| U.S.N. | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|

# B.M.S. College of Engineering, Bengaluru-560019

**Autonomous Institute Affiliated to VTU**

## September / October 2023 Semester End Main Examinations

**Programme: B.E.** **Semester: IV**
**Branch: Artificial Intelligence and Machine Learning** **Duration: 3 hrs.**
**Course Code: 22AM4PCOPS** **Max Marks: 100**
**Course: Operating Systems**

**Instructions**: 1. Answer any FIVE full questions, choosing one full question from each unit.
2. Missing data, if any, may be suitably assumed.

**Important Note:** Completing your answers, compulsorily draw diagonal cross lines on the remaining blank pages. Revealing of identification, appeal to evaluator will be treated as malpractice.

| | | UNIT - I | CO | PO | Marks |
|---|---|---|---|---|---|
| 1 | a) | Explicate the working of read system call diagrammatically. | CO1 | PO2 | **10** |
| | b) | Explain thread implementation in user space and kernel space. | CO1 | PO2 | **10** |
| | | **UNIT - II** | | | |
| 2 | a) | Explain producer consumer problem using semaphores. | CO2 | PO2 | **10** |
| | b) | Consider the following set of processes and answer the following questions:<br>Process \| Arrival Time \| Burst Time<br>P1 \| 2 \| 6<br>P2 \| 5 \| 2<br>P3 \| 1 \| 8<br>P4 \| 0 \| 3<br>P5 \| 4 \| 4<br>i. Draw Gantt charts showing the execution of these processes using FCFS, preemptive SJF, non-preemptive SJF.<br>ii. Compute Turnaround time and waiting time for each process. | CO2 | PO3 | **10** |
| | | **OR** | | | |
| 3 | a) | Justify how readers writers' problem can be used to access database efficiently. Support your answer with code snippet. | CO2 | PO4 | **10** |
| | b) | For the processes listed below, draw Gantt charts using pre-emptive and non-preemptive priority scheduling algorithm. Assume a large priority number has a higher priority. Calculate average weighting time and average turnaround time. | CO2 | PO3 | **10** |

| | | | | | | |
|---|---|---|---|---|---|---|
| | | | <table><tr><th>Process</th><th>Arrival time</th><th>Burst time</th><th>Priority</th></tr><tr><td>$P_1$</td><td>0</td><td>6</td><td>4</td></tr><tr><td>$P_2$</td><td>3</td><td>5</td><td>2</td></tr><tr><td>$P_3$</td><td>3</td><td>3</td><td>6</td></tr><tr><td>$P_4$</td><td>5</td><td>5</td><td>3</td></tr></table> | | | |
| | | **UNIT – III** | | | |
| 4 | a) | Consider the following page reference string:<br>4, 7, 6, 1, 7, 6, 1, 2, 7, 2.<br>Assuming 3 empty frames in memory find number of page faults would occur for:<br>i) Optimal Page Replacement Algorithm.<br>ii) FIFO Page Replacement Algorithm.<br>iii) LRU Page Replacement Algorithm. | *CO2* | *PO3* | **10** |
| | b) | Illustrate the mapping between virtual and physical memory address with an example. | *CO2* | *PO2* | **10** |
| | | **OR** | | | |
| 5 | a) | Justify the need of demand paging. With a neat diagram explain the steps in handling the page fault. | *CO2* | *PO2* | **10** |
| | b) | Demonstrate the working of the following page replacement algorithms with suitable examples for each.<br>i) Second Chance algorithm<br>ii) Working set model | *CO2* | *PO3* | **10** |
| | | **UNIT - IV** | | | |
| 6 | a) | Assume a disk with 200 tracks and disk request queue has random requests in it. The requested disks in the order are as follows:<br>55 ,58 ,39 ,18 ,90 ,160 ,150 ,38 ,184<br><br>Starting from track 100, calculate average seek length using FCFS, SSTF, SCAN algorithm. Examine the best disk scheduling algorithm for the above scenario? | *CO2* | *PO3* | **10** |
| | b) | Explain any five system calls relating to file operations in detail with syntax for each. | *CO2* | *PO2* | **10** |
| | | **UNIT - V** | | | |
| 7 | a) | Explain the various ways of deadlock recovery and the necessary conditions for a deadlock to occur. | *CO2* | *PO2* | **10** |
| | b) | Consider the following snapshot of a system: | *CO3* | *PO2* | **10** |

| Process | Allocation | | | | Max | | | | Available | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | A | B | C | D | A | B | C | D | A | B | C | D |
| P0 | 2 | 0 | 0 | 1 | 4 | 2 | 1 | 2 | 3 | 3 | 2 | 1 |
| P1 | 3 | 1 | 2 | 1 | 5 | 2 | 5 | 2 | | | | |
| P2 | 2 | 1 | 0 | 3 | 2 | 3 | 1 | 6 | | | | |
| P3 | 1 | 3 | 1 | 2 | 1 | 4 | 2 | 4 | | | | |
| P4 | 1 | 4 | 3 | 2 | 3 | 6 | 6 | 5 | | | | |

Apply Banker's algorithm to find:

i) Is the system being safe? If so, give the safe sequence.

ii) If process P2 requests (0, 1, 1, 3) resources can it be granted immediately?

******