processes
         2. Allocation of secondary memory to processes
         3. Protection attributes of blocks of main or virtual memory
         4. Information needed to manage virtual memory
   2. I/O Tables
      1. Basic:
         1. Used by the OS to manage the I/O devices and channels of the computer system
         2. At any given time, an I/O device may be available or assigned to a particular process
      2. If I/O is in operation the OS needs to know:
         1. Status of the I/O operation
         2. The location in main memory being used as the source of destination of the I/O transfer
   3. File Tables
      1. Provide info about:
         1. Existence of files
         2. Location on secondary memory
         3. Current status
         4. Other attributes
   4. Process Tables
      1. Basic:
         1. Must be maintained to manage processes
         2. There must be some reference to memory, I/O, and files, directly or indirectly
         3. The tables themselves must be accessible by the OS and are therefore subject to memory management
5. Process Control Structures
   1. Basic:
      1. To manage and control a process the OS must know:
         1. Where the process is located
         2. The attributes of the process that are necessary for management
      2. Process Location
         1. A process must include a program or set of programs to be executed
         2. A process will consist of at least sufficient memory to hold the programs and data of that process
         3. The execution of a program typically involves a stack that is used to keep track of procedure calls and parameter passing between procedures
      3. Process attributes
         1. Each process has a number of attributes that are used by the OS for process control
         2. Collection of program, data, stack, and attributes is referred to as the process image
         3. Process image location will depend on the memory management scheme being used
   2. Process Identification
      1. Each process is assigned a unique numeric id
   3. Processor State information
      1. Consists of contents of processor registers
         1. User-visible registers
         2. Control and status registers
         3. Stack pointers
      2. Program status words (PSW)
         1. Contains condition codes plus other status information
6. Process Creation
   1. Steps:
      1. Assigns a unique pid to the new process
      2. Allocates space for the process
      3. Initializes the process control block
      4. Sets the appropriate linkages
      5. Creates or expands other data structures
7. Mode Switching
   1. If no interrupts are pending the processor:
      1. Proceeds to the fetch stage and fetches the next instruction of the current program in the current process
   2. If an interrupt is pending
      1. Sets the program counter to the starting address of an interrupt handler program
      2. Switches from user to kernel mode so that the interrupt processing code may include privileged instructions