International Islamic University Chittagong

Department of Computer Science & Engineering

B. Sc. in CSE Semester Final Examination, Spring 2019

Course Code: CSE 1221 Course Title: Computer Programming II

Total marks: ***50*** Time: ***2:30*** hours

[Answer any ***two*** questions from **Group-A** and any ***three*** questions from **Group-B;**
Separate answer script must be used for Group-A and Group-B;
Figures in the right hand margin indicate full marks.]

Group-A

1. a) Define operator overloading? Write the rules of operator overloading. 2

   b) Overload the (-) operator for the **coord** class using friend function. 3

   c) A friend function cannot be used to the overload the assignment operator (=). Explain why? 3

   d) Answer the following questions with explanation 2
      i) When an operator is overloaded, does it lose any of its original functionality?
      ii) Can the precedence of an overloaded operator be changed? Can the number of operand be altered?

2. a) Why and when to use inheritance? What happens to the public, private and protected members when they are inherited in public, private and protected mode? 3

   b) "Protected members are accessible in the class that defines them and in classes that inherit from that class." – Explain with a simple program example. 2

   c) What is the output of the following codes: 2

```
#include<iostream>
using namespace std;
class A{
public:
int a, b;
A(){
      cout<< "Constractor of base class." <<endl;
    }
    void add(inti = 1, int j = 3){
      a = i; b = j;
      cout<< "Result of a+b: " <<a+b<<endl;
    }
};
class B: public A{
int c, d;
public:
    void add(inti = 2, int j = 4){
      c = i; d = j; a = i; b = j;
      cout<< "Result of a+b+c+d: " <<a+b+c+d<<endl;
    }
};
intmain(){
    B obl;
obl.add();
}
```

   d) What problem associates with multiple inheritance and how is it solved by virtual inheritance? Explain with example. 3

3.a) "Bitwise **& operator** can be overloaded and it can only be overloaded to perform bitwise operations." – Do you agree with this statement? Why or why not? 2

b) Correct the following code segments: 3

(1)
```
#include<iostream>
using namespace std;
class A{
int a, b;
public:
A(inti, int j){
        a = i;
        b = j;
    }
    operator + (inti = 12){
        A temp;
temp.a = a + i;
temp.b = b + i;
    }
    void show(){
cout<< a << b <<endl;
    }
};
void main(){
    A ob1(10,5), ob2;
    ob2 = ob1 + 10;
    ob2.show();
}
```

(2)
```
#include<iostream>
using namespace std;
class A{
int a, b;
public:
    operator + (A ob){
        A temp;
temp.a = a + i;
temp.b = b + i;
        return temp;
    }
};
void main(){
    A ob1, ob2, ob3;
    ob1.a = 10; ob1.b = 12;
    ob2.a = 10; ob2.b = 12;
    ob3 = ob1 + ob2;
cout<< ob3.a << ob3.b <<endl;
}
```

c) Write short notes on the following terms. 2

1. Hybrid inheritance, 2. Super class, 3. Multilevel inheritance, 4. Nested class

d) Apply inheritanceto remove redundancyfrom the following code(including from the constructor function): 3

```
using namespace std;
    class MathTeacher {
    string profession;
    int age;
    public:
    MathTeacher (string s, inta){ profession = s; age = a; }
    void display()
        {
cout<< "My profession is: " << profession <<endl;
    cout<< "My age is: " << age <<endl;
    walk();
    talk();
    }
    void walk() { cout<< "I can walk." <<endl; }
    void talk() { cout<< "I can talk." <<endl; }
    void teachMaths() { cout<< "I can teach Maths." <<endl; }
    };
    class Footballer {
    string profession;
    int age;
    public:
        void Footballer (string s, inta){ profession = s; age = a; }
    void display()
        {
cout<< "My profession is: " << profession <<endl;
    cout<< "My age is: " << age <<endl;
    walk();
    talk();
    }
    void walk() { cout<< "I can walk." <<endl; }
    void talk() { cout<< "I can talk." <<endl; }
    void play() { cout<< "I Play football." <<endl; }
    };
```

Group-B

4.a) What is the relation between ios, istream, ostream, cin and cout? 2

b) Write a program in c++ to show the following output: 3

```
     ***Hello C++
****2.153625
     12.555468***
```

c) What is a stream? How streams are used for input and output in c++? 2

d) Design an inserter for a class named VECTOR which has two private integer members named x and y. 3

5. a) Write the output of the following program: 2

```
#include <iostream>
using namespace std;
class A
{
public:
    virtual void print(){
      cout<< "A class print function \n"; }
    void show(){
cout<< "A class show function \n";
        this ->print();
    }
};
class B: public A {
public:
    void print() {
cout<< "B class print function \n" ;
    }
    void show() {
cout<< "B class show function \n";
        this ->print();
    }
};
intmain() {
    A *b = new B;
    b ->show();
    return 0;
}
```

b) What do you mean by *Polymorphism* in OOP? Write the difference between early binding and late binding. Give examples of static and dynamic polymorphism. 3

c) What are the reasons to use pure virtual functions? Give an example program to explain the answer. 3

d) What is an abstract class? Why abstract class cannot be instantiated? 2

6.a) What is an exception? What are the advantages of using exception handling mechanism in a program? 3

b) Write a generic function, called min ( ), that returns the lesser of its two arguments. 3

c) What is generic function and generic class? 2

d) Rewrite the following code using generic function: 2

```
#include<iostream>
using namespace std;

int Add(int a, int b){
    return a+b;
}
double Add(double a, double b){
    return a+b;
}
string Add(string a, string b){
    return a+b;
}
void main(){
    string a = "Hello ";
    string b = "C++";
    cout << Add(a,b) << endl;
    cout << Add(4,2) << endl;
    cout << Add(4.5,2.5) << endl;
}
```

7.a) What will happen if an exception is thrown for which there is no corresponding catch statement? 2

b) Write a program to insert 20 items in a vector and all of them are 8 using an STL algorithm, after that remove 5 elements from that vector from the index 6. 2

c) Write a program containing a possible exception. Use a try block to throw it and a catch block to handle it properly. 2

d) What is STL? Define a container, an iterator and an algorithm as they relate to the STL. 4