

**Sixth Semester B.E. Degree Examination, Dec.2019/Jan.2020**  
**Operating Systems**

Time: 3 hrs.

Max. Marks: 80

**Note: Answer any FIVE full questions, choosing ONE full question from each module.**

**Module-1**

1. a. What is operating system? Explain multiprogramming and time sharing systems. (06 Marks)
- b. Explain dual mode operating in operating system with a neat block diagram. (05 Marks)
- c. What are system calls? Briefly point out its types. (05 Marks)

**OR**

2. a. Explain process states with state transition diagram. Also explain PCB with a neat diagram. (06 Marks)
- b. What is interprocess communication? Explain its types. (05 Marks)
- c. With a neat diagram, explain the concept of virtual machines. (05 Marks)

**Module-2**

3. a. For the process listed below, draw Gantt charts using pre-emptive and non-preemptive priority scheduling algorithm. A larger priority number has a higher priority. Calculate Average Weighing Time and Average turnaround time.

Jobs	Arrival Time	Burst Time	Priority
J <sub>1</sub>	0	6	4
J <sub>2</sub>	3	5	2
J <sub>3</sub>	3	3	6
J <sub>4</sub>	5	5	3

(06 Marks)

- b. Is CPU scheduling necessary? Discuss the five different scheduling criterias used in the computing scheduling mechanism. (05 Marks)
- c. Explain multithreading models. (05 Marks)

**OR**

4. a. Define semaphores. Explain its usage and implementation. (06 Marks)
- b. Explain Reader-Write problem with semaphore in detail. (05 Marks)
- c. What are monitors? Explain dining Philosopher's solution using monitor. (05 Marks)

**Module-3**

5. a. System consists of five jobs (J<sub>1</sub>, J<sub>2</sub>, J<sub>3</sub>, J<sub>4</sub>, J<sub>5</sub>) and three resources (R<sub>1</sub>, R<sub>2</sub>, R<sub>3</sub>). Resource type R<sub>1</sub> has 10 instances, resource type R<sub>2</sub> has 5 instances and R<sub>3</sub> has 7 instances. The following snapshot of the system has been taken.

Jobs	Allocation			Maximum			Available		
	R <sub>1</sub>	R <sub>2</sub>	R <sub>3</sub>	R <sub>1</sub>	R <sub>2</sub>	R <sub>3</sub>	R <sub>1</sub>	R <sub>2</sub>	R <sub>3</sub>
J <sub>1</sub>	0	1	0	7	5	3	3	3	2
J <sub>2</sub>	2	0	0	3	2	2			
J <sub>3</sub>	3	0	2	9	0	2			
J <sub>4</sub>	2	1	1	2	2	2			
J <sub>5</sub>	0	0	2	4	3	3			

Find need matrix and calculate the safe sequence by using Banker's algorithm. Mention the above system is safe or not safe. (06 Marks)

- b. What is dead lock? What are necessary conditions an operating system must satisfy for a dead lock to occur? (05 Marks)
- c. What is a Resource Allocation Graph (RAG)? Explain how RAG is very useful in describing deadly embrace by considering own example. (05 Marks)

**OR**

- 6 a. What are Translation Load aside Buffer (TLB)? Explain TLB in detail with a simple paging system with a neat diagram. (06 Marks)
- b. Given the memory partitions of 100 K, 500 K, 200 K, 300 K and 600 K apply first fit, best fit and worst fit algorithms to place 212K, 417K, 112K and 426K. (05 Marks)
- c. Describe both internal and external fragmentation problems encountered in a contiguous memory allocation scheme. (05 Marks)

**Module-4**

- 7 a. Consider the following page reference stream: 7, 0, 1, 2, 0, 3, 0, 4, 2, 3, 0, 3, 2, 1, 2, 0, 1, 7, 0, 1. How many page faults would occur for LRU and FIFO replacement algorithms assuming 3 frames? Which one of the above is most efficient? (06 Marks)
- b. Explain demand paging system. (05 Marks)
- c. What is thrashing? How can it be controlled? (05 Marks)

**OR**

- 8 a. Explain briefly the various operations performed on files. (06 Marks)
- b. Explain the various access methods of files. (05 Marks)
- c. Explain various allocation methods in implementing file systems. (05 Marks)

**Module-5**

- 9 a. Explain the various Disk Scheduling algorithms with example. (08 Marks)
- b. Explain access matrix method of system protection. (08 Marks)

**OR**

- 10 a. With a neat diagram explain in detail components of a Linux system. (06 Marks)
- b. Explain the different IPC mechanisms available in Linux. (05 Marks)
- c. Explain process scheduling in a Linux system. (05 Marks)

\* \* \* \*

