National University of Computer and Emerging Sciences, Lahore Campus

Course:	**Programming Fundamentals**	**Course Code:**	**CS-1002**
Program:	**BS (CS)**	**Semester:**	**Fall 2024**
Date:	**11 September 2024**	**Total Marks:**	**15**
Section:	**BCS- 1F**	**Name:**	
Exam:	**Quiz-1**	**Roll Number:**	
Time:	**20 Minutes**		

Instruction/Notes: Do not write anything on the back side of this sheet. (Penalty: -5)
Any form of plagiarism will result into negative penalty.

Q#1: Determine the output of the program. (Assume that there is no syntax error)

```
int main(){
   int x = 8, y = 10, z = 2, r = 3;
   if (++x % y-- / ++z % 2 == 0){
      cout << "inside if\n";
      r = r + x++ * --y / z;
      y = x + r / y;
   }
   else {
      cout << "inside else\n";
      r = r + x++ * --y / z;
      z = x + r / y;
   }
   cout << x << " " << y << " " << " " << z << " " << r;
   return 0;
}
```

Working (show the working of instructions that will execute in this program.**)**

Output:

Attempt the programming question on Answer sheet

Q#2: In the small tech company, TechSecure Inc., a recent incident has raised concerns among the developers about the security of their systems. During a routine update, a piece of malware was accidentally installed on several systems. To ensure that the malware did not compromise their systems, the developers decided to implement a security check. Julia, a software engineer at TechSecure Inc., was tasked with developing a tool to verify if the malware was still present on the systems. Julia designed a clever encryption system as part of her tool. The encryption system converts any non-zero input into a binary 1 and keeps zeros as 0. To confirm that the malware had not altered any system, Julia implemented an **even parity check**. This check involves ensuring that the number of zeros in a 3-bit binary number is even. If the count of zeros in the binary number is even, it suggests that the system is malware-free. Otherwise, the presence of malware might be indicated. Your task is to write a c++ program for this problem which first accept a three-bit number in an integer variable. Check the sample input for clarity.

Sample Input and output:	**Sample Input and output:**
Enter a three-bit binary number: 110 **The system is affected by the malware** //Reason: There is only a single 0's in the input which means the 0's are odd and system is affected by the malware.	Enter a three-bit binary number: 100 The system is safe //Reason: Since there are two 0's in the input which means the 0's are even and system is malware free.