S.No. : 565 NCS 4402

No. of Printed Pages : 09

Following Paper ID and Roll No. to be filled in your Answer Book.

PAPER ID : 43210 Roll No. 1 2 3 0 4 3

B. Tech. Examination, 2024-25

(Even Semester)

OPERATING SYSTEMS

Time : Three Hours] *[Maximum Marks : 60*

Note :– Attempt all questions.

SECTION–A

1. Attempt all parts of the following : 8×1=8

   (a) Kernel-level threads cannot share the code segment. True / False

   (b) Consider an arbitrary set of CPU-bound processes with unequal CPU burst lengths submitted at the same time to a computer system. Which process scheduling algorithms would minimize the average waiting time in the ready queue?

[P. T. O.

(c) A file is organized so that the ordering of data records is the same as or close to the ordering of data entries in some index. Then that index is called.

(d) If every edge in the Resource Allocation Graph (RAG) is an allocation edge, then it implies a deadlock. True / False

(e) Threads from different processes can share the same memory space. True / False

(f) A system has 6 identical resources and N processes competing for them. Each process can request atmost 2 resources. What is the maximum value of N that could lead to a deadlock?

(g) In demand paging, when are pages loaded into memory?

(h) In a paging system, what is the primary function of the page table?

SECTION – B

2. Attempt any two parts of the following : $2 \times 6 = 12$

(a) Define CPU scheduling. Explain any two CPU scheduling algorithms with their advantages and disadvantages.

(b) Consider a system having m resources of the same type. These resources are shared by 3 processes A, B and C which have peak demands of 3, 4 and 6 respectively. For what value of m, deadlock will not occur?

(c) Calculate the size of memory if its address consists of 22 bits and the memory is 2-byte addressable.

(d) Explain user-level and kernel-level threads. Compare them based on context switching, performance and OS support.

SECTION – C

Note :– Attempt all questions. Attempt any two parts from each questions. 8×5=40

3. (a) Explain the necessary conditions for deadlock with suitable examples. Describe any two methods for handling deadlocks in operating systems.

(c) Suppose a system has 12 magnetic tape drives and at time t_0 three processes are allotted tape drives out of their as given below :

	Maximum Needs	Current Needs
P0	10	5
P1	4	2
P2	9	2

At time t_0 the system is in safe state. What will be safe sequence so that deadlock is avoided?

4. (a) Consider three concurrent processes P1, P2 and P3 as shown below, which access a shared variable D that has been initialized to 100 : P1:: D = D + 20 P2:: D = D + 50 P3:: D = D + 10. The processes are executed on a uniprocessor system running a time-shared operating system. If the minimum and maximum possible values of D after the three processes have completed execution are X and Y respectively, then calculate the value of X-Y.

(b) The following two functions P1 and P2 that share a variable B with an initial value of 2 execute concurrently.

P1 () {	P2 () {
C = B – 1;	D = 2 * B;
B = 2 * C	B = D – 1;
}	}

What will be the number of distinct values that B can possibly take after the execution?

(c) Does Peterson's algorithm satisfy the three conditions of a critical section solution : mutual exclusion, progress and bounded waiting? Justify each.

5. (a) Consider the following solution to the producer-consumer synchronization problem. The shared buffer size is N. Three semaphores empty, full and mutex are defined with respective initial values of 0, N and 1. Semaphore empty denotes the number of available slots in the buffer, for the consumer to read from. Semaphore denotes the number of available slots in the buffer, for the producer to write to. The placeholder variables, denoted by P, Q, R and S, in the code below can be

assigned either empty or full. The valid semaphore operations are : wait ( ) and signal ( ).

Producer :	Consumer

do {
wait(P);
wait(mutex);
//Add item to buffer
signal (mutex)
signal (Q);
} while (1);
	do {
	wait (R);
	wait (mutex);
	//consume item for buffer
	signal (mutex)
	signal (S);
	} while (1);

What will be assignments to P, Q, R and S will yield the correct solution?

(b) Demonstrate how semaphores can be used to enforce mutual exclusion in a critical section :

(i) Write pseudocode for wait() and signal() operations.

(ii) Show how mutual exclusion is maintained between two processes.

(iii) Explain what could happen without semaphores.

(c) A database allows multiple readers but only one writer at a time. Initial semaphore values :

mutex = 1, rw_mutex = 1, read_count = 0

Sequence of operations :

Reader R1 enters → R2 enters → Writer W1 tries to enter → R1 exits → R2 exits

Explain :

(i) Trace the changes in semaphore values and read_count.

(ii) Indicate when W1 is blocked and when it can proceed.

(iii) Describe how semaphores enforce synchronization here? 5

6. (a) Given the following processes :

Process	Arrival Time	Burst Time
P1	0 ms	5 ms
P2	1 ms	4 ms
P3	2 ms	2 ms
P4	3 ms	1 ms

Explain :

(i) Simulate round Robin scheduling (Quantum = 3 ms).

(ii) Create a Gantt chart.

(iii) Calculate waiting time, turnaround time and average waiting time.

(b) Explain the structure and role of the Process Control Block (PCB) is an operating system. What information does it store and how is it used during context switching? Illustrate with a diagram.

(c) Differentiate between shared memory and message passing as IPC mechanisms. Discuss the advantages, disadvantages and use cases of each.

⌘⌘⌘