Lab 06 Dynamic Array(OOP)

Topic(s): Default, Non-Parameterized, Parameterized & Copy Constructors. Constant Functions, Dynamic Allocation of Objects, Object Passing/Return in Functions.

IMPORTANT INSTRUCTIONS:

Please keep in mind the following points while coding. Violating any of these will result in credit deduction.

- There should be no memory leakage in your class. There should be no dangling pointers.
- Make functions, objects, variables as constant wherever possible.
- Create Default, Parameterized and Copy Constructor whether mentioned or not.
- Create Setters and Getters for all attributes.
- Follow the appropriate naming conventions as explained in class. Submit your files following the submission format explained in class.

Create a class Dynamic Array with the following functions and member variables.

```
class DynamicArray
{
    int size;
    int* ptr;
public:
    DynamicArray();
    DynamicArray(const DynamicArray & other);
    DynamicArray(const int size,const int * arr);
    void Print() const ;
    void Add(const int value);
    void Subtract(const int value);
    void Multiply(const int value);
    void Divide(const int value);
    int GetIndexOf(const int value) const;
    int GetValueAtIndex(const int value)const;
    void InsertValueAtEnd(const int value);
    void DeleteValue(const int value);
    void CopyArray(const int size,const int * arr);
    void CopyObject(const DynamicArray & other);
    void SortAscending ();
    void SortDescending ();
    DynamicArray MergeArray(const DynamicArray & other);
    DynamicArray MergeArrayAndSort(const DynamicArray & other);

    ~DynamicArray();
};
```

The Source.cpp will look like this

```
#include<iostream>
using namespace std;
#include "DynamicArray.h"
int main()
{
	DynamicArray obj1;
	obj1.InsertValueAtEnd(1);
	obj1.InsertValueAtEnd(13);
	obj1.InsertValueAtEnd(134);
	obj1.InsertValueAtEnd(11);
	obj1.Print(); // 1,13,134,11
	DynamicArray obj2(obj1),obj3 = obj1;
	obj2.Print(); // 1,13,134,11
	obj3.Print(); // 1,13,134,11
	obj2.InsertValueAtEnd(200);
	obj1.InsertValueAtEnd(2);
	obj3.DeleteValue(13);
	obj1.Print(); // 1,13,134,11,2
	obj2.Print(); // 1,13,134,11,200
	obj3.Print(); // 1,134,11
	obj1.Add(1);
	obj2.Subtract(3);
	obj3.Multiply(2);
	obj1.Print(); // 2,14,135,12,3
	obj2.Print(); // -2,10,131,8,197
	obj3.Print(); // 2,268,22
	obj1.Subtract(1);
	obj2.Add(3);
	obj1.Print(); // 1,13,134,11,2
	obj2.Print(); // 1,13,134,11,200
	DynamicArray obj4 = obj1.MergeArray(obj2);
	obj4.Print(); // 1,13,134,11,2,200

	//// Dynamically declare two DynamicArray objects and call functions via Arrow Pointer.
	////
	////
	////
	return 0;
}
```

Test all the functions like in the example given above

If there is any bug in your code or there is an error, try to solve it yourself and if that error or bug is still there after spending a few minutes on it then call us, we shall help you.

Remember "Debugger" is your best friend in this course and in your life too.