CS 4310 Operating Systems
Exam 1
Max: 200 points
(10/23/2025)

Name : Devin Khun 016117972

Read these instructions before proceeding.

- **Closed book. Closed notes. You can use calculator.**
- **You have 75 minutes to complete this exam.**
- **Important Notes:**
  - *During the exam, all students need to join the Zoom meeting*
  - *No questions will be answered during the exam about the exam questions. Write down your assumptions and answer the best that you can. In the last 15 minutes of the exam, if you have additional questions about submitting your exams, you can ask your questions on the Chat room of Zoom Meeting and your questions will be answered there.*
  - *Just in case you have trouble of submitting your exam here @Canvas, alternative way is to submit your completed exam to Prof. Young by emailing*
    gsyoung@cpp.edu
- **Turn in your completed exam in one PDF file including all pages here @Canvas.**

Possible ways to submit your exam:
(1) Print out the exam paper. Write your answers on the exam paper. Scan your completed exam papers or take photos of them. Then turn in **one PDF file** here @Canvas .
(2) Read the exam from the computer screen and answer questions on your own white papers (number your answers). Scan your exam answers or take photos of them. Then turn in **one PDF file** here @Canvas.
(3) Use your own way to prepare your completed exam in **one PDF file** and turn in here @Canvas.

Q.#1 (36)	Q.#2 (30)	Q.#3 (30)	Q.#4 (44)	Q.#5 (36)	Q.#6 (24)	Total (200)

1. (36 points)
Fill in the blanks (4 pts each)

(a) Operating System provides interface between software and hardware of a computer system.

(b) Two of the classical IPC (Inter-Process Communication) problems are the Sleeping Barber Problem and Dining Philosophers Problem.

(c) One main feature of the third generation operating system is using multiprogramming technique to save CPU time.

(d) Threads are processes in a process.

(e) A semaphore is accessed only through two standard atomic operations

down (or P) and up (or V).

(f) To implement the process, operating system maintains each process's information in its process control block (PCB).

(g) Give two benefits of thread (compared to process).

Threads share resources

Utilization of multi-processor systems to shorten run time.

2. (30 points)

(a) **(12 points)** Briefly explain critical section problem.

The critical section problem is the issue of ensuring that when multiple processes or threads access shared resources (like variables, files, or memory), only one process executes its critical section (the part of code that accesses the shared resource) at a time. The goal is to prevent race conditions, ensuring mutual exclusion, progress, and bounded waiting.

(b) **(18 points)** Briefly describe two solutions we covered in the class for solving the critical section problem.

Peterson's Solution: A software-based solution for two processes. Uses two shared variables flag[i] (indicates a process wants to enter its critical section) and turn (whose turn it is). It ensures mutual exclusion and satisfies progress and bounded waiting without using hardware support.

Test-and-Set Lock (Hardware Solution): The Test-and-Set instruction is a hardware-supported atomic operation used to solve the critical section problem. It works by testing a lock variable and setting it in one indivisible step which means no other process can interrupt during that operation. A shared lock variable is initialized to false (unlocked). When a process wants to enter its critical section, it repeatedly performs TestAndSet(lock) until it returns false. If it returns false, the process has acquired the lock and can enter the critical section. If it returns true, another process already holds the lock, so it keeps looping (busy waiting). When the process leaves the critical section, it sets lock=false.

3. (30 points) (10 points each)
Disk requests come in to the disk driver for cylinders at time zero in the order of
17, 27, 4, 11, 32, 2, 36.

Starting position is 30. A seek takes 20 msec per cylinder moved.

How much seek time is needed for

(a) FCFS (First-come, first served)

$30 \rightarrow 17 \rightarrow 27 \rightarrow 4 \rightarrow 11 \rightarrow 32 \rightarrow 2 \rightarrow 36$

cylinders: $13 + 10 + 23 + 7 + 21 + 30 + 34 = 138$

total seek time = 138×20 ms = 2,760 ms

(b) SSF (Shortest Seek First)

$30 \rightarrow 32 \rightarrow 36 \rightarrow 27 \rightarrow 17 \rightarrow 11 \rightarrow 4 \rightarrow 2$

cylinders: $2 + 4 + 9 + 10 + 6 + 7 + 2 = 40$

total seek time = 40×20 ms = 800 ms

(c) Elevator Algorithm (initially moving upward)

