Vocabulary

1. **Multiprogramming**
    - Is the technique of utilizing several programs concurrently in a single computer system
2. **Mechanism**
    - Is low-level methods or protocols that implement a needed piece of functionality
    - Is low-level machinery in OS
    - Does not dictate policies
3. **Policies**
    - Are algorithms for making some kind of decisions within the OS
    - Is high-level intelligence in OS
    - Does not dictate mechanism
4. **CPU Bound**
    - Means the rate at which process progresses is determinted primarily by the speed of the CPU.
    - Has very long CPU bursts, infrequent I/O bursts
5. **I/O Bound**
    - Means the rate at which process progresses is determinted primarily by the speed of I/O Subsystem.
    - Has short CPU bursts, frequent (long) I/O bursts
6. **Non-preemtive Scheduling**
    - Is the type of scheduling that once the CPU has been allocated to a process, it keeps the CPU until it terminates or blocks
7. **Preemtive Scheduling**
    - Is type of scheduling where CPU can be taken from a running process and allocated to another
8. **Context Switching**
    - Is dispatching a process from a ready queue

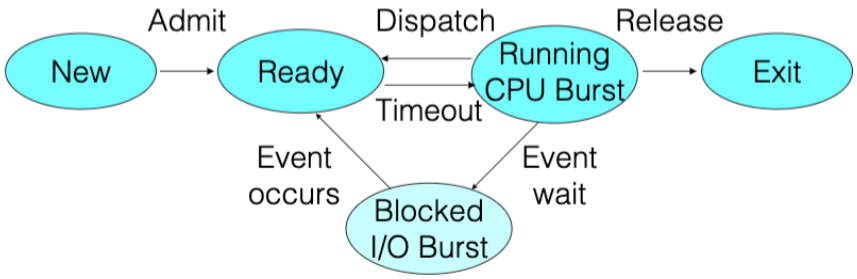
9. **Convoy Effect**
    - All other processes wait for the one big process to release the CPU

1 Thread Life Cycle

- **CPU-Bound:** has very long CPU bursts, infrequent I/O bursts
- **I/O-Bound:** has short CPU bursts, frequent (long) I/O bursts
- **During I/O bursts, CPU is not needed**

2 Recall State Diagram

- Thread/Process is blocked during I/O burst and therefore **does not use CPU**



3 Scheduling Goals

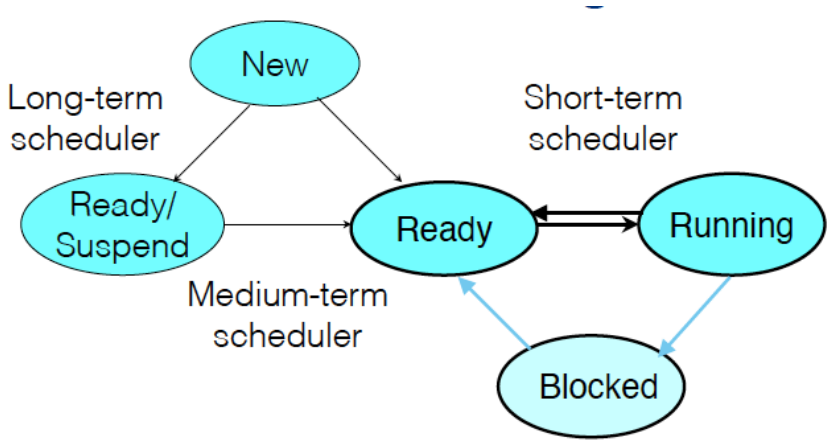
- All Systems
    - Fairness - Each process receives fair share of CPU
    - Avoid starvation
    - Policy enforcement - Usage policies should be met
    - Balance - All parts of the system should be busy
- Bach Systems
    - Throughput - Maximize job completed per hour
    - Turnaround time - Minimize time between submission and completion
    - CPU utilization - Keeps the CPU busy all the time

4 Scheduling Goals

- Interactive Systems
  - Response time - Minimize time between receiving request and starting to produce output
    - Response time = First Run Time - Arrival Time
  - Proportionality - "Simple" tasks complete quickly
- Real-Time Systems
  - Meet deadlines
  - Predictability

5 Process State Diagram

- Dispatching a process from the ready queue is called **context switching**



6 Algorithm: Shortest Job First

- Is optimal with respect to **average wait time**

7 Algorithm: Round Robin

- Designed for time-sharing systems
- Pre-emptive
- Ready queue is circular
- Choice of quantum is chritical
  - We want q to be large w.r.t the context switch time