Version Timeline

1. convertArrayToSTL\_v1
   1. converts string arrays (names and p2names) to unordered\_set containers
2. convertArrayToSTL\_v2a
   1. Converted unordered\_map to map, so I could sort on topPlayer container.
   2. Changed map topPlayer pair to (int, string) , so the container sorted the scores in descending order.
      1. map<int, string> topPlayer(pair(hiScore ,“name”)
   3. Problem: Couldn’t figure out how to find a person’s name when name was the value.
3. convertArrayToSTL\_v2b
   1. Changed map topPlayer pair to (string, int), so I could find a name
      1. map<string, int> topPlayer(pair(“name”, hiScore)
4. battleship\_v1\_BROKEN
   1. Version 1: Is based off my CIS 5 project 2 version 6 code.
   2. Added convertArrayToSTL\_v2b and it broke it.
5. battleship\_v2
   1. Version 2: Is based off my CIS 5 project 2 version 6 code.
   2. Ran the original code to make sure the game worked. Then I commented everything out and added convertArrayToSTL\_v2b. It works.
   3. Converted set2Upper() to use mutating algorithm. Updated getName() with set2Upper().
   4. Converted scores array to a set and then created a list with a couple of values, and then copied the set into the list using copy() and insert\_iterator.
6. convertArrayToSTL\_v3
   1. Converted scores array to a set and then created a list with a couple of values, and then copied the set into the list using copy() and insert\_iterator.
   2. Copied the updated setTopPlyrs() to battleship\_v2.
7. battleship\_v3
   1. Created shuffleThis(). It works on arrays, but not on a set container.
   2. so I created getMin() and used std::min() to find the shortest name between 2 strings and then set player 2’s name in main().
8. convertArrayToSTL\_v4
   1. Converted guess char array to a stack container.
   2. Counted num ships. You need at least 3 ships in a container to eventually get a winner.
   3. Copied working version into battleship\_v4
9. battleship\_v4
   1. Created a menu that allows user to view CIS17-C concepts applied to the game or to play Battleship but with char arrays.
   2. Concept’s function contains all the project requirements in one location.
10. battleship\_v5
    1. Implemented queue with number of ships each player has.
    2. queue<string> ships, setShips(), prntQue()
    3. Added constructor() to prepare to implement Classes
    4. Changed pushName() to setSet() and reduced some redundant code
    5. setStack() initializes a stack with a list
       1. Can I initialize stack with linked list and then access the link’s pointer member? NO, you can’t.
11. convertArrayToSTL\_v5
    * 1. Deleted everything and copied all the STL functions from battleship\_v4 to this program because I want to add these cis17c concepts to my one of my games that is already in classes and is completely working.
      2. The problem with battleship\_v5 is that the game functions aren’t in classes in this version because I based it off my cis5\_project\_2\_battleship.
      3. setStack() sets stack with list
      4. prntStckFIFO()
      5. prntStckRev()
      6. Tried reverse stack with reverse() and recursive insert\_at\_bottom() but couldn’t get it to work.
12. project1\_convertArrays\_to\_STL\_v6
    1. Looking highest score in map<>topPlyr
    2. Passed map reference to topPlyrs()
    3. Problem: Couldn’t get ostream to work in getName() with a list or a vector or with stream.
13. project1\_convertArrays\_to\_STL\_v7
    1. Added Game class
       1. Constructor:
          1. Initialized all the STL concepts in the constructor and start()
14. battleship\_v6
    * 1. Copied my cis17b\_yahtzee\_v29 because I need the game to be in classes.

To do:

* setSet(): make fnames.insert( pair(index,name));
* Add classes. Use Yahtzee\_v29 and cis17a\_project\_2 as guides.
  + Player, Game, Board
  + Add structures for Board
    - 2 game boards, 2 guess boards
* Changed guess container from a stack to map
* iterators
* Change set<float>s to a: array, map, or linked list

Before submitting:

* Uncomment cin inside of getName()

After submission:

* Clean up code that had to be included to meet project 1 requirements.