Object Oriented Programming – Fall 22

(BS-CS-F22)

Lab-5 (IntArray)

Lab Instructor: Maa'm Sanam Ahmad

Total marks=66

Instructions:

- **Indent your code properly.**
- **Use meaningful variable and function names.**
- **Use the camelCase notation.**
- **Use meaningful prompt lines/labels for all input/output.**
- **Do NOT use any GLOBAL variable(s). However, global named constants may be used.**
- **This is an individual lab, you are strictly NOT allowed to discuss your solution with fellow colleagues, even not allowed to ask how is he/she is doing, it may result in negative marking. You can ONLY discuss with your TAs or with me. • Anyone caught in an act of plagiarism would be awarded an "F" grade in this Lab.**

Do Validations on inputs where required otherwise 1 mark will be deducted for every wrong validation.

TASK-1:

Task Description:

You are tasked with creating a IntArray class.

```
class IntArray {
      int* arr;// pointer to dynamic 1-D array
      int size;
}
```

Data Members:

- size (an integer for size of array)
- arr (a pointer to the dynamically allocated array)

Constructors: *2.5+2.5+5*

- Implement a **Default constructor** for `IntArray` class that assigns 0 to size and nullptr to the arr

- Implement an **Overloaded constructor** for `IntArray` class that accepts size of array colms allocates array
- Implement Copy Constructor of the IntArray that should make deep copy for the instance that is being created.

Member Functions:

- Implement the getter and setter functions for each member variable of the `IntArray` class.The setter function for arr data member should index no as well as value to set for that index. Like **arr[a] =value** . **(marks =5)**
- Implement a member function `void display()const` of the `IntArray` class which should neatly display whole array on the screen. **(marks =2)**
- Implement Destructor of the IntArray class that should deallocate all the memory safely and there should not be any issue of **memory leak as well as dandling pointer**. **(marks =3)**
- Implement a public member function of the array class named **IntArray addArrays (const IntArray obj) const** that should take a array instance passed by value and add that array to the calling array and return new array that will be result of addition of two arrays. **(marks=2)**
  Note *Two arrays can be added only when their size is same*
- Implement a public member function named **IntArray subArray(const IntArray) const .**This array should be subtracted from the calling array and return new array that will result from subtraction. **(marks=2)**
  Note *Two arrays can be subtracted only when their size is same*
- Implement a public member function named **IntArray mulArray(const IntArray) const .**This array should be multiplied with the calling array and return the resultant array. **(marks=2)**
  Note *Two arrays can be multiplied only when their size is same*
- Implement a public member function named **bool palindrome() const** . This function should tell whether array is palindrome or not. **(marks=10)**

1	2	3	4	3	2	1

This is palindrome.

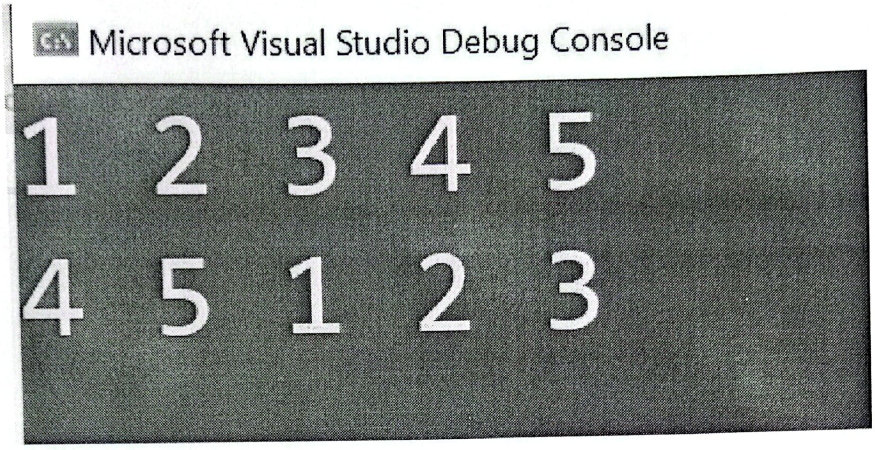
- Implement a public member function named **IntArray reverse () const .** This function should returned new array that should be reversed version of the calling array. **(marks=10)**

1	2	3	4	5	6	7

7	6	5	4	3	2	1

- Implement a public member function named **IntArray cyclicRotate(int n) const**

- Here n is number of rotations. Following array is rotated two times. **(marks=10)**



Menu based Driver: **(marks=10) if main is not implemented correctly or not running then 0 will be awarded in this section.**

- Please write a menu-based driver program that should first take size of array and create an IntArray instance,then input for each index. After that verify correctness by displaying array. In the same way do for the second Array and verify all functions
- The return type of each function is also a IntArray so returned Array should also be displayed.
  For example
  IntArray a(5), b(5) // constructor will be called
  // call setter in loop and initialize both matrices
  IntArray result=a.addArray(b); // here the copy constructor will be called for result array.
  result.display() // verify that the result of addition is correct

Very important information: Carefully deallocate memory in the destructor as well as array of students in main function make pointer nullptr avoiding pointers to become dangling pointers.

In case of a pointer

```
If(ptr!=nullpr)
  {
delete ptr; ptr=nullptr;
}
// in case of dynamically allocated array
If(ptr!=nullptr)
{
 delete [] ptr;
ptr=nullptr; }
```