For my simulation I would like to make a simple simulation of a forest fire (very original idea). The program will randomly generate a forest and then select another random spot for a fire to start and the simulation will start from there. I plan to use wind direction and wind speed for the 2 variables that the user can manipulate. At no wind the fire will simply spread to all adjacent squares, however depending on the wind speed the fire can spread multiple squares away in the direction the wind is. As I have no knowledge on actual wind speeds and how that affects fire spreading, I will represent it in number of spaces the fire can spread.

**Classes**

|  |
| --- |
| **Tree** |
| **Fields:**  -Boolean isBurning  -Char icon; |
| **Methods:**  +setBurning()  + bool getBurning() |

Represents Each Tree in the Forrest

|  |
| --- |
| **Board** |
| **Fields:**  **-Vector<vector<Tree>> forest;** |
| **Methods:**  **+createForest()**  **+ Tree getIndex(int x, int y)**  **+ Vector<vector<Tree>> getForest()**  **+spreadFire(int windX, int windY, int windSpeed)** |

Handles control of the simulation and how the fire spreads. It will send the vector for the board out.

|  |
| --- |
| **Render** |
| **Fields:**  -**int colorOne**  **-int colorTwo** |
| **+Render(int colorOne, int colorTwo, int Background)**  **+printGrid(vector<vector<Tree>> board());** |

Handles the drawing of the output including controlling the color of the console

**Tests:**

There should be a test for each method in every class.

Tree: Needs a test for its set and get function

Board: ill need the board to print off its output after createForest to make sure it creates it properly.

The search in getIndex will need testing to make sure it finds the right one

SpreadFire will need the most testing. I think Ill have to split it up and test the individual parts.

The output in toString will have to be tested to make sure it is correct

Render

I will send a test output to it to insure it prints correctly.