Lab 4

Course: CSE 165
Section: 02L & 03L
Due: Sunday, October 10, at 11:59 pm

All the exercises below are selected from the textbook: Thinking in C++ (volume 1).

1. [Exercise-27 on Page 658] This exercise creates the design pattern called proxy. Start with a base class Subject and give it three functions: f(), g(), and h(). Now inherit a class Proxy and two classes Implementation1and Implementation2from Subject. Proxy should contain a pointer to a Subject, and all the member functions for Proxy should just turn around and make the same calls through the Subject pointer. The Proxy constructor takes a pointer to a Subject that is installed in the Proxy (usually by the constructor). In main(), create two different Proxy objects that use the two different implementations. Now modify Proxy so that you can dynamically change implementations. [30 points]

2. [Exercise-4 on Page 505] Write a function that takes a pointer argument, modifies what the pointer points to, and then returns the destination of the pointer as a reference. [15 points]

3. [Exercise-25 on Page 508] Create a class containing an array of int. Index through this array using a pointer to member. [20 points]

4. [Exercise-1 on Page 349] Create a Text class that contains a string object to hold the text of a file. Give it two constructors: a default constructor and a constructor that takes a string argument that is the name of the file to open. When the second constructor is used, open the file and read the contents into the string member object. Add a member function contents() to return the strings so (for example) it can be printed. In main(), open a file using Text and print the contents. [25 points]

5. [Exercise-4 on Page 717] Modify C14:Combined.cpp so that f() is virtual in the base class. Change main() to perform an upcast and a virtual call. [10 points]

Requirements:

* Usage of spaces, blank lines, indention, and comments for readability.

* Descriptive names of variables, functions, structs, classes, and objects (if any).

* Appropriate usage of structs, classes, and objects (if any).

Penalties:

* 10-point deduction per day late until zero.

* Zero if you have possession of a copy of online solutions or work done by someone else.