

# Maze Maker/Solver: An Educational Tool

By Nathan Joyal CS 480 Fall 2021

Advisor: Randy Appleton

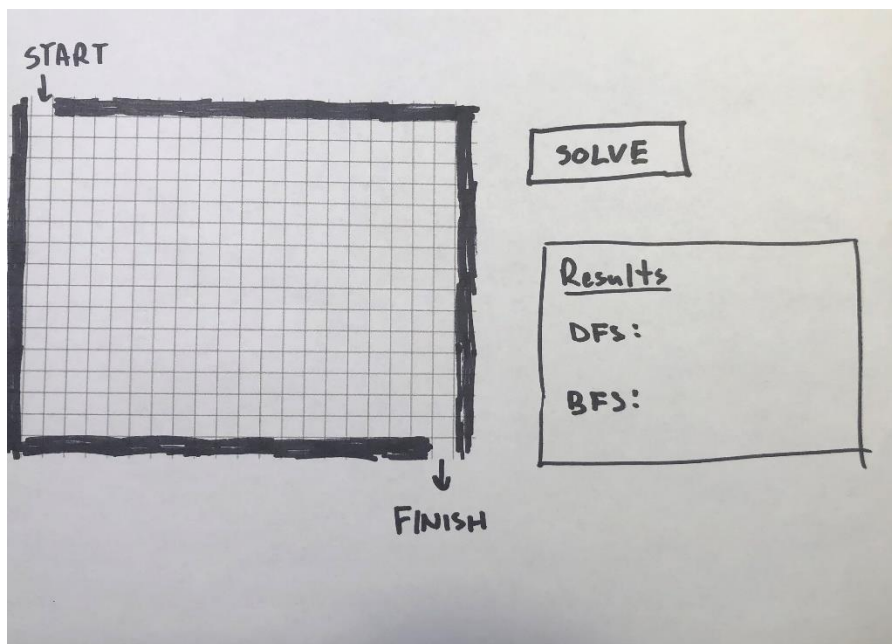
Grading Committee: Andy Poe, Jeff Horn

## Objective

To create an interactive learning tool to demonstrate fundamental concepts about search as a computer science concept to middle and high school aged users.

## Project Description

The following rough drawing is a representation of what the program will look like to the end user.



The user will be able to use the mouse to select squares in the grid. By clicking on the squares, the user toggles whether the squares are filled or open, with the goal of creating a maze. When the user is satisfied with their maze, they will click on the “solve” button, at which point the computer solves the maze.

The interesting part here is that the computer will solve the maze with both a “Depth First” algorithm and a “Breadth First” algorithm. It will track the progress of both in solving the maze visually on the user’s screen, and report statistical information about the results – how many squares were explored, how long the algorithm took to run, etc. The goal in doing this, is to allow young users to explore and learn about these concepts by creating various mazes for the computer to solve.

## Learning Objectives

In order to create a maximum degree of accessibility and portability for this product, I want to program it in JavaScript mounted in a web interface using HTML 5 and HTML Canvas (I think). Since I have never programmed in any of these languages / environments at all, I have QUITE A BIT of work ahead of me in terms of learning basic syntax and operation. For this reason, as you will see below, my grading rubric is broken into two sections, “Learning Goals” and “Development Goals”. The Learning Goals section is quite substantial, because learning and demonstrating the basic working knowledge of the technology involved will be a substantial portion of the accomplishment demonstrated through this project.

Specifically what I need to learn will come into focus as I progress deeper into this project. However, immediately in addition to a basic understanding of syntax is JavaScript, I will need to develop a working knowledge of class, methods, program organization, function closures, one or more visual libraries, and recursive use of the stack in JavaScript. I will also have to learn how to code some basic HTML 5, and how to interface my JavaScript application with a webpage it is mounted within. Additionally, my initial research leads me to believe that I may have to use HTML Canvas visual objects to display my JavaScript program to the end user. Alternatively, time may compel me to ignore many of the HTML aspects I hope to master, and focus on creating a stand-alone JavaScript program that could later be incorporated into a website. This will also depend on the functionality of any visual libraries I am able to discover and master within JavaScript itself.

## Developmental Objectives

My immediately development goals are for the program to appear to the user more or less in keeping with the rough image included here. The map squares should functionally toggle between filled and not filled, and this should be clearly and fluidly represented on the screen. The solve button should work to activate the computer. Once active, the computer will succeed in using both Depth-First Search and Breadth-First Search. The computer’s progress will be tracked on the map, once it is finished. Results about how many squares were visited by and how long each algorithm took will be displayed to the user. These goals represent what I would consider a “minimum functioning” product.

Immediately after this, my first “stretch” goals would include adding the ability to save and load maps. Then I would intend to package the product with a handful of pre-fab maps, specifically at least one where BFS excels, and one where DFS is superior. Additionally, I would like to have the squares the computer visits while solving the maze be marked on the maze in real time, as the computer solves it. Two different colors will be used to indicate BFS and DFS progress, and will be displayed on the map in a way such as not to conflict with each other. Basic sound effects could be incorporated for toggling the squares and for coloring tiles during the computer’s search. Basic animations for buttons, computer searching squares, and adding results information to the screen could also be incorporated. Lastly, the program would ideally be mounted in a web application which could easily be hosted online. This would create the maximum amount of portability to reach the widest audience possible.