Computer Programming

Sessional 2, Spring 2015

Date: 30-4-2015 **Marks: 55** **Time: 90 mins.**

Attempt all questions on answer sheets.

Question 1: Give brief answers to the following.

i. What is the output of the following program? **[5]**

```
class A
{
public:
   A() { cout << "In A's constructor" <<
endl; }
   ~A() { cout << "In A's destructor" <<
endl; }
};

class B : public A
{
public:
   B() { cout << "In B's constructor" <<
endl; }
   ~B() { cout << "In B's destructor" <<
endl; }
};
```

```
class C : public B
{
public:
   C() { cout << "In C's constructor"
<< endl; }
   ~C() { cout << "In C's destructor"
<< endl; }
};

int main()
{
   C x1;
   C *x2 = new C;
}
```

ii. Given the following declarations: **[3]**

```
char     c = 'A';
char   * p = &c;
char ** p2 = &p;
```

Examine each of the following expressions and write its resultant type (i.e. int, etc) or ERROR.

&p2	*p2	p2 + 1

iii. Write the correct declaration for a function foo that takes pointer to a float, a pointer to a pointer to a char and returns a constant pointer to a pointer to an integer. **[2]**

iv. Having a pointer as a class data member can lead to some undesirable side effects. What should add to a class to avoid these side effects? **[3]**

v. What is an initializer list? Describe two of its uses by giving examples. **[2]**

vi. What is the difference between the keywords struct and class. **[2]**

vii. Why can't we use the this pointer in nonmember functions? **[1]**

viii. Which member functions of a class are created automatically by the compiler if they are not included (by the programmer) in the class definition? **[4]**

ix. How many times is the copy constructor called in the following code: **[3]**

```
Widget f(Widget u)
{
   Widget v(u);
   Widget w = v;
   return w;
}
```

```
int main()
{
   Widget x;
   Widget y = f(f(x));
   return 0;
}
```

Question 2: The following code on the left side lists a driver for a class `Set`. When executed, the code prints the output given on the right side. Implement the class `Set` which will allow the code above to run without any errors. **[30]**

```
int capacity = 10;
Set set1(capacity);
cout << "set1 = "<< set1 << endl;

set1 = set1 + 2; // Add an element to the set
set1 = set1 + 5;
cout << "set1 = "<< set1 << endl;

int arr[] = {1,2,3};
Set set2(arr,3), set3(arr,3);
cout << "set2 = "<< set2 << endl;

set2 = set2 + set1; // Union operation
cout << "set2 = "<< set2 << endl;

set3 = set2;
cout << "set3 = "<< set3 << endl;

set2 = set3 - set1; // Set difference

if ( set2 == set3 )
    cout << set2 << " == " << set3 << endl;
else
    cout << set2 << " != " << set3 << endl;

set2 = 2 + set2; // Add an element to set
cout << "set2 = "<< set2 << endl;
```

```
set1 = [ ]

set1 = [ 2 5 ]

set2 = [ 1 2 3 ]

set2 = [ 1 2 3 5 ]

set3 = [ 1 2 3 5 ]

[ 1 3 ] != [ 1 2 3 5 ]

set2 = [ 2 1 3 ]
```

Best of Luck!