**Module Assignment Summary**

[Module Assignment Summary](https://faytechcc.blackboard.com/webapps/blackboard/content/listContent.jsp?course_id=_33824_1&content_id=_5842793_1)

**Assignments**

* CH9 : Streams (due 11/29)
* CH10: Inheritance (due 12/6)
* M5HW1: Object-Oriented Design 1 (due 11/29)
* M5HW2: Object-Oriented Design 2 (due 12/6)

**Optional (Bonus) Assignments** (will replace your lowest score in any one category)

* CH11BONUS: Recursion
* CH14BONUS: Containers

### [CH9](https://faytechcc.blackboard.com/webapps/assignment/uploadAssignment?content_id=_5842796_1&course_id=_33824_1&group_id=&mode=view)

**CH9 - due 11/29**

**Chapter 9**covers the iostream functionality of C++ and (in general) how streams function.

For this assignment, complete the Participation Exercises within Chapter 9.

### [CH10](https://faytechcc.blackboard.com/webapps/assignment/uploadAssignment?content_id=_5842797_1&course_id=_33824_1&group_id=&mode=view)

**CH10 - due 12/6**

**Chapter 10** extends our work on classes and objects by further examining topics such as inheritance.

For this assignment, complete the Participation Exercises within Chapter 10.

### [Bonus Submission](https://faytechcc.blackboard.com/webapps/assignment/uploadAssignment?content_id=_5842798_1&course_id=_33824_1&group_id=&mode=view)

You can complete any of the remaining chapters Participation Exercises, and screenshot the results, below, in order to recieve bonus credit (replacing your lowest score in any other category to date).

I recommend Ch 11 (Recursion) and Ch 14 (Containers) as they are generally useful -- but the others are also fine.

### [M5HW1](https://faytechcc.blackboard.com/webapps/assignment/uploadAssignment?content_id=_6155443_1&course_id=_33824_1&group_id=&mode=view)

M6HW1 - Online Shopping Cart (Part1)

due 11/29.

This assignment sends us back to work with Chapter 7 (classes and objects) to assemble the back-end code that would be needed for an online shopping cart.

A common "design pattern" for complex programs is Model View Controller. To briefly summarize MVC, the concept states that the code used for the data, the "business logic", should be distinct from the code for the user interface. As programs add more and more complex user interfaces (such as web-based front ends), it's all the more important to keep this separation.

In other words, this program wouldn't represent a full online storefront, but it would provide the logic to drive that storefront.

For M6HW1, complete 7.26 (Online Shopping Cart, Part 1). You do **not** need to use the online web-based lab grader -- you can submit your program as usual.

### [M6HW2](https://faytechcc.blackboard.com/webapps/assignment/uploadAssignment?content_id=_6155444_1&course_id=_33824_1&group_id=&mode=view)

M5HW2 - Online Shopping Cart (part 2)

For this assignment (due 12/6) you will implement the second half of the program, as listed in Section 7.27 of Zybooks.

Combined, these two exercises show your ability to implement an object-oriented design.

### [M5BONUS](https://faytechcc.blackboard.com/webapps/assignment/uploadAssignment?content_id=_6118936_1&course_id=_33824_1&group_id=&mode=view)

**M5BONUS - due 12/6**

(Due to the complexity of this assignment, I have converted it to **optional, extra credit.** A new M5HW1 and M5HW2 are now posted.)

We've spent this module looking at pointers, which are commonly used to store references to other objects.

For this assignment, you should rework the Text Adventure program from the previous module to use pointers.

For example, a Room object may contain exits which are Room\* (pointer to other rooms). For a player to move from one room to another, the program should be able to change the "current player location" from the current room to the new room.

**Bronze (80/100)** - Rework your text adventure program so that Room pointers are used to track what room the player is in, and how exits work. You will need to properly reference / dereference pointers as needed (as discussed in the reading).

**Silver (90/100) -**Expand the existing map of 5 rooms to contain 8 or more rooms. Each should have a name, a brief description, and at least one exit to/from another room.

**Gold (100/100) -**Make any necessary changes to turn your "miscellaneous collection of rooms" into a simple game. This means that the game should have a simple introduction (in text) explaining the premise, as well as some win condition (and optionally a loss condition).

You can use any sort of simple "story" for this -- haunted house, scavenger hunt, exploring an alien spaceship -- but gameplay-wise it is basically a "walking simulator" where the gameplay consists of moving between rooms, looking around for clues, and then finally reaching some ending location.

You may wish to add items, puzzles, locked doors, and the like, and you are free to make any changes necessary to do so.