

### B3.4-R3: OPERATING SYSTEMS

**NOTE:**

1. Answer question 1 and any FOUR questions from 2 to 7.
2. Parts of the same question should be answered together and in the same sequence.

**Time: 3 Hours**

**Total Marks: 100**

1.
  - a) Does a process incur more execution overhead compared to a thread? Justify your answer.
  - b) Distinguish between multiprocessing and multiprogramming.
  - c) What are the "special files" in Unix?
  - d) What is the main difference between worm and virus?
  - e) State the practical limitations of implementing non-preemptive SJF algorithm.
  - f) What is the difference between a long-term scheduler and a short-term scheduler?
  - g) How can a single copy of a text editor be used to serve multiple users in a time-sharing system?

**(7x4)**
  
2.
  - a) What is TLB? Find out the effective memory-access time with an 80% hit ratio and the following access times:  
TLB access time: 20ns; MM access time: 100ns
  - b) Describe the public-key encryption scheme and mention how is it advantageous to the data-encryption standard.

**(8+10)**
  
3. Consider the following page reference during a given time interval for a memory consisting of 5 frames: y,c,z,c,d,a,y,a,e,a,y,f,d,e. Using the i) FIFO replacement strategy and ii) the LRU replacement strategy compare the results. Repeat both FIFO and LRU replacement strategies for memory with 3 frames and same page reference pattern. Comment on the findings and draw a conclusion justifying the adoption of a particular replacement strategy.

**(18)**
  
4.
  - a) What does 'init' do? What happens to the parent process id of a child when the parent terminates before its child? When does a child become 'zombie'?
  - b) With reference to Unix when do the following situations occur?
    - i) Single process table entry contains pointers to the same file table entry.
    - ii) Different file table entries point to the same i-node table entry.
    - iii) Shell 'forks' a copy of itself and 'waits' for the child to terminate.
  - c) How does CPU time-slice affect the Round-Robin algorithm?

**(8+6+4)**
  
5.
  - a) Show and explain an implementation of the classical producer-consumer (producer produces an item, keeps it in a buffer from where the consumer is picking it up) problem using semaphore.

- b) What is dynamic loading? Mention its advantage. How is dynamic linking performed? Mention any disadvantage that you can think of for both the schemes. (10+8)

6.

- a) What is meant by a domain and the rights on it? Describe a *Capability list* and ways of protecting it from user tampering.
- b) Rewrite the following code introducing code parallelism wherever applicable:

```

For i = 1 to k
    a(i) = b(i) + c(i)
For j = 1 to k
    d(j) = x(j) - y(j)
For p = 1 to k
    x(p) = y(p) + b(p)
read(m,n,o,r)
q = m*n + r/o
write(q)

```

- c) Using preemptive SJF(shortest-job-first) algorithm draw the Gantt chart and calculate the average waiting time for the following processes:

| <u>Process</u> | <u>Arrival time</u> | <u>Burst time</u> |
|----------------|---------------------|-------------------|
| P <sub>0</sub> | 0                   | 6                 |
| P <sub>1</sub> | 2                   | 4                 |
| P <sub>2</sub> | 3                   | 10                |
| P <sub>3</sub> | 7                   | 9                 |

(9+4+5)

7.

- a) Where and how “bit vector/table” is used? What are the advantages and disadvantages of the technique?
- b) What is deadlock? How can deadlock be prevented by not allowing “Hold and Wait” ? Is it a feasible policy?
- c) How can synchronization be achieved when two processes communicate by message passing?
- d) Provide a programming example of multithreading giving improved performance over a single-threaded solution.

(5+5+5+3)