National University of Computer and Emerging Sciences, Lahore Campus

Course:	Computer Programming	Course Code:	CS103
Program:	BS(Computer Science)	Semester:	Fall 2018
Duration:	60 Minutes	Total Marks:	45
Paper Date:	16-Nov-2018	Weight	15
Section:	All	Page(s):	1
Exam:	Midterm II	Roll No: Section	

Instructions: - Questions during exam are not allowed. Take reasonable assumptions where needed.

Question 1 [40 Marks]: Complete the definition of the class given below such that the main program runs successfully. Make sure that your program doesn't consume extra memory space and it should not leak any memory.

```
class Set{
private:
     int* elements; //To save elements of a Set
     int size;      //Total number of elements in a Set
};

void main()
{
     int arr1[] = {10,20,30,40};
     Set s1(arr1 , 4);//Creates a Set with elements of arr1 and size
= 4

     int arr2[] = {5,15,55};
     Set s2(arr2 , 3);//Creates a Set with elements of arr2 and size
= 3

     cout<<"Set 1:\t"<<s1<<"Set 2:\t"<<s2;   //Prints Set 1:
{10,20,30,40}                                          //Set 2:
{5,15,55}

     Set s3;
     s3 = s1+s2; //Takes sorted Union of s1 and s2

     cout<<"Set 3:\t"<<s3;    //Prints Set 3: {5,10,15,20,30,40,55}

     int arr3[] = {1,2,100,-1};    //-1 is delimiter
     s3 = arr3 + s1;           //Returns sorted Union
     cout<<"Set 3:\t"<<s3;    //Prints Set 3: {1,2,10,20,30,40,100}

     cout<<s1--;          //Prints {10,20,30,40}
     cout<<s1;      //Prints {9,19,29,39}
}
```

Question 2 [5 Marks]: Write exact sequence of function calls for following code:

s3 = arr1 + s2 + arr3;

Question 1 Solution:

```
#include<iostream>
using namespace std;
class Set
{
	friend ostream& operator<<(ostream& out, Set& rhs);
	friend Set operator+(int* arr , Set& rhs);
private:
	int* elements;		//To save elements of a Set
	int size;		//Total number of elements in a Set
	static int* MergeSortedArrays(int*& arr1, int*& arr2, int& size1, int& size2)
	{
		int* result_elements = new int[size1+size2];

		int i, j, k;
		i = j = k = 0;
		while(i<size1 && j<size2)
		{
			if(arr1[i] < arr2[j])
			{
				result_elements[k] = arr1[i];
				i++;
			}
			else
			{
				result_elements[k] = arr2[j];
				j++;
			}
			k++;
		}
		while (i<size1)
		{
			result_elements[k] = arr1[i];
			i++;
			k++;
		}
		while (j<size2)
		{
			result_elements[k] = arr2[j];
			j++;
			k++;
		}
		return result_elements;
```

```
	}
public:
	Set(int* data = 0 , int s = 0)
	{
		size = s;
		if(size > 0)
		{
			elements = new int[size];
			for(int i=0; i<size ; i++)
			{
				elements[i] = data[i];
			}
		}
		else if(data == 0)
			elements = 0;
		else
		{
			int count = 0;
			while (data[count] != -1)
			{
				count++;
			}
			size = count;
			elements = new int[size];
			for(int i=0; i<size ; i++)
			{
				elements[i] = data[i];
			}
		}
	}

	Set operator+(Set& rhs)
	{
		Set result;
		result.size = size + rhs.size;
		result.elements = MergeSortedArrays(elements, rhs.elements, size,
rhs.size);
		return result;
	}
	Set(Set& rhs)
	{
		size = rhs.size;
		elements = new int[size];
		for(int i=0; i<size ; i++)
		{
			elements[i] = rhs.elements[i];
		}
```

```
	}
	~Set()
	{
		if(elements != 0)
			delete[] elements;
	}
	Set& operator=(Set& rhs)
	{
		if(this != &rhs)
		{
			if(size != rhs.size)
			{
				if(elements != 0)
					delete[] elements;
				size = rhs.size;
				elements = new int[size];
			}
			for(int i=0; i<size ; i++)
			{
				elements[i] = rhs.elements[i];
			}
		}
		return *this;
	}
	Set operator--(int)
	{
		Set prevState(*this);
		for(int i=0; i<size ; i++)
			elements[i]--;
		return prevState;
	}
};
ostream& operator<<(ostream& out, Set& rhs)
{
	if(rhs.elements != 0)
	{
		out<<"{ ";
		for(int i=0; i<rhs.size ; i++)
		{
			out<<rhs.elements[i]<<" , ";
		}
		out<<"}\n";
	}
	return out;
}
Set operator+(int* arr , Set& rhs)
{
```

```
        Set result(arr);
        result = result + rhs;
        return result;
}
```

Question 2 Solution:

1- Operator(int*, Set)
2- Operator+(int*)
3- Operator=