

University of Asia Pacific
Department of Computer Science and Engineering
Mid-Semester Examination Spring-2020
Program: B.Sc. in Computer Science and Engineering

Course Title: Operating System Course No. CSE 405 Credit: 3.00

Time: 1.00 Hour. Full Mark: 60

There are **Four** Questions. **Answer three questions including Q-1 and Q-2.**

1. a. If you design an Operating System, what will be your design goal? Describe the Computer System structure and describe the role of OS in your own words. [10]
- b. In your opinion which one is better: Symmetric or Asymmetric multi-processing systems? What are their fundamental differences? [10]

2. a. [10]

Process	Burst Time	Arrival Time	Priority
P	10	0	2
Q	5	7	1
R	8	9	3
S	7	10	4

Apply pre-emptive priority scheduling for the given scenario and prepare the grant chart and calculate the average waiting time.

- b. [10]

Process	Burst Time	Arrival Time
A	8	3
B	4	0
C	5	4
D	9	5

Apply round robin algorithm (quantum = 4) for the given scenario and prepare the grant chart and calculate the average waiting time.

3. a. What are the different states of a process? [5]
- b. Draw a transition diagram describing the states of a process explaining the transition between various states. [5]
- c. Write a program using the fork system call where n number of child process is created using the same parent process and each child process will print: [10]
“I am child of “parent id” and my id is “The child id”
And the parent will print:
“I am parent of “n” number of child and my id is “parent id”

OR

4. a. What are the two types of Schedulers. Which one is responsible for multi processing? Explain. [5]
- b. Explain the difference between protection and security? Give real life example. [5]
- c. What will be the outcome of the following code? Mention the two possible variations. [10]

```
#include <stdio.h>
#include <sys/types.h>
#include <unistd.h>

void fork_exp()
{
    int x = 1;

    if (fork() == 0)
        printf("Child has x = %d\n", ++x);
    else
        printf("Parent has x = %d\n", --x);
}

int main()
{
    fork_exp();
    return 0;
}
```