National Institute of Technology Hamirpur
Himachal Pradesh-177 005, India
End Semester Theory Examination (November2023)
Department of Computer Science and Engineering

Course: B. Tech.   **Semester:** 3rd
Subject Code: CS-211   **Duration:** 3Hrs.
Subject Name: OOP's   **Max. Marks:** 50
Date: 22/11/2023   **Time:** 9:30am – 12:30pm
General Instructions (All the questions are compulsory)

Q.1 a) Why is it necessary to overload an operator? List at least four rules for Operator Overloading.

b) How polymorphism is accomplished in C++? Give a suitable example.   (5+5 = 10)

Q.2 a) What is ambiguity in multiple inheritance? Explain with example how the ambiguity can be resolved?

b) We want to calculate the total marks of each student of a class in Physics, Chemistry and Mathematics and the average marks of the class. The number of students in the class are entered by the user. Create a class named Marks with data members for roll number, name and marks. Create three other classes inheriting the Marks class, namely Physics, Chemistry and Mathematics, which are used to define marks in individual subject of each student. Roll number of each student will be generated automatically.   (5+5 = 10)

Q.3 Demonstrate hybrid inheritance with the help of suitable example. State the reason for

making a class virtual with the help of example.   (6)

Q4. Differentiate between function overloading and function overriding.   (4)

Q5. What is the concept of data hiding? Is it possible to access the private data members without using member functions? If yes, explain the procedure with programming example.   (6)

Q6. What is the difference between calling methods for constructors and destructors? Is it

possible to overload a destructor? Justify your answer.   (5)

Q.7 What is the need of streams and files? How can we read and write from a file? Explain with the help of suitable program.   (5)

Q.8 Give the output of the following:

a)
```cpp
#include <iostream>
using namespace std;
class A
{
int id;
public:
A (int i) { id = i; }
void print ()
{
cout << id << endl;
}
};
int main()
{
A a[2];
a[0].print();
a[1].print();
return 0;
}
```

b)
```cpp
#include <iostream>
using namespace std;
class A
{
int aid;
public:
A(int x)
{ aid = x; }
void print()
{ cout << "A::aid = " <<aid; }
};
class B
{
        int bid;
public:
        static A a;
        B (int i) { bid = i; }
};
A B::a(10);
int main()
{
B b(10);
b.a.print();
return 0;
}
```

(2+2=4)