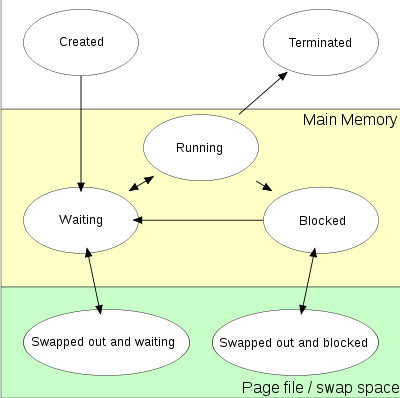
Processes

Define in 20 words or less:

* Process – An instance of a computer program that is being executed.
* Context Switch – The process of storing and restoring the state of a process so that execution can be resumed later. “Change running process”
* Process-Control Block – The data structure in the kernel containing the information needed to manage a particular process.
* Process Scheduling – The method in which threads or processes are given access to system resources.
* Process Starvation – When a process is perpetually denied necessary resources to run.

Draw a diagram defining the ready-execute-wait process cycle:



List four distinct pieces of information that need to be included in a process-control block:

* Process Number
* Program Counter
* Registers
* Scheduling Info
* Memory Limits
* I/O Status Info
* Accounting Info

Describe 3 distinct process scheduling algorithms.

* First-Come, First Served
  + Non-preemptive
    - Process must voluntary relinquish by terminating or by making a system call
  + Intrinsic priority: Arrival time
  + Expected performance?
  + Limitations?
  + Benefits?
* Shortest-Job-First
  + Non-preemptive
  + Better response for short jobs
  + Provable optimal
    - Min avg. waiting time for a pool of jobs
  + Disadvantage(s)?
* Priority
  + Associate priority with each process
    - One queue for each priority level
    - FCFS within each level
  + Problem(s) with priority?
  + SJF is special case
  + Starvation possible
  + Solution?
    - Increase priority with “age”
    - Implement aging?
* Round-Robin
  + Preemptive
  + Process “runs” time-slice (q) units
  + FCFS special case of RR (q = infinity)
  + Time sharing
    - Each process “executes” at speed 1/n
    - q = ???
  + Balance q with context-switch time
* Multilevel Queue
  + Partition *Ready Queue*
  + Each Q has own schedule method
    - E.g. foreground (interactive, RR) and background (batch, FCFS) queues
  + Approaches
    - Higher queues have absolute priority over lower level queues OR
    - Time slice among queues

Compare and contrast Round-Robin and Shortest-Job-First scheduling with respect to “fairness” and “throughput”:

* RR – each process gets a small amount of time before being preempted to the next process. While this is fair, it may take longer for each process. Better response, but longer throughput.
* SJF – Not fair at all (*great* response for short jobs), but better throughput in the long run.

Describe a simple addition to priority scheduling that will prevent starvation:

* Aging. As processes get “older” their priority increases until they are taken care of.