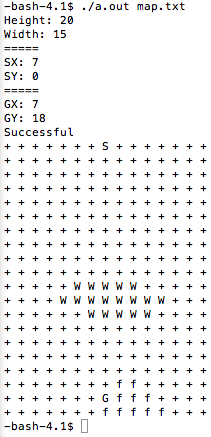
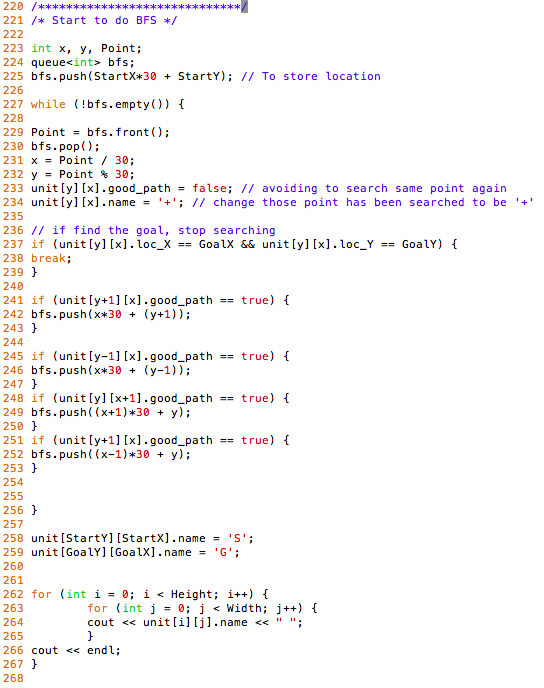
### This project will use Breadth First Search, Lowest Cost Search, Iterative Deeping Search, and A\* Search, to find the path from start point to goal point in an ASCII grid map. In the experiment, it shows the advantages and disadvantages of each algorithm, and it will also tell the idea how each algorithm works, and how it search in the grip, how many times for compiling, and how many grids will be searched.

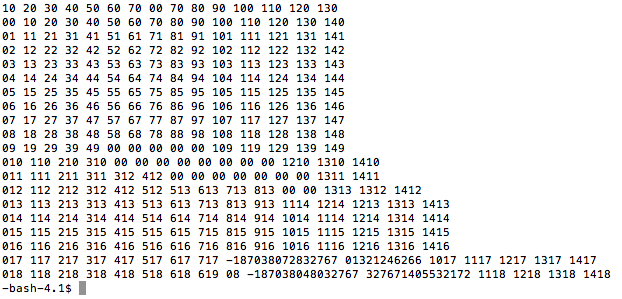
This Project used BFS to search the map from Start point to the Goal point. The main idea of my work is to. The first picture will shows output map with those points which has been searched and mark it with the symbol ‘+’(pic1); The next picture will show the shortest path that avoid these water area from Start to Goal. Besides, in general the map will show information in these symbols: start point mark with ‘S’, goal point marked with ‘G’. To follow up the requirement, the word ‘W’ means water and can’t be traveled, and ‘@’shows the shortest path.

In the first part of BFS, it will need to travel from the start point, and check each points beside of it, if that is not been marked (not been traveled) and can be traveled (not ‘W’ water), the program will store these points information (made by a number which convert by location X and Y, see line 231,232) into the queue, the while loop will keep searching until it finds the goal point. As the output shows, the symbol ‘f’ beside of G means not searched yet, because the program already stop when it found

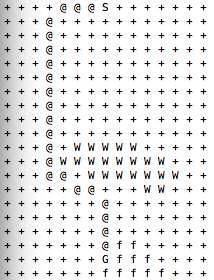
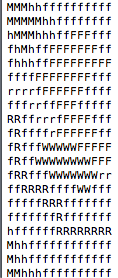
the goal.

(pic1)

The second part of the BFS is to find out the shortest path from the start point to goal point. The method used to find the path is to store the “Father Location Information” in the Node, this father location (fx and fy) will contain the location which the current node come from, in the other words, each node points to higher level layer. With this information, we can find the path from goal point to start point easier. The pic2 shows how each point’s father node’s location.

 (pic2)

The picture3 shows the final map, which has shortest path after doing BFS. It avoid the water area and successful to make the path from Goal point to the Start point. BFS is a algorithm which good for small amount searching, because even it’s a 15x20 grid, it took about 5 seconds to find the shortest path. To compare with the original map in picture 4, we can see that it search most of points in the map, it takes much memory on searching.

(pic3) (pic4)

To conclusive, BFS takes a lot of memories and time for searching, but this algorithm is greatly with map searching. If there has one path, use BFS algorithm will find at least one path, and if there has more than one path, BFS will find the shortest path of a map.