$30 \rightarrow 32 \rightarrow 36 \rightarrow 27 \rightarrow 17 \rightarrow 11 \rightarrow 4 \rightarrow 2$

cylinders: $2 + 4 + 9 + 10 + 6 + 7 + 2 = 40$

total seek time = 40×20 ms = 800 ms

4. (44 points)

(a) (24 points) A RAID can fail if **four** or more of its drives crash within a short time interval. Suppose that the probability of one drive crashing in a given hour is P. What is the probability of a k-drive RAID failing in a given hour?

$$P_0 = (1-P)^k$$
$$P_1 = kP(1-P)^{k-1}$$
$$P_2 = \binom{k}{2}P^2(1-P)^{k-2}$$
$$P_3 = \binom{k}{3}P^3(1-P)^{k-3}$$

$$P(\text{RAID fails}) = 1 - [P_0 + P_1 + P_2 + P_3]$$
$$= 1 - \left[(1-P)^k + kP(1-P)^{k-1} + \binom{k}{2}P^2(1-P)^{k-2} + \binom{k}{3}P^3(1-P)^{k-3}\right]$$

(b) (20 points) Consider a computer system that has cache memory, main memory (RAM) and disk, and the operating system uses virtual memory. It takes 1 nsec to access a word from the cache, 8 nsec to access a word from the RAM, and 80 ms to access a word from the disk. If the cache hit rate is 70% and main memory hit rate (after a cache miss) is 90%, what is the average time to access a word?

Average access time:

$$T = (0.70)(1\text{ ns}) + (0.30)(0.90)(8\text{ ns}) + (0.30)(0.10)(8\times 10^7\text{ ns})$$
$$= 2{,}400{,}002.86\text{ nsec}$$

5. (36 points) (12 points each)
Given a set of five processes (All are 100 % CPU-bound).

Process	CPU-Time	Priorities	Arrival Time
P_1	7	3	0
P_2	3	5	0
P_3	5	1	0
P_4	4	4	0
P_5	1	2	0

Draw the corresponding Gantt chart (illustrating the execution of the processes) and calculate the average turnaround time for the schedule using the following three scheduling algorithms. Ignore process switching overhead.

(a) First-Come-First-Serve (run in order of P_1, P_2, P_3, P_4, P_5):

P_1	P_2	P_3	P_4	P_5

0 7 10 15 19 20

Average turnaround time = $\frac{7+10+15+19+20}{5} = 14.2$

(b) Priority Scheduling (with 5 being the highest priority):

P_2	P_4	P_1	P_5	P_3

0 3 7 14 15 20

Average turnaround time = $\frac{3+7+14+15+20}{5} = 11.8$

(c) Round Robin (with time quantum = 3):

P_1	P_2	P_3	P_4	P_5	P_1	P_3	P_4	P_1

0 3 6 9 12 13 16 18 19 20

Average turnaround time = $\frac{6+13+18+19+20}{5} = 15.2$

6. (24 points) (12 points each)

(a) Given a set of four processes (All are 100 % CPU-bound).

Process	CPU-Time	Arrival Time
P_1	10	0
P_2	2	0
P_3	5	0
P_4	3	0

Draw the Gantt chart and calculate the average turnaround time for the schedule using the Shortest-Job-First scheduling algorithm.

P_2	P_4	P_3	P_1

0 2 5 10 20

Average turnaround time = $\dfrac{2+5+10+20}{4} = 9.25$

(b) Given a set of four processes (All are 100 % CPU-bound).

Process	CPU-Time	Arrival Time
P_1	10	0
P_2	4	4
P_3	1	2
P_4	3	1

Draw the Gantt chart and calculate the average turnaround time for the schedule using the ***Shortest-Remaining-Time-First*** preemptive scheduling algorithm.

		✓	✓	✓	✓
P_1	P_4	P_3	P_4	P_2	P_1

0 1 2 3 5 9 18

Average turnaround time = $P_3 = 3-2 = 1$ $P_4 = 5-1 = 4$ $P_2 = 9-4 = 5$ $P_1 = 18-0 = 18$

$\dfrac{1+4+5+18}{4} = 7$