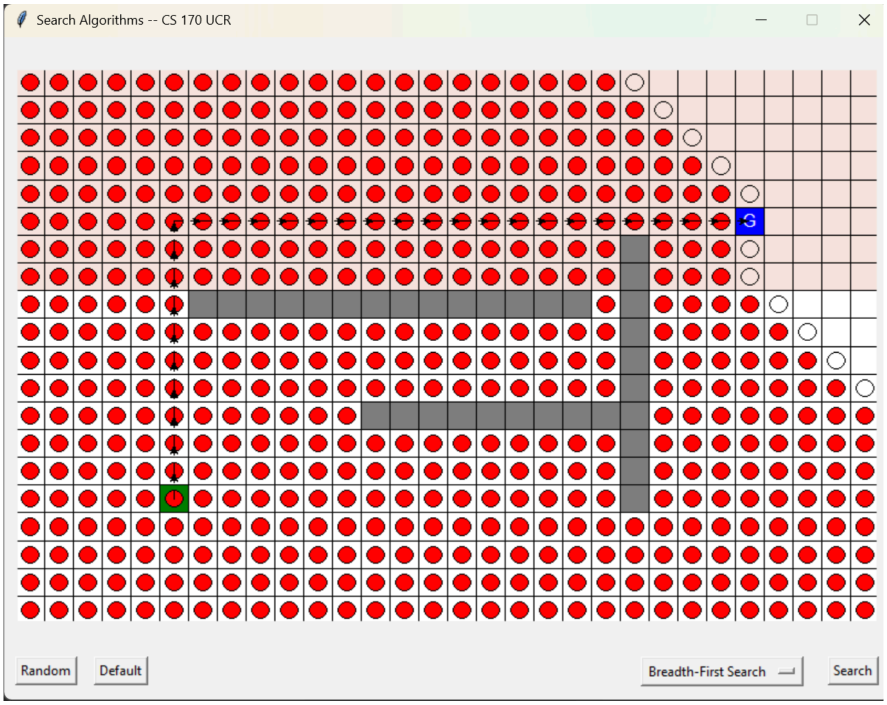
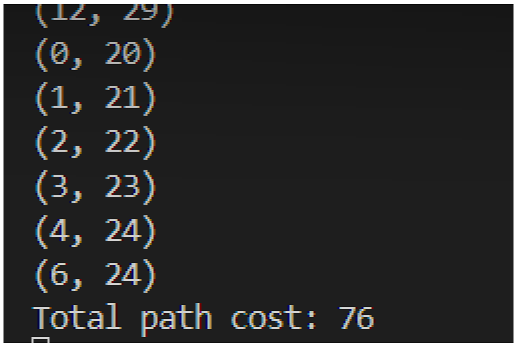
Natalie Shin, Ray Wang

