Graham Brown and Hasan Tiwana

CS302 Final Project Timelog

- March 3, 2020
    - Present: all members
    - Time spent: 30 minutes
    - We brainstormed project ideas together and began working on our project proposal. We came up with a variety of different ideas, but have not yet decided on a final idea.
- March 5, 2020
    - Present: all members
    - Time spent: 1 hour
    - We decided we wanted to use graphs as our primary data structure, and explored various different ideas related to that. Our project idea we are leaning towards is a prison escape game that uses an adjacency list to move through different scenes throughout the game.
- March 9, 2020
    - Present: all members
    - Time spent: 3 hours
    - I downloaded Unity and began watching YouTube videos to familiarize myself with the interface. After learning how to create new scenes and attach C# code to objects, I created a title screen with some test buttons. Finally, I created a Github repository and uploaded what I had created so far to share with my partner.
- March 13, 2020
    - Present: all members
    - Time spent: 30 minutes
    - While working on Challenge 7, we created a set of milestones to work towards and goals we wanted to achieve. We also created a rubric with a set of criteria we wanted to meet for when our project is finally finished.
- March 20, 2020
    - Present: all members
    - Time spent: 1.5 hours
    - We planned out the flow of our game, planning details such as how the player interacts with the game, what the objective is, and how the player can win the game. We discussed how we want to create our game's map environment, considering options such as using images, hand-drawing graphics, or creating our own models. We also discussed how we want to implement the graphs and adjacency lists we're using for our data structure. For our adjacency list, we'll represent edges with buttons, where each button leads to an indexed scene, and

the player will interact with these buttons to try to make it to the end and win the game.

- March 28, 2020
    - Present: all members
    - Time spent: 2 hours
    - We restructured our game, changing some of our original ideas. Our new idea would still use the same data structures, but instead we wanted to create a memorization-based game that was more player-oriented. For our new idea, we would create a grid with randomly placed obstacles. Then we would create a path from a start position to an end position on the other side of the grid. The player must follow this exact path, or they are sent back to the start position. We hope that adding a new layer of player involvement will make our game more interesting, complex, and enjoyable.
- April 2, 2020
    - Present: all members
    - Time spent: 30 minutes
    - We decided to use Dijkstra's algorithm to create a random path, and we began thinking of a theme for our game. Two of the main themes we are deciding between are a "lost in the woods" theme and a minefield theme.
- April 8, 2020
    - Present: Hasan Tiwana
    - Time spent: 2 hours
    - I created a graph of nodes represented by a grid, where some of the nodes are normal, walkable nodes and some are obstacles that you must go around. As of right now, players can walk around on all tiles, but once we implement Dijkstra's method and find a path, we will limit movement to that single path.
- April 13, 2020
    - Present: all members
    - Time spent: 2 hours
    - The two of us implemented Dijkstra's algorithm in our new Player class, taking turns coding while screensharing and discussing. We created a function that shows the path that is found by changing the sprites for tiles in that path to test that it works. One issue we found was that with some arrangements of obstacles, there is no possible path from the start position to the end position. As a result, if a path cannot be found, we reinitialize the grid with new obstacles and retry Dijkstra's algorithm.
- April 19, 2020
    - Present: all members
    - Time spent: 2.5 hour

    - We implemented more features to our game. If the player steps on a tile that isn't in the path found, they are sent back to the start position. We decided on using 8x8 for the grid size and we experimented with using different models to try to add a theme to our game, but ended up leaving it as is for our lab demo. We also decided to add an easy mode and a hard mode for players seeking a harder challenge. Easy mode will briefly show you the path you have to follow, while hard mode forces you to find it through trial and error. Both modes force you to memorize the path.
- April 20, 2020
    - Present: all members
    - Time spent: 30 minutes
    - We discussed some of the smaller details in our program, such as grid size, number of obstacles to create, and the color scheme of our project.
- April 23, 2020
    - Present: all members
    - Time spent: 2 hours
    - We added a theme to our game, and titled our game "Mine Meadows." Whenever a player steps on a tile that isn't in the path, we change the sprite to reveal a landmine on that tile. We also created variables for keeping track of whether the player chose easy or hard mode, and changed the gameplay accordingly.
- April 25, 2020
    - Present: Hasan Tiwana
    - Time spent: 30 minutes
    - Cleaned up some of our code to make it more neat, added some comments throughout. I simplified any code that could be simplified and reorganized where the files were located in our project folder.