Time: 3 Hrs   MM: 50

Note: Attempt all questions.

Q1. (a) Consider a set of n tasks with known runtimes r1, r2, ... rn to be run on a uniprocessor machine. Identify the CPU scheduling algorithm that will result in the maximum throughput and why?

(b) Illustrate spinlock and busy-waiting conditions?

(c) Why can't you disallow mutual exclusion in order to prevent deadlocks? Mutual exclusion is important for which type of OS services? Discuss.

(d) A paging scheme uses a Translation Lookaside buffer (TLB). The effective memory access takes 160 ns and a main memory access takes 100 ns. Determine the TLB access time (in ns) if the TLB hit ratio is 60% and there is no page fault?   (2 x 4)

Q2. (a) Discuss situations in which the MFU page-replacement algorithm generates fewer page faults than the LRU page-replacement algorithm. Also identify under what circumstances the opposite holds.

(b) Consider a multilevel feedback queue scheduling (MLFBQ) with three queues q1, q2 and q3. q1 and q2 use round-robin algorithm with time quantum (TQ) = 5ms, and 4ms respectively. q3 uses FCFS algorithm. Find the average waiting time and average turnaround time for executing the following process?

Process	Burst time (ms)
P1	8
P2	22
P3	4
P4	12

                                                                                                   (2, 4)

Q3. (a) Can a multithreaded solution using multiple user level threads achieve better performance on multiprocessor system than on a single processor? Explain.

(b) Consider the following code segment:

```
pid_tpid;
pid = fork();
if (pid == 0) {/* child process */
fork();
thread_create(...);
}
fork();
```

i. Estimate how many unique processes are created?
ii. Estimate how many unique threads are created?   (2, 2+2)

Q4. (a) Identify the circumstances under which the rate monotonic scheduling is inferior to earliest deadline first scheduling in meeting the deadlines associated with each

process. Consider two processes where p1=50, t1=25 and p2=75, t2=30. Illustrate the scheduling of these processes using earliest deadline first scheduling.

(b) Implement `wait()` and `signal()` operations for a system using the `test_and_set()` instruction. Annotate how your solution is handling busy waiting.

(4, 4)

Q5. (a) Explain solution to the dining philosophers' problem using semaphores. Don't write the code, only explain the strategy.

b) Consider the following pseudocode; where S is a semaphore initialized to 5 in line #2 and counter is a shared variable initialized to 0 in line #1. Assume that the `counter++` operation in line #7 is not atomic. Detect the possible values that Counter can have at the end of execution if 5 threads execute function `parop` simultaneously? Explain. Also check if deadlock is possible in given operation or not.

```
int counter = 0;
Semaphore S = init(5);
void parop(void)
{
    wait(S);
    wait(S);
    counter++;
    signal(S);
    signal(S);
}
```
(3, 3)

Q6. (a) What happens when an instruction cannot be executed due to a page fault may be during instruction fetch or during operand fetch? How will this instruction get executed eventually? Consider different scenarios. (3)

(b) Define Thrashing? Describe the methods to avoid it. (3)

(c) Illustrate the goals, principles, and domain of protection of operating systems. (3)

Q7. (a) Let a disk drive has 5000 cylinders from 0 to 4999. Currently drive is at 143rd cylinder, and the previous request was at cylinder 125. Queue of pending request in FIFO order 1|86, 1470, 913, 1774, 948, 1509, 1022, 130. Evaluate the total distance the disk arm moves to satisfy all the pending requests for each of the following disk scheduling algorithms from current position i) FCFS ii) LOOK.

(b) Contrast the performance of techniques of allocating disk blocks i.e. continuous, linked and indexed for sequential and random access. (4, 3)

CO	Mapping of COs with Questions for attainment purpose
CO1	1(b), 2(a), 7(b)
CO2	4(a), 4(b), 6(b)
CO3	1(b), 1(c), 2(b), 3(a), 3(b), 5(a), 5(b)
CO4	1(d), 2(a), 6(a), 7(a)
CO5	6(c)

THE END