[Sample] Midterm Examination for Operating Systems

(July 29, 2020)

# Name: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Student ID: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Problem 1.** Consider the following set of processes with the length of the CPU-burst time given in milliseconds:

Processes Burst Time

*P1* 4

*P2* 5

*P3* 2

*P4* 1

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

(a) Draw four Gantt charts illustrating the execution of these processes using FCFS (first-com first-served), SJF (shortest-job-first), and RR (round-robin, quantum = 1) scheduling. Write a process number (P1, P2, P3, or P4) in one of cells (time proceeds from left to right).

First-Come First-Served (FCFS)

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| p1 | p1 | p1 | p1 | p2 | p2 | p2 | p2 | p2 | p3 | o3 | p4 |

Shortest-Job-First (SJF)

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| p4 | p3 | p3 | p1 | p1 | p1 | p1 | p2 | p2 | p2 | p2 | p2 |

Round-Robin with FCFS

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| p1 | p2 | p3 | p4 | p1 | p2 | p3 | p1 | p2 | p1 | p2 | p2 |

(b) What is the waiting time of each process for each of the scheduling algorithms in part (a)?

|  |  |  |  |
| --- | --- | --- | --- |
|  | First-Come First-Served | Shortest-Job-First | Round-Robin |
| *P1* | 0 | 3 | 6 |
| *P2* | 4 | 7 | 7 |
| *P3* | 7 | 1 | 5 |
| *P4* | 11 | 0 | 3 |

**Problem 2.** The following is a pseudo-program (procedure) for the dining philosopher problem: 2 philosophers spend their lives thinking and eating. They share a table and 2 chopsticks (1 pair). A philosopher gets 2 chopsticks to eat. After eating, the philosopher puts down the 2 chopsticks.

***Philosopher1 Philosopher2*** **wait**(chopstick1); **wait**(chopstick2); **wait**(chopstick2); **wait**(chopstick1); philosopher 1 eats; philosopher 2 eats; **signal**(chopstick1); **signal**(chopstick2); **signal** (chopstick2); **signal** (chopstick1);

The above procedure may cause a deadlock. Describe why does the procedure cause the deadlock? In addition, draw the resource-allocation graph of this problem.

|  |  |
| --- | --- |
| Reason: | Resource-allocation graph: |

**Problem 3.** Given that memory partitions of 150 KB, 250 KB, 50 KB, 300 KB, and 190 KB (in order), the processes of P1=225 KB, P2=125 KB, P3=50 KB, and P4=115 KB arrive in order and are waiting to place on one of each partition. Note that each partition can hold a single process.

1. How would the processes place on each partition by using the First-fit or Best-fit? Write a process number (P1, P2, P3, or P4) in one of cells.

|  |  |  |  |
| --- | --- | --- | --- |
| **Partitions (Holes)** | **Placement of processes** | | |
| **First-fit** | **Best-fit** |  |
| **150 KB** | p2 | p2 |  |
| **250 KB** | p1 | P1 |  |
| **50 KB** | p3 | p3 |  |
| **300 KB** | p4 |  |  |
| **190 KB** |  | p4 |  |