Midterm Makeup – Take Home Examination
Due Wednesday, 08 May 2019

To complete this makeup successfully you need to provide a Word or PDF document that contains two sets of things:

First, go over each of the questions on your examination and describe, in that document, what, if anything was wrong with your answer. If you received less than a score of 1 for a question, you should assume there is something wrong. This is intended to have you do a critical examination of each of your answers. You may refer to my solutions if you wish, but will get no credit for repeating any part of those answers.

The second thing you need to do is to answer, as completely as you can, all ten questions below. For questions like the first, after you finish implementing the code, please copy the source code into your word document.

I cannot accept your makeups after Wednesday, May 8th, 2019. Please upload the document to the college server, CSE687 folder.

1. Implement a thread-safe vector that provides the same functionality as the std::vector, but supports safe read and write access by multiple concurrent threads.

2. Describe where you have used the five main principles discussed in class, SRP, LSP, OCP, DIP, and ISP, in your design of Project #4.

3. Please provide the definition of a Message dispatcher used to process messages received by the Client GUI from the Code Publisher server of Project #4. Provide at least one lambda, bound to a message command by the dispatcher.

4. What is a constructor initialization sequence and why is it used? Give an example of demo code from the college server that uses one or more initialization sequences.

5. Why has the Cpp11-BlockingQueue made its copy operations =delete? You can find that code in Handouts/Repository/Cpp/Cpp11-BlockingQueue.

6. Given the code fragments below, where the ellipsis (…) represents code that has not been disclosed to you:

    class X { … }; class Y { public: explicit Y(const X& x); … };

    what can you say about the compilation and execution of each of the following statements? Describe each of the operations that occur as this code executes.

    Y func(Y y) { … }
    X x;
    Y y = func(Y(x));

7. Suppose that objects of a class are required to correctly carry out copy and assignment operations. Support or refute the claim that the class must implement a copy constructor and assignment operator. You must consider all the cases (what are they?).

8. Use the std::vector class to hold a collection of strings. Use algorithm(s) from the STL and a lambda to concatenate the strings.

9. What is the purpose of defining both an interface and an abstract class in an inheritance hierarcy?

10. Given the declaration:

    void X::mf(Y &y);

    How would you change the declaration to declare that the operation of this function will not change the state of the object invoking it? What can you do with the re-declared function that you could not do in its original form?