CPU Scheduling

Chapter 5

Preemptive Scheduling

Time sliced
- Process scheduling takes place when
  - Process blocks on I/O (running to waiting)
  - Process terminates
  - Process is interrupted
  - Process becomes ready – I/O completes (waiting to ready)
- The system is non-preemptive or cooperative if scheduling only takes place under the first two conditions
- The system is preemptive if scheduling takes place under all four conditions

Scheduling Criteria

What is the best algorithm
- CPU utilization — keep the CPU as busy as possible
- Throughput — number of processes finished / time-unit
- Turnaround time — complete time - submission time
- Waiting time — total time in the ready queue
- Response time — elapsed from submission to begin responding

First-Come, First-Served

FCFS (non-preemptive)
- Arrival order: P1, P2, P3
  - Waiting time: P1 = 0, P2 = 24, P3 = 27
  - Average wait: (0 + 24 + 27) / 3 = 17

<table>
<thead>
<tr>
<th>Process</th>
<th>Burst Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>P1</td>
<td>34</td>
</tr>
<tr>
<td>P2</td>
<td>3</td>
</tr>
<tr>
<td>P3</td>
<td>3</td>
</tr>
</tbody>
</table>

Here is a diagram showing the execution of processes P1, P2, P3:

```
0  2  4  6  8  10  12  14  16  18  20  22  24
p1----------p2----------p3----------p4
```

First-Come, First-Served

Sensitive to arrival order
- Arrival order: P2, P3, P1
  - Waiting time: P1 = 6, P2 = 0, P3 = 3
  - Average wait: (6 + 0 + 3) / 3 = 3

<table>
<thead>
<tr>
<th>Process</th>
<th>Burst Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>P1</td>
<td>34</td>
</tr>
<tr>
<td>P2</td>
<td>3</td>
</tr>
<tr>
<td>P3</td>
<td>3</td>
</tr>
</tbody>
</table>

Here is a diagram showing the execution of processes P2, P3, P1:

```
0  3  6  9  12  15  18  21  24  27  30
P2 P3   P1
```

Shortest Job First

SJF (non-preemptive – shortest next CPU burst, p. 160)
- Average wait = (3 + 16 + 9 + 0) / 4 = 28 / 4 = 7

Here is a diagram showing the execution of processes p1, p2, p3, p4:

```
0  2  4  6  8  10  12  14  16  18  20  22  24
p1----------p2----------p3----------p4
```
Shortest Job First
Preemptive – shortest remaining time first (p. 162)

- Average wait
- \((10 - 1) + (1 - 1) + (17 - 2) + (5 - 3)\) / 4
- \((9 + 0 + 15 + 2) / 4 = 6.5\)

\[
\tau_{n+1} = \alpha \tau_n + (1 - \alpha) \tau_n; 0 \leq \alpha \leq 1
\]

Estimating The Next CPU Burst
Exponential average
**Multilevel Queue Scheduling**

- Characterized by
  - Number of queues
  - Scheduling algorithm for each queue
  - Aging (policy to upgrade a process to a higher priority queue)
  - Policy to demote a process to a lower priority queue
  - Policy inserting a new process (choose which queue)

**Multi-Processor Scheduling**

- Asymmetric
  - One processor runs O/S — runs scheduling
  - Other processors only run user code

- Symmetric multiprocessing (SMP)
  - All processors access queues
  - Require synchronization
  - Process may have affinity to a processor (for cache)
  - Processes may be pushed periodically
  - Processes may be pulled to an idle processor (may have a threshold)