CURTIN UNIVERSITY

Computing Discipline
School of Electrical Engineering, Computing and Mathematical Sciences

Test – S1 / 2018

SUBJECT: OPERATING SYSTEMS Code COMP2006

TIME ALLOWED:

90 minutes test preceded by a 10-MINUTE READING PERIOD during which time only notes may be made. The supervisor will indicate when answering may commence.

AIDS ALLOWED:

To be supplied by the Candidate: Nil

To be supplied by the University: Nil

Calculators are NOT allowed.

GENERAL INSTRUCTIONS:

This paper consists of Four (4) questions with a total of 100 marks.

ATTEMPT ALL QUESTIONS

Name: _____

Student No: _____

Tutorial Time/Tutor: _____

QUESTION ONE (total: 25 marks): Operating Systems.

a) **(4 marks).** A host communicates with an I/O controller using one of the following three methods: polling, interrupt, or direct memory access.

Describe one main advantage and one main disadvantage of using the interrupt method as compared to using polling.

Answer:

b) **(3 marks).** Explain if the following statement **true** or **false**: the purpose of DMA (Direct Memory Access) is so that the processor can read or write directly to a device.

Answer:

c) **(4 marks).** Explain the basic different between the asymmetric multiprocessing and the symmetric multiprocessing systems.

Answer:

d) **(4 marks).** What is the main difference between an interrupt and a trap (or exception)? Is *segmentation violation* an interrupt or a trap or neither? Justify your answer.

Answer:

e) **(4 marks).** What is the purpose of system calls? Describe two examples of system calls.

Answer:

f) **(6 marks).** An operating system typically has two modes of operations: kernel and user modes. Discuss the hardware supports that are used by the OS to implement the modes of operations.

Answer:

END OF QUESTION ONE

QUESTION TWO (total: 23 marks): Processes and threads

a) **(Total: 8 marks).** Assume five different process states as discussed in the textbook / lecture: *New* (the process is being created), *Running* (actually using the CPU at that instant); *Waiting* (unable to run until some external event happens), *Ready* (runnable; temporarily stopped to let another process run); and *Terminated* (the process has finished execution).

Which transition does represent each of the following events? Explain your answer.

- A process is trapped for divide by zero.
- A process is waiting for input from keyboard.
- There is a context switch from process A to process B.

Answer:

- A process is trapped for divide by zero:

- A process is waiting for input from keyboard:

- There is a context switch from process A to process B:

b) **(3 marks).** Explain if the following statement is **true** or **false**: Switching between user level threads of the same process is often more efficient than switching between kernel threads of the same process because unlike the kernel level threads, user level threads share the same memory address space.

Answer:

c) **(4 marks).** How many times does the following program print hello? Explain your answer. **Hint.** How many processes are there to print "hello" when i=0? When i=1? When i=2?

```
main()
{
int i;
for (i=0; i<3; i++)
    printf("hello\n");
    fork();
}
```

Answer:

d) **(4 marks).** Explain one advantage and one disadvantage for two processes to use *shared memory* for communication instead of using *message passing*.

Answer:

e) **(4 marks).** Explain two advantages and two disadvantages of user-level threads.

Answer:

END OF QUESTION TWO

QUESTION THREE (total: 25 marks): Process Synchronization

a) **(Total: 8 marks).**

(i) **(2 marks).** What is a race condition?

(ii) **(2 marks).** What is a critical section?

(iii) **(4 marks).** Give one example to illustrate a critical section and a race condition.

Answer:

(i)

(ii)

(iii)

b) **(Total: 6 marks).**

(i) **(2 marks).** Briefly define **progress** condition for correct solution to critical section problem.

(ii) **(4 marks).** Show that the following two-process solution for the critical section problem **does not** meet the **progress** requirement, but it satisfies the **mutual exclusion** requirement.

```
// Process Pi
repeat
        flag[i] = true;
        while (flag[j] ) do no-op;

        critical section

        flag[i] = false;

        remainder section
until false;
```

Answer:

(i)

(ii)

c) **(3 marks).** Describe what the *Test-and-Set* (*lock*) does.

Answer:

d) **(Total: 8 marks).** Consider two processes that are concurrently executing an instruction $i = i + j$, where i is initialized to 2 and j is initialised to 1. Assume that the instruction is implemented in machine language as:

```
R1 = i;
R2 = j;
R1 = R1 + R2;
i = R1;
```

(i) **(4 marks).** Show that without synchronization, the final value of i can be 3. Note that the correct result should be 4.

(ii) **(4 marks).** Use the wait(A) and signal(A) operations of the semaphore to synchronize the two processes. Note that the initial value of semaphore A must be specified.

Answer:

(i)

(ii)

END OF QUESTION THREE

QUESTION FOUR (total: 27 marks): CPU Scheduling

a) **(Total: 5 marks).** Dispatcher.

(i) **(3 marks).** List three functions of a dispatcher.

(ii) **(2 marks).** What is dispatch latency?

Answer:

(i) Three functions:

(ii) Dispatch Latency:

b) **(4 marks).** Which of the following scheduling algorithms can lead to starvation? FIFO, Shortest Job First, Priority, Round Robin. Justify your answer.

Answer:

c) **(4 marks).** Explain the main reason for CPU scheduling to favour I/O bound processes over CPU-bound processes.

Answer:

d) **(6 marks).** Consider a system running ten I/O-bound tasks and one CPU bound task. Assume that the I/O-bound tasks issue an I/O operation once for every two millisecond of CPU computing and that each I/O operation takes 20 milliseconds to complete. Also assume that the context-switching overhead is 0.1 millisecond and that all processes are long-running tasks. What is the CPU utilization for a round-robin scheduler when:

- The time quantum is 2 milliseconds
- The time quantum is 10 milliseconds

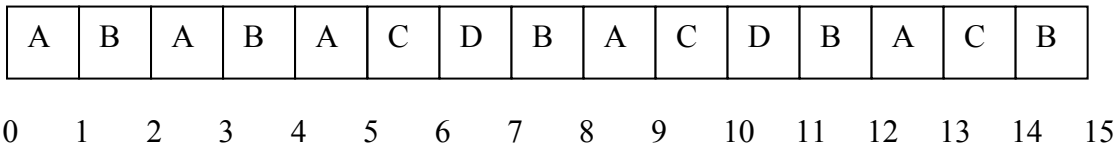
Answer:

e) **(Total: 8 marks).** Consider the following set of processes with their arrival times and burst time. Assume that the context switch overhead is 0.

Process	Arrival Time	Burst Time
A	0	5
B	1	5
C	5	3
D	6	2

(i) **(4 marks).** Draw a Gantt chart for a First-Come-First-Serve Schedule.

(ii) **(4 marks).** Given the following Gantt chart produced by the round robin (RR) scheduling algorithm with quantum 1, compute the average waiting time and average turn around time for part. Assume the Gantt chart is correct.



Answer:

(i) FIFO

(ii) RR Turnaround time:

Waiting time:

END OF TEST PAPER