Object Oriented Programming – Fall 22

(BS-CS-F22)

Lab-7

Lab Instructor: Mam Sanam Ahmad

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.**
- ❖ **You should have implemented driver program for each task**

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

TASK-1: Total Marks (20)

Implement a class *ObjectTracker* that can track how many objects of a class are present in memory. It should increment the object counter when a new instance of the object tracker is created. It should decrement the counter when any object is removed from the memory.

Implement all constructors, destructors, and necessary accessors and mutator functions. The user should be able to get the number of active objects of the class. In order to verify creation of objects you should dynamically allocate a random sized array; the size of the array will be a random number and this will be generated by the program. After allocation of array every object at each index of the array should be able to tell us that how many objects of the class are present. Intentionally deallocated memory for an instance and again tell how many objects are present there in memory of this class.

```
srand(time(NULL)); // #include <cstdlib>
int size = rand() % 20 + 1; // this will generate random numbers in range 1 and
20
ObjectCounter **arr = new ObjectCounter*[size];
```

TASK-2: Total Marks (10)

Develop a C++ program to find the area of a rectangle by converting the member of a class square (data member: side) which is a friend class of rectangle (data member: height, width). Declare Rectangle as a friend of Square so that Rectangle member functions could have access to the private member of square.