National University of Computer and Emerging Sciences

Islamabad Campus

0027

Operating Systems (CS2006)

Sessional-I Exam

Course Instructor(s):
Dr. Asif Malik, Ms. Safoora Qurban
Section(s): (CY / SE)

Total Time (Hrs): 1
Total Marks: 40
Total Questions: 8

Date: Sep 24, 2025

[redacted]

Roll No Course Section Student Sig.

Do not write below this line.

Attempt all the questions.

- **Questions 7 and 8** must be **answered** on this **question paper**. Answers written elsewhere will not be marked.
-
- **Bonus Marks:** Students who attempt **all questions (including all parts)** in the **correct sequence** on the specified answer sheets will be awarded **two bonus marks.**

[CLO 1: Describe services provided by the modern Operating Systems.]

Q1: Does the distinction between kernel mode and user mode function as a primary form of protection in operating systems? Justify your answer.

[3 marks]

[CLO 1: Describe services provided by the modern Operating Systems.]

Q2: Two operating systems provide the same services: process management, memory management, and file management. One exposes these services through high-level APIs, and the other only through low-level system calls. Do you think this difference affect application developers?

[3 marks]

[CLO 1: Describe services provided by the modern Operating Systems.]

Q3: Explain the purpose of the commands $gcc\ -c\ main.c$ and $./main$. What system services will run in backend while executing these?

[2 marks]

[CLO 1: Describe services provided by the modern Operating Systems.]

Q4: If the interrupt mechanism is completely disabled in a multiprogramming system, can the CPU still handle I/O operations such as disk reads or keyboard input efficiently? Justify your answer.

[3 marks]

[CLO 5: Understand the dead locks and memory management.]

Q5: Direct Memory access is used for high speed I/O devices in order to avoid increasing the CUP's execution load. **[3+3+3=9marks]**

a. How does the CPU interface with the device to coordinate the transfer?

b. How does the CPU know when the memory operations are complete?

c. The CPU is allowed to execute other programs while the DMA controller is transferring data? Does this process interfere with the execution of user programs if so, describe what forms of interference are caused?

[CLO 1: Describe services provided by the modern Operating Systems.]

Q6. If both monolithic and microkernels provide the same OS services, do you think the monolithic kernel is faster but less secure, while the microkernel is slower but more reliable? Explain your reasoning.

[4 marks]

[CLO 2: Implement solutions employing concepts of Processes and Threads.]

Q7: A system uses Circular Queue Scheduling with the following parameters:

- Time slice = 2 ms for even registration number and 2.5 ms for odd registration number
- Context switch time = 2.5 ms for even registration number and 2 ms for odd registration number (applied before every CPU slice, except at time 0)
- A periodic interrupt occurs every 10 ms, immediately preempting the currently running process.
- If multiple processes are ready, the scheduler follows RR order.

The system has four processes as shown below:

Process	Arrival Time (ms)	Burst Time (ms)
P1	0	5.5
P2	5	5.0
P3	10	3.5
P4	15	2.0

[5 marks]

Draw the Gantt chart/timeline of process execution up to completing all processes. Clearly mark context switch intervals, execution intervals, and interrupts.

[CLO 2: Implement solutions employing concepts of Processes and Threads.]

Q8: **[11 marks]**

Consider the following C Program. Assume there are no errors in the program and everything works. Dry run in the rough space provided. Then answer the questions below.

```
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <sys/wait.h>

int main()
{
  int i = 0;

  // OUTER LOOP (WHILE LOOP)
  // ------------------------------------------------
  // For ODD registration numbers → outer loop runs 3 times
  // For EVEN registration numbers → outer loop runs 2 times
  // ------------------------------------------------
  while (i < 3)   // <-- change "3" to "2" if registration number is EVEN
  {
    if (fork() == 0)
    {
      printf(" ! \n");
      printf("My process ID is %d (random)\n", getpid());
      printf("My parent process ID is %d (last 3 digits of reg. no.)\n", getppid());
      exit(0);
    }
    else
    {
      int j = 0;

      // INNER LOOP (DO-WHILE LOOP)
      // ------------------------------------------------
      // For ODD registration numbers → inner loop runs 2 times
      // For EVEN registration numbers → inner loop runs 3 times
      // ------------------------------------------------
      do
      {
        if (!fork() != 0)
        {
          printf(" £ \n");
          printf("My process ID is %d (random)\n", getpid());
          printf("My parent process ID is %d (last 3 digits of reg. no.)\n", getppid());
          exit(0);
        }
```

C

```
        j++;
    } while (j < 2);   // <-- change "2" to "3" if registration number is EVEN

    // WAIT LOOP (FOR LOOP)
    for (int k = 0; k < 2; k++)
      wait(NULL);
  }

  i++;
}

printf(" % \n");
printf("My process ID is %d (random)\n", getpid());
printf("My parent process ID is %d (last 3 digits of reg. no.)\n", getppid());

return 0;
}
```

a) How many times is "!" displayed? **(1 Mark)**	83
b) How many times is "£" displayed? **(1 Mark)**	1
c) How many times is "%" displayed? **(1 Mark)**	0
d) What is the total number of processes created (including the parent)? **(1 Mark)**	?
e) How many maximum zombie processes can be created in the above code? **(1 Mark)**	2

f) Suppose the following lines are removed:
`for (int k=0; k<2; k++)`
`wait(NULL);`
If the parent terminates immediately after completion, how many processes can become orphan? **(2 Marks)**
	?

g) If you run the program multiple times, will PIDs change? Justify your answer. **(1 Mark)**	

Yes when we run program multiple times so its PID change because there are many processess run in OS when we run again so it proces id is change.