Joseph Lewis

Jaycob Callies

10/16/2018

PA 4 report

1. **Problem Statement**

The object of this programming assignment was to create a board game that would have a varying size and players could be added and removed from the game during game play. Also, the number of players is less than the x and y dimensions of the board.

1. **Experimental Setup**

The CPU we used for this experiment is a “Intel(R)Core(TM)i7-4710HQ”, which is a 64-bit operating system. The clock speed used for testing was 2.50 GHz. The RAM use for this experiment had a memory value of 8 gigabytes with a speed of 1600 MHz. All the testing in this experiment was conducted in the EECS SSH server, which involved the Unix environment using g++.

**3.0 Algorithm Design**

To solve this problem, we first had to create a class named player. This class described a player’s location and id number. The runtime complexity for all the functions in the player class where O(1). The player class had the functions setid, setx, sety, setAll, getid, getx, gety, print, copy constructor, constructor, and a destructor. For all of these functions the space complexity is constant Θ(1) since no function allows for a variable amount of data to be used.

After we implemented the player class, we created another class named Board. The class was used to describe the game board and implement functions needed to insert, remove, print, and move players. We used a BST map as the required data structure, and stored the players in the map with a key corresponding to their ID value, and the data being the player object itself. The functions for default constructor, parameterized constructor, destructor, getM, getN, setM, and setN all have constant Θ(1) time complexity as they do not traverse a through the map, Remove and Find, have Θ(log(n)) time complexity since they only need to read the ID which is the key, while Insert, MoveTo, PrintByID have Θ(n) time complexity since they need to compare x and y values which are not the key. For all of these functions the memory complexity is constant Θ(1) since no function allows for a variable amount of data to be used.

**4.0 Test Results**

The following result is in the file ouput.txt. The final result is:

ERROR! ID is already being used.

ERROR! Location is already occupied.

Removed player 7 from the board.

Player List:

Player 1 is located at (3, 2) (x, y).

Player 2 is located at (4, 3) (x, y).

Player 3 is located at (4, 6) (x, y).

Player 4 is located at (9, 3) (x, y).

Player 5 is located at (6, 3) (x, y).

Player 6 is located at (8, 2) (x, y).

Player 8 is located at (9, 8) (x, y).

Removed player 8 from the board.

Player List:

Player 1 is located at (3, 2) (x, y).

Player 2 is located at (4, 3) (x, y).

Player 3 is located at (4, 6) (x, y).

Player 4 is located at (9, 8) (x, y).

Player 5 is located at (6, 3) (x, y).

Player 6 is located at (8, 2) (x, y).

Removed player 5 from the board.

Player List:

Player 1 is located at (3, 2) (x, y).

Player 2 is located at (4, 3) (x, y).

Player 3 is located at (4, 6) (x, y).

Player 4 is located at (6, 3) (x, y).

Player 6 is located at (8, 2) (x, y).

Player List:

Player 1 is located at (3, 2) (x, y).

Player 2 is located at (3, 3) (x, y).

Player 3 is located at (4, 6) (x, y).

Player 4 is located at (6, 3) (x, y).

Player 6 is located at (8, 2) (x, y).

Removed player 1 from the board.

Player List:

Player 2 is located at (3, 3) (x, y).

Player 3 is located at (4, 6) (x, y).

Player 4 is located at (3, 2) (x, y).

Player 6 is located at (8, 2) (x, y).

Removed player 6 from the board.

Player List:

Player 2 is located at (3, 3) (x, y).

Player 3 is located at (4, 6) (x, y).

Player 4 is located at (8, 2) (x, y).

ERROR! ID does not exist on board.

Player List:

Player 2 is located at (3, 3) (x, y).

Player 3 is located at (4, 6) (x, y).

Player 4 is located at (8, 2) (x, y).

Removed player 3 from the board.

Player List:

Player 2 is located at (3, 3) (x, y).

Player 4 is located at (4, 6) (x, y).

ERROR! ID does not exist on board.

Player List:

Player 2 is located at (3, 3) (x, y).

Player 4 is located at (4, 6) (x, y).

Player List:

Player 2 is located at (3, 3) (x, y).

Player 4 is located at (4, 3) (x, y).