CSC415 OPERATING SYSTEM PRINCIPLES

Homework 1 Solution

1. What is the difference between symmetric multiprocessing and asymmetric multiprocessing? (10 Points)

   Sol:
   Symmetric multiprocessing treats all processors as equals, and I/O can be processed on any CPU. Asymmetric multiprocessing has one master CPU and the remainder CPUs are slaves. The master distributes tasks among the slaves, and I/O is usually done by the master only.

2. Why do user programs have to make system calls rather than just executing the code for the system calls themselves? (10 Points)

   Sol:
   System calls execute potentially dangerous operations using privileged instructions. Because of their potential to cause harm to the system, privileged instructions can only be executed by the kernel.

3. What are the four sections in the address space of a process? What is the Process Control Block? (10 Points)

   Sol:
   The address space of a process consists of the text section, the data section, a heap, and a stack.
   Each process is represented by a Process Control Block in the operating system. It contains many pieces of information associated with the process, including process state, program counter, CPU registers, CPU-scheduling information, *etc*.

4. Assuming there are no errors, how many new processes will the following code create? Why? (10 Points)

   ```
   if (!fork())
           fork();
   ```

```
else {
        fork();
        fork();
}
```

Sol:
There are 5 new processes created.
We use P to denote the parent process. When P executes fork() in the condition of if statement, it creates a child process, denoted as C1. After that, P executes the two fork() statements in else, while C1 execute the fork() in the body of if. When C1 executes the fork() in the body of if, it creates a new process, denoted as CC1. Then, both C1 and CC1 terminate. When P runs the first fork() in the body of else, it creates a new process, denoted as C2. Then, both P and C2 execute the second fork(), which will create two new processes, denoted as C3 and CC2. After that, all of them terminate. Hence, five new processes are created in total, i.e., C1, CC1, C2, C3, and CC2.

5. Describe the actions taken by a kernel to context-switch between processes? (10 Points)

   Sol:
   In general, the operating system must save the state of the currently running process and restore the state of the process scheduled to be run next.

6. What are the two models for Inter-Process Communication? What are their differences? (10 Points)

   Sol:
   The two models for Inter-Process Communication are shared memory and message passing.
   In shared memory, cooperating processes establish and share a region of memory. They exchange information by reading data from and writing data to the shared region.
   In message passing, cooperating processes use the message-passing facility provided by OS, i.e., the send() and receive() operations, to communicate with each other.

7. Consider the following set of processes, with the length of the CPU-burst time given in milliseconds: (40 Points)

Process	Burst Time	Priority
*P*1	10	3
*P*2	1	1

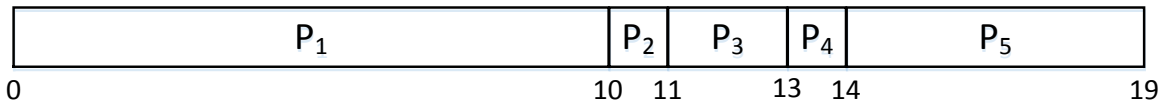
*P*3	2	3
*P*4	1	4
*P*5	5	2

The processes are assumed to have arrived in the order P1, P2, P3, P4, P5, all at time 0.

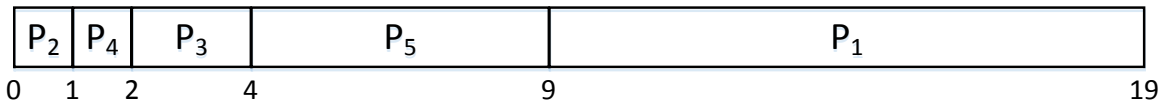
a. Draw four Gantt charts illustrating the execution of these processes using FCFS, SJF, a nonpreemptive priority (a smaller priority number implies a higher priority), and RR (quantum = 1) scheduling.
b. What is the turnaround time of each process for each of the scheduling algorithms in Part a?
c. What is the average waiting time for each of the scheduling algorithms in Part a?

Sol:

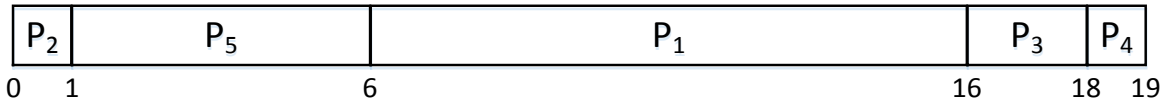
a. FCFS



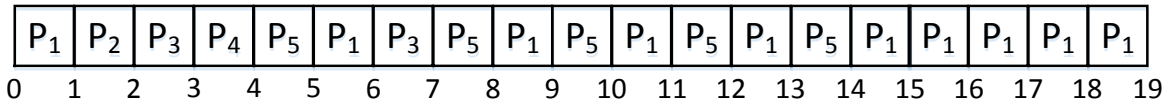
SJF



Nonpreemptive Priority



RR



b. FCFS

P1: 10 P2: 11 P3: 13 P4: 14 P5: 19

SJF

P1: 19 P2: 1 P3: 4 P4: 2 P5: 9

Nonpreemptive Priority

P1: 16 P2: 1 P3: 18 P4: 19 P5: 6

RR
P1: 19 P2: 2 P3: 7 P4: 4 P5: 14

c. FCFS (0+10+11+13+14)/5=48/5=9.6
SJF (9+0+2+1+4)/5=16/5=3.2
Nonpreemptive Priority (6+0+16+18+1)=41/5=8.2
RR (9+1+5+3+9)/5=27/5=5.4