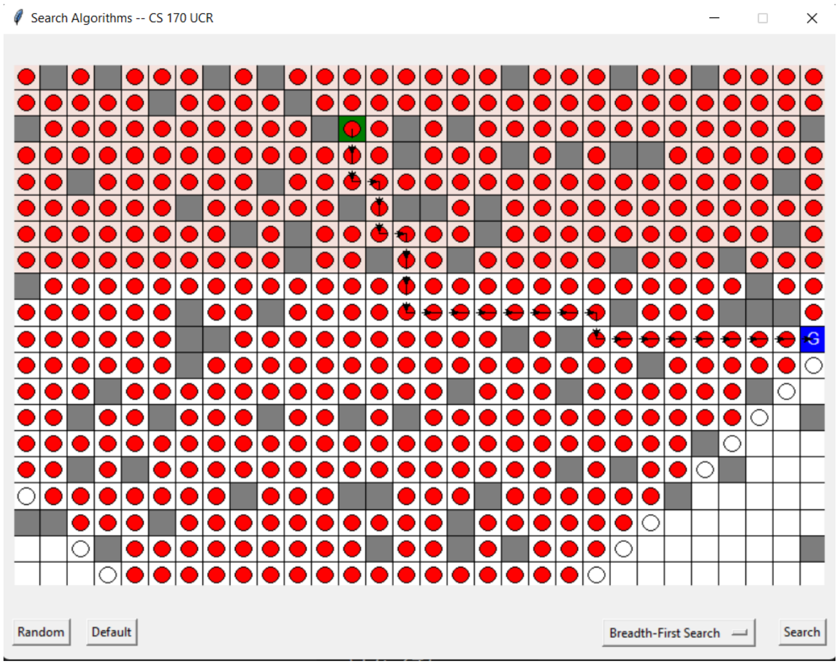
BFS Given Map

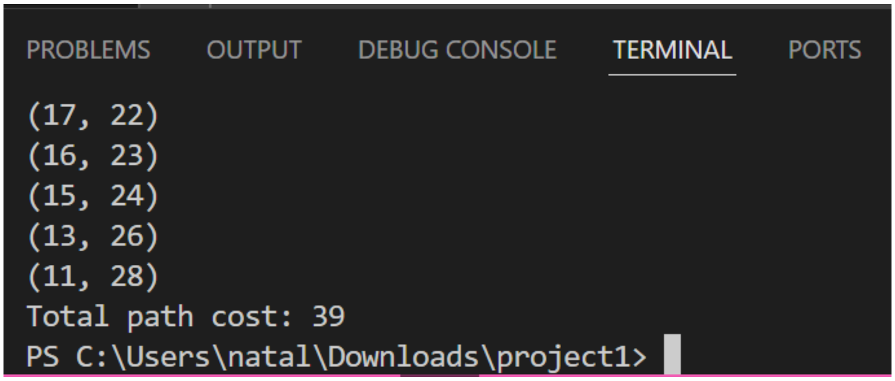




```
(12, 29)
(0, 20)
(1, 21)
(2, 22)
(3, 23)
(4, 24)
(6, 24)
Total path cost: 76
```

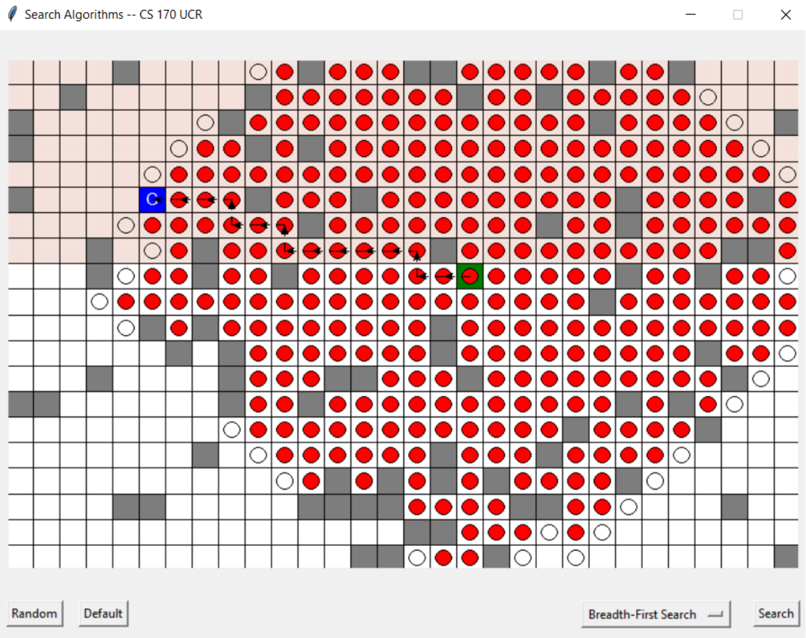
BFS Random:

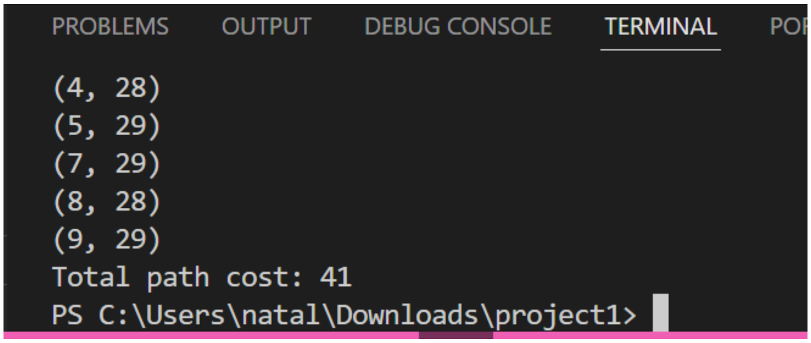




```
(17, 22)
(16, 23)
(15, 24)
(13, 26)
(11, 28)
Total path cost: 39
PS C:\Users\natal\Downloads\project1>
```

