National University of Computer & Emerging Sciences, Karachi
FAST School of Computing
Final Examination Spring -2022
1st June 2022, 08:30 am – 11:30 am

Course Code: CS2006	Course Name: Operating Systems
Instructor Name: Dr. Ghufran Ahmed, Ms. Anaum Hamid, Ms. Tania Erum, Ms. Mubashra Fayyaz, and Ms. Safia Baloch	
Student Roll No:	Section No:

Instructions:

- Return the question paper.
- Students are not allowed to write anything on the question paper except roll number & section.
- Read each question completely before answering it. There are **6 questions on 4 pages**

Time Allowed: 180 minutes. **Max Marks: 100**

1. **Operating Systems Structure & Services:** ***(Estimated time = 15 min)*** **[Marks: 3+3 = 6]**

[CLO: 1]

a. Briefly describe how the operating system link and loads files for the compilation of a program with the help of a flow diagram.

b. Explain the problem that is solved by caching? If a cache can be made as large as the device for which it is caching (for instance, a cache as large as a disk), why not make it that large and eliminate the device?

2. **Process & CPU Scheduling:** ***(Estimated time = 25 min)*** **[Marks: (6+5+4+10 = 25]**

[CLO: 2]

a. Consider the following **Code segment:**

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

I. How many unique processes are created?
II. How many unique threads are created?
III. Illustrate your answer using a tree diagram.

b. List the actions taken by a kernel to context switch between processes.

c. Briefly describe the types of latencies that affect the performance of real-time systems.

d. Consider the set of processes in **Table 1**, with the length of the CPU burst given in *milliseconds*. Draw Gantt charts that illustrate the execution of these processes. Calculate **average turnaround time**, and **average waiting time** using the **Round Robin** algorithm with **Time Quantum = 2ms.**

Table 1		
Process	Arrival Time	Burst Time (ms)
A	0	5
B	1	3
C	2	1
D	3	2
E	4	3

3. **Synchronization:** ***(Estimated time = 30 min)*** **[Marks: 7+4+4 = 15]**

[CLO: 3]

a. Consider the following synchronization problem. A group of children is picking chocolates from a box that can hold up to N chocolates. A child that wants to eat chocolate picks one from the box to eat unless the box is empty. If a child finds the box to be empty, she wakes up the mother and waits until the mother refills the box with N chocolates. Unsynchronized code snippets for the child and mother threads are as shown below:

```
//Child
while (true)
getChocolateFromBox()
eat()

//Mother
while (true)
refillChocolateBox(N)
```

Modify the code of the mother and child threads by adding suitable synchronization such that a child invokes getChocolateFromBox() only if the box is non-empty, and the mother invokes refillChocolateBox(N) only if the box is fully empty. Solve this question by using only semaphores and no other synchronization primitive. The following variables have been declared for use in your solution.

```
int count = 0;
semaphore m, fullBox, emptyBox;
                //initial values of semaphores are not specified
```

I. Initial values of the semaphores
II. Code for the child thread
III. Code for the mother thread.

b. What are the limitations in Peterson's solution to improve the system performance of modern computer architectures?

c. Suppose two threads execute the following C code concurrently, accessing shared variables a, b, and c.

Calc. all possible values of c when two processes are concurrently running.

Initialization

```
int a = 4;
int b = 0;
int c = 0;
```

Thread 1

```
if (a < 0) {
    c = b - a;
} else {
    c = b + a;
}
```

Thread 2

```
b = 10;
a = -3;
```

4. **Memory Management:** ***(Estimated time = 30 min)*** **[Marks: 6+6+4+4+4 = 24]**

[CLO: 2]

a. Consider a logical address space of 1024 pages with 1024 words per page, mapped onto a physical memory of 128 frames.
   i. How many bits are required in the logical address?
   ii. How many bits are required in the physical address?
   iii. How many entries are in the page table if an inverted page table is used?

b. Given six memory partitions of 300 KB, 600 KB, 350 KB, 200 KB, 750 KB, and 125 KB (in order), how would the first-fit, best-fit, and worst-fit algorithms place processes of size 115 KB, 500 KB, 358 KB, 200 KB, and 375 KB (in order)?

c. Consider a process of size 83,456KB with 2KB of the page size. Calculate internal fragmentation.

d. In the notion of paging, elaborate the role of the Table Lookaside buffer with the help of a diagram.

e. Allocate the frames proportionally to the following processes from 256KB memory:
   i. P1 of size 20KB = _____
   ii. P2 of size 80 KB = _____
   iii. P3 of size 70 KB = _____
   iv. P4 of size 30 KB = _____

5. **Virtual Memory:** ***(Estimated time = 40 min)*** **[Marks: 4+5+6+5 = 20]**

[CLO: 2]

a. Under what circumstances do page faults occur? Describe the actions taken from the operating system when a page fault occurs. Illustrate your answer with a proper diagram.

b. Assume page time service time is 10ms in a computer system with an average memory access time is 20 ns. If a page fault is generated for every 10^6 memory accesses. What is the Effective Access Time for the memory?

c. Consider the following page reference string:

8 0 1 2 0 9 0 4 2 9 0 9 2 1 2 0 1 8 0 1

Assuming demand paging with three frames and all frames are initially empty. Further, the allocation of the first unique pages will cost one fault each. Calculate the **fault ratio** by following the listed below page replacement algorithms?

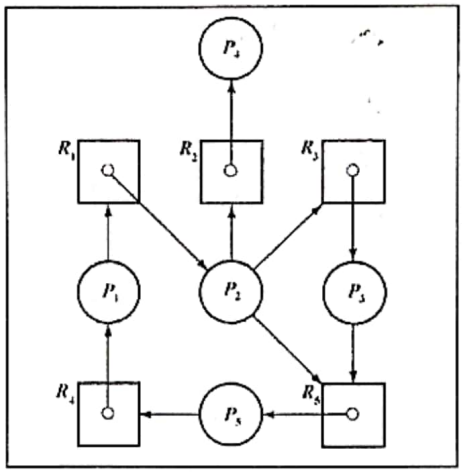
I. FIFO
II. LRU
III. Optimal Page replacement algorithm.

d. Assume there is an initial 1024 KB segment where memory is allocated using the Buddy system.
   I. Draw the tree illustrating how the following memory requests are allocated: request 240 bytes, request 120 bytes, request 60 bytes, request 130 bytes.
   II. How much memory is available after following releases of memory? release 240 bytes, release 60 bytes, release 120 bytes.

6. **Deadlock:** ***(Estimated time = 30 min)*** **[Marks: (3+3) +(2+2) = 10]**

[CLO: 3]

a. Consider the following resource allocation graph.



i. Convert it into a matrix representation (i.e., Allocation, Need, and Available).
ii. Find the safe sequence of the execution of all processes in the given system. Show allchanges in the Available (work array) after the termination of every process.

b. A system has three processes (P1, P2, P3) and three reusable resources (R1, R2, R3). There is one instance of R1, two instances of R2, and three instances of R3. P1 holds an R1 and an R3 and is requesting an R2. P2 holds an R3 and is requesting an R1 and an R2. P3 holds two R2 and an R3 and is requesting an R1.
   i. Describe the system by resource allocation graph.
   ii. Is the system in deadlock, and if so, which processes are involved showing them by Wait-for-graph?

**********Best of Luck**********