

Maze Solver

Spring 2018

This project takes in a maze text file and determines if it solvable or not. It was originally to practice data structures, specifically stacks. Over the years I have made tweaks to improve it and attempt other things I was learning about. The image below is a small example. The program starts at S and attempts to travel to G.

```
Yes it can be solved! :)
S ### . . # . . ### . . # .
0 ### . . . # . . # . # . .
0 # . . . # . # # . . # # # .
0 0 # 0 # . # 0 0 # . # # 0 0 #
# 0 0 0 0 # # 0 0 # . # 0 0 # 0
0 # # # 0 0 0 # 0 # . . # 0 0 0
0 0 0 0 # 0 0 0 0 # # # 0 0 # 0
0 0 # # # # 0 # # # . . # 0 # 0
0 # # . # # 0 0 # # # # 0 0 0 0
0 0 # # 0 # # 0 # . # . # # 0 0
# 0 0 0 0 0 0 0 # # . # 0 0 0 0
. # 0 # # 0 0 # # . # # 0 # 0 #
. # # # # # 0 0 0 # 0 0 0 # 0 0
# # . . . # # 0 0 0 0 0 0 # # 0
. . . # # . . # # # # 0 . # 0
# # . . . . . # # # # # # G
```

At one point in my coding career I found this program to not only be complex but also frustrating. I distinctly remember spending far too long on a bug that turned out to be a misspelled variable. I have come a long way from writing my first linked lists (misspellings still haunt me) and this program has come with because it looks interesting and its one of my first “real” programs.

This program has enough going on to allow me to tinker with it while not being so complex I get severely confused. One of the first things I did was play with the visual solution. The code makes it easy to switch from DFS to BFS and the difference is fun to see. I also animated the solution by printing out every move so I could watch it work.

Later I added color and played around with the best “path” character to represent where the algorithm had traveled to. Neither is too complex but both add a lot for the observer.

Another addition was writing a script to run this program. I find this program is an interesting one to show off to anyone because of its simple and visual representation. So I created an alias for a script that runs through three different sizes of maze.

None of the code for this project is all that impressive, even compared to work I would do later in the year. But the enjoyable visuals make it my favorite program to tinker with and come back to.

You can check out the code for this project [here](#).