National University of Computer and Emerging Sciences, Lahore Campus

	Course Name:	Operating Systems	Course Code:	CS2006
	Degree Program:	BSCS	Semester:	Spring 2023
	Exam Duration:	60 Minutes	Total Marks:	55
	Paper Date:	28-Feb-2023	Weight	15%
	Section:	ALL Sections except Section 6A	Page(s):	3
	Exam Type:	Mid 1 Exam		

Student : Name:_____ **Roll No.**_____ **Section:**_____

Instruction/Notes: Attempt all questions on the given answer sheet. Clearly mention your attempted question no. with the answers on the answer sheet. Avoid unnecessarily explanation.

CLOs	CLO-2	CLO-2	CLO-2	CLO-3	
Questions	Q-1	Q-2	Q-3	Q-4	Total
Total Marks	20	10	10	15	**55**
Marks Obtained					

Question 01: (20 points)

A) What output will be at **Line X** and **Line Y**? (10 points)

```
#include <sys/types.h>
#include<iostream>
#include<unistd.h>
using namespace std;

#define SIZE=5

int nums[SIZE] = {0,1,2,3,4};

int main() {
      int i;
      pid_t pid;
      pid = fork();
      if (pid == 0)
            {
```

```
            for(i=0; i<SIZE; i++) {
            nums[i]*=-i;
            cout<< "CHILD: "<<nums[i]<<endl; // LINE X
            }
      } else if (pid > 0) {
            wait(NULL);
            for(i=0; i<SIZE; i++)
            cout<< "Parent: "<<nums[i]<<endl; // LINE Y
            }
return 0;
}
```

B) Consider the following code: (5+5 = 10 points)

```
if ((fork() || fork()) && fork())
      fork();
printf ("Hello World");
```

How many times **"Hello World"** will be printed on the screen? Also draw the ***Process Tree*** of the given program.

Question 02: (10 points)
What is the context-switch? Explain it with the help of a diagram. Why is context-switch considered as overhead?

Question 03: (10 points)

- **A)** Write a C program that forks a child process that ultimately becomes a zombie process. This zombie process must remain in the system for at least 10 seconds. (4 points)
- **B)** Why do we need **init** process? (2 points)
- **C)** Answer **YES/NO** and provide a brief explanation. (4 points)

(a) Can two processes be concurrently executing the same program executable?

(b) Can two running processes share the complete process image in physical memory (not just parts of it)?

Question 04: (15 points)
The following processes are being scheduled using a **Preemptive, Round-Robin** scheduling algorithm.

Process	Arrival	Burst	Priority
P1	0	20	40
P2	25	25	30
P3	30	25	30
P4	60	15	35
P5	100	10	5
P6	105	10	10

Each process is assigned a numerical priority, with a higher number indicating a higher relative priority. In addition to the processes listed below, the system also has an idle task (which consumes no CPU resources and is identified as P_{idle}). This task has priority 0 and is scheduled whenever the system has no other available processes to run. The length of a time quantum is 20 units. If a process is preempted by a higher priority process, the preempted process is placed at the end of the queue.

1. Show the scheduling order of the processes using a **Gantt chart**.
2. What is the **Turnaround time** for each process?
3. What is the **Waiting time** for each process?