Prototype: Project 2

By Azmat Ishaq
CART 253 with Professor Pippin Barr

The Game

The proposal for my project is to create a JavaScript maze game that can be played in a computer browser. In order to make it interesting, I will try to make the walls of the maze slowly close in on the player in a dynamic fashion.

Also, the player character will move slower when the walls are touching them. At that moment a countdown timer will begin before the character is crushed and the game is over. If the player can get to the end of the maze without getting crushed, they win the game.

If possible, I will also add a fog of war to make the game even more challenging.

States

I will utilize states to code the game.

First, the user will be introduced to a title state. The title state will present the instructions on how to play the game.

The player will be able to press a key, by using the keyPressed() function, in order to proceed to the animation state to begin playing.

If the player completes the maze successfully, by reaching the end, then the winning state of the game will be triggered.

If the player loses, by getting squeezed by the wall, then the losing state of the game will be triggered.

After either outcome, I will use a function to reset the game so the player can go back to the title screen.

The Maze

Ideally I would be able to randomize the maze on every playthrough. However, for now, I will be making the map manually.

For the prototype simulation I have created a simple grid structure using an array to represent the playing area. The prototype currently has a randomized maze. If the randomness of it can be maintained without sacrificing the gameplay then it will remain integrated into the final project.

I can use this grid to track the player and place items on the map. The current map is based on Pippin's example: https://editor.p5js.org/pippinbarr/sketches/Xq3qsbQWA.

I will probably have to use the dist() function to trigger certain events as well. For example, it will be useful when the distance between the player character and the walls has reached a certain point.

The walls will be in their own array and will be placed on top of the established grid. I have already placed a single wall in the prototype simulation and will soon begin testing how the player can interact with it more dynamically.

The Timer

I will have to get the timer to start and reset whenever certain events occur in the game. Along with conditionals, I will likely use the dist() or map() function, as well as setTimeout(), in order to accomplish this.

Classes and Contact Between Objects

Classes will be utilized in order to make interaction between different elements of the code easier and more efficient.

However, getting objects to interact, could be challenging. For example, I am not sure how to get the walls to actually squeeze the character. It will likely involve using the dist () or map() functions. Furthermore, if the character is touching the wall but not being squeezed I would have to make sure that the suffocation timer does not begin unnecessarily. Using if else statements in this context may be the solution.

The Fog of War

In order to make the maze more difficult, it may make sense to have a fog of war. This way the player would have to explore the maze in order to solve it. This would also likely involve utilizing the dist() function. However, I will have to explore more options in this regard.

Checkpoints

There will be checkpoints that the player has to collect which will be created, and displayed, using an array. Furthermore, if the randomized maze option can be implemented, I will attempt to randomize the checkpoint’s position as well.

The checkpoints can be represented by a small colourful item on screen that the player character has to touch. Once they touch the item, it will disappear and a small chime will be executed to mark the collection of a checkpoint item.

Once the player touches the checkpoint, the walls will diminish back to a less constraining size. If else statements will likely be utilized for this also.

The Player and Character

The player will navigate the maze with the Arrow Keys. The character model will be a 2d sprite with a retro color and design.

Other Challenges

There will be many challenges with this project. Getting the maze to give the appearance that it is interacting with the player character authentically will be difficult. If it simply overlaps the player the user will get no sense that they are being squeezed.

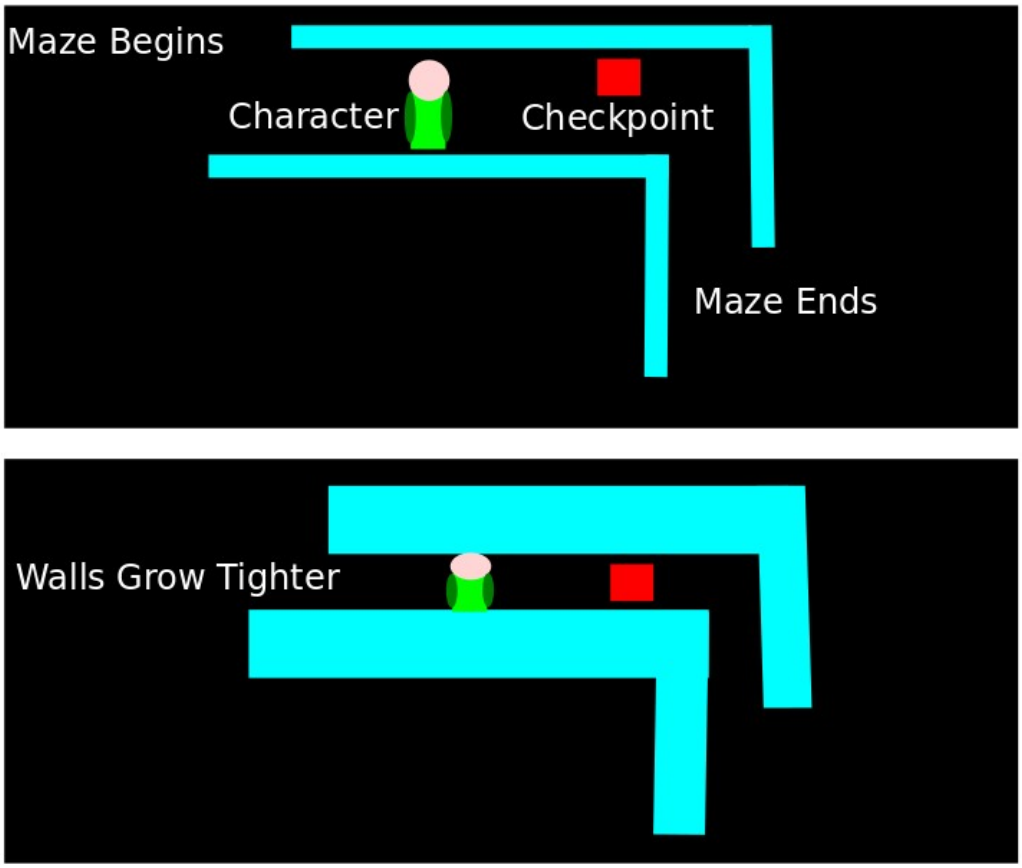
Furthermore the collision detection between objects may prove to be a challenge. This is something I will have to study more thoroughly in order to implement competently.

Content Creep

I will also try to watch out for content creep. This is when, as a developer, one attempts to add too many features to what they are making.

Images

Here is an illustration of a rough outline for the game. The maze will be more sophisticated once it is completed. Furthermore, the games appearance and colors will have a 2d retro feel to them.



Here is another sketch I drew: