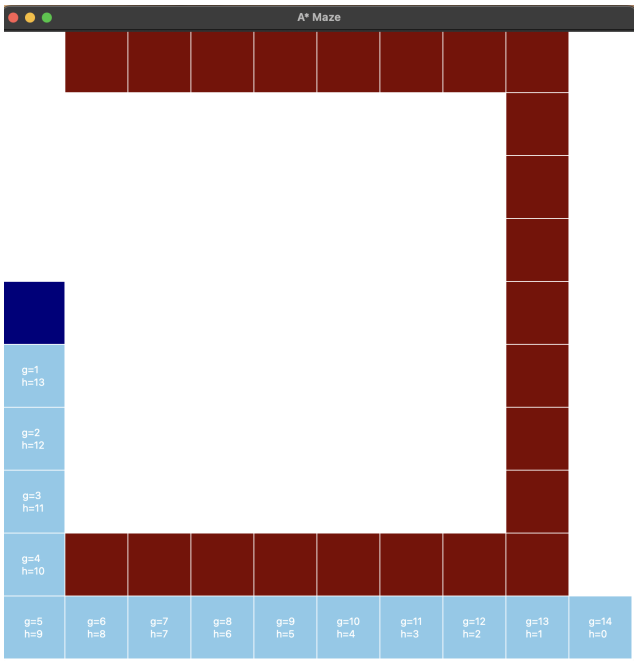
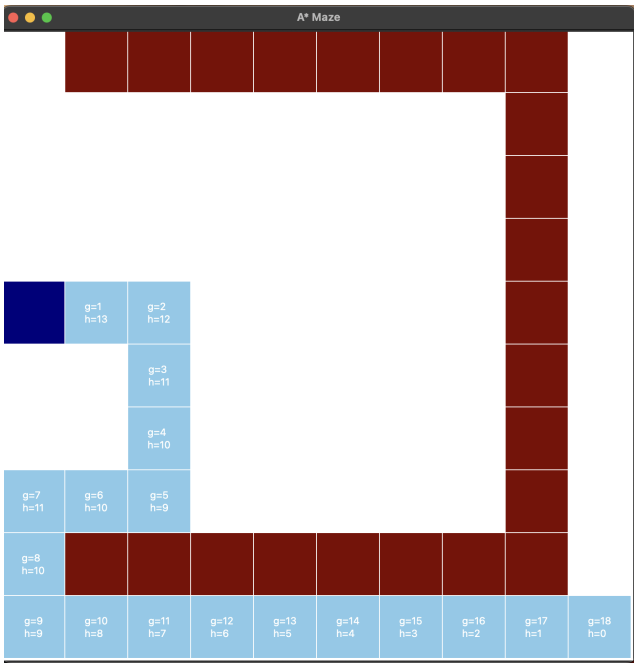
#HW3 problem 1 Simon Tice





The A* algorithm is on the left and the greedy best-first algorithm on the right. The reason these two algorithms produce different results is because of the evaluation function. The A* algorithm takes into account the cost of how far the algorithm has gone, as well as the estimated cost to the goal, so it balances the two and finds the shortest path every time. On the other hand, the greedy best-first algorithm only considers the estimated cost to get to the goal without taking into account the distance already gone. Because it just blindly follows the heuristic, it can go the wrong way and take suboptimal paths.

The way I altered the algorithm was by finding the place the A* algorithm updated the evaluation function, and then I deleted the + g(n) so it only took into account h(n).

```
##### USE THIS FOR A*
## self.cells[new_pos[0]][new_pos[1]].f = new_g + self.cells[new_pos[0]][new_pos[1]].h

## USE THIS FOR GREEDY BEST FIRST
self.cells[new_pos[0]][new_pos[1]].f = self.cells[new_pos[0]][new_pos[1]].h
```

As you can see, the greedy best-first algorithm is lacking the “new_g +”