International Islamic University Chittagong

Department of Computer Science & Engineering
B.Sc. in CSE, Mid Term Examination, Spring 2019
Course Code: CSE-1221 Course Title: Object Oriented Programming I
Total Marks: **30** Time: **90 minutes**
Answer any **three** of the following questions. Figures in the right hand margin indicate full marks.

1 (a) Define object and class? Write down the relations between Classes, Structures and unions. 1+3

(b) What are the difference between object oriented programming and structure programming? 3

(c) What is the output of the following program? 1

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

void demo()
{
    static int count = 0;
    cout << count << " ";

    count++;
}
int main()
{
    for (int i=0; i<5; i++){
        demo();
    }
    return 0;
}
```

(d) What are the differences between public and private class members? Write the differences between traditional and modern versions of C++. 1+1

2 (a) Define constructor and Parameterized Constructor. What is the output of the following program? 1+3

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

class Line {
   public:
      void setLength( double len );
      double getLength( void );
      Line();
      ~Line();
   private:
      double length;
};

Line::Line(void) {
   cout << "Object created" << endl;
}
Line::~Line(void) {
   cout << "Object is being deleted" << endl;
}
```

```
void Line::setLength( double len ) {
   length = len;
}
double Line::getLength( void ) {
   return length;
}

int main() {
   Line line;
   line.setLength(6.0);
   cout << "Length of line : " <<
line.getLength() <<endl;

   return 0;
}
```

(b) What is automatic inline function? When should we use inline and when shouldn't? Give an example. 2

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

Ⓐ

```
#include<iostream>
using namespace std;
int x = 10;
void fun(int x)
{
   cout << ::x << endl;
}
int main()
{
    fun(5);
    return 0;
}
```

Ⓑ

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

class Demo {
   public:
      int i;
      Demo(int i) {
         this->i = i;
      }
};

main() {
   Demo m(5);
   cout<<m.i;
}
```

(d) How to use private member variables of a class in a non-member function of that class? Explain with an example. 2

3 (a) Define function overloading. How a compiler can distinguish an overloaded function? Explain with example. 3

(b) Define default argument with example. Find error(s) of following codes, if any write the error message: 1+3

Ⓐ

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

class printData {
   public:
      void print(int i) {
        cout << i << endl;
      }
      void print(double  f) {
        cout << f << endl;
      }
      void print(char* c) {
        cout << c << endl;
      }
};

int main() {
   printData pd;
   pd.print(500.263);
   pd.print(5);
   pd.print("Hello C++");
   return 0;
}
```

Ⓑ

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

class amb{
public:
    int i, j;
    amb(int x, int y){ i = x; j = y; }
};
void func(amb a, amb &b){
    cout << a.i << endl;
}
void func(amb a, amb b){
    cout << b.j << endl;
}

int main(){
    amb ob1(12, 13), ob2(14, 15);
    func(ob1, ob2);
    return 0;
}
```

(c) Explain why you must be careful when returning objects from a function. Write the output of the following program: 1+2

```
#include<iostream>
using namespace std;
class MyClass{
public:
    MyClass(){ cout<<"Constructing"<<endl; }
    ~ MyClass(){ cout<<"Destructing"<<endl; }
};
```

```
MyClass aFunction(MyClass m){
     cout<<"In aFunction"<<endl;
     return m;
}
int main()
{
    cout<<"In Main"<<endl;
    MyClass ob;
    ob = aFunction(ob);
    cout<<"The End"<<endl;
    return 0;
}
```

4 (a) What is the purpose of using *new* operator? Write the output of the following code: 1+2

```
#include<iostream>
using namespace std;
int main(){
    int n = 12;
    int *p = new int[n];

    if(!p){
        cout << "Allocation failed" << endl;
    }
    for(int i = 0; i < n; i++){
        *(p+i) = i +1;
    }
    for(int i = 0; i < n; i++){
        cout << *(p+i) << " ";
    }
    delete[] p;
    return 0;
}
```

(b) What are the purposes of overloading a constructor function? The following program misses the constructor function(s). Complete the program so that each of the statements of the main function is supported and it runs error freely. You can use any variable name of your choice, if required. 1+3

```
#include<iostream>
using namespace std;
class OC
{    int i;
public:
    void display() { cout<<i<<endl; }
};
int main()
{
    OC A(4);
    OC B = A;
    OC C;
    cout<< "id of A : "; A.display();
    cout<< "id of B : "; B.display();
    cout<< "id of C : "; C.display();
    return 0;
}
```

(c) In which situations copy constructor is called? Write a simple program by using copy constructor. 1+2