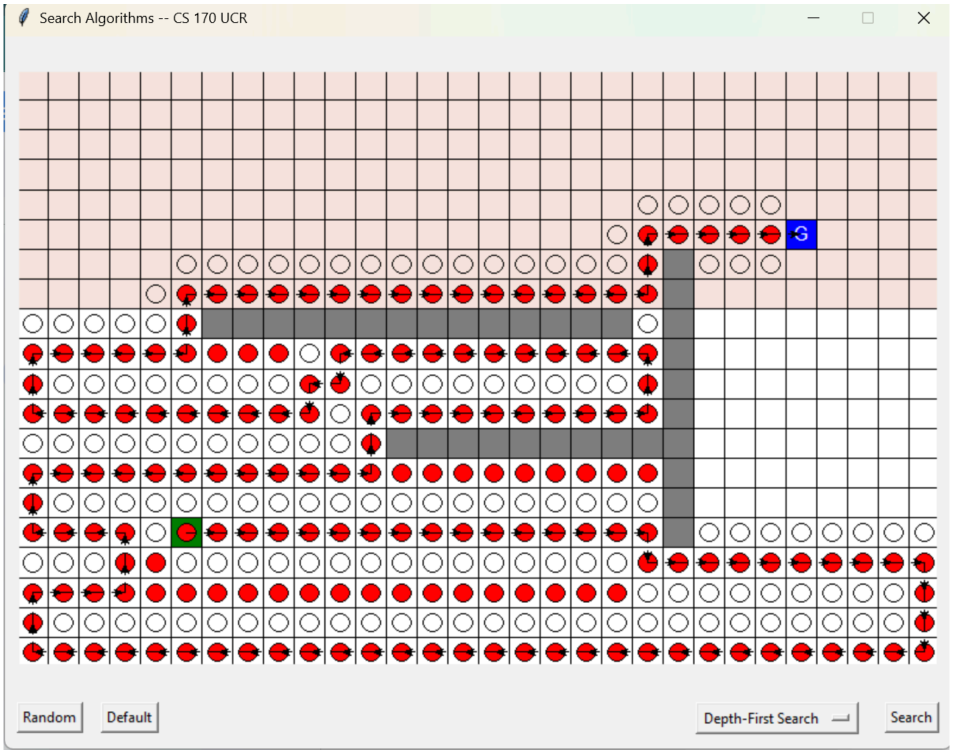
BFS Random:

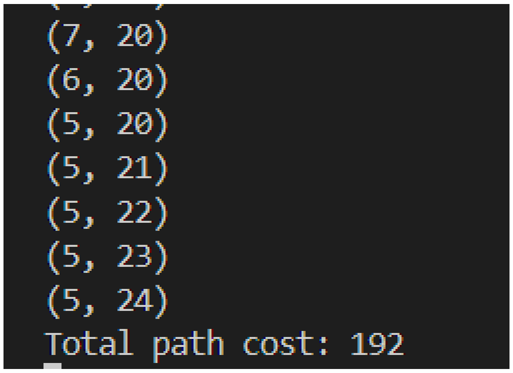




```
(4, 28)
(5, 29)
(7, 29)
(8, 28)
(9, 29)
Total path cost: 41
PS C:\Users\natal\Downloads\project1>
```

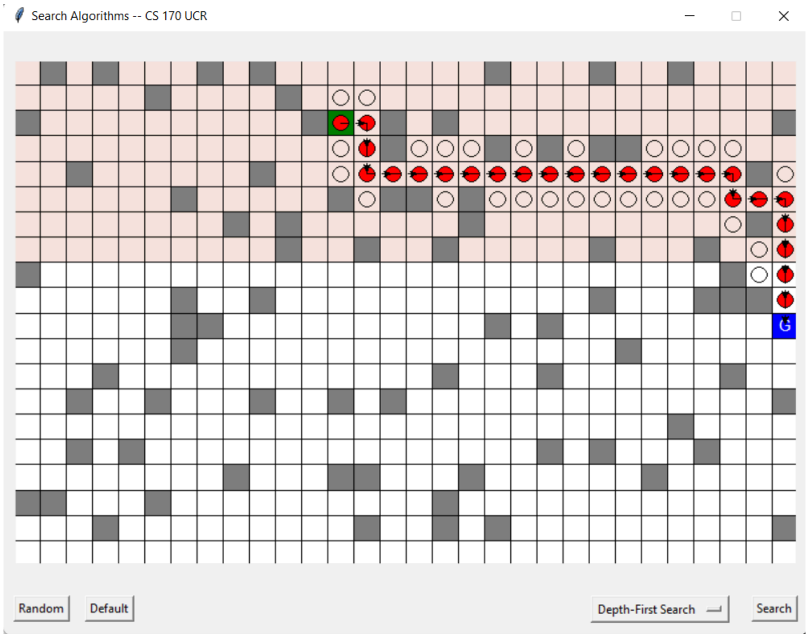
DFS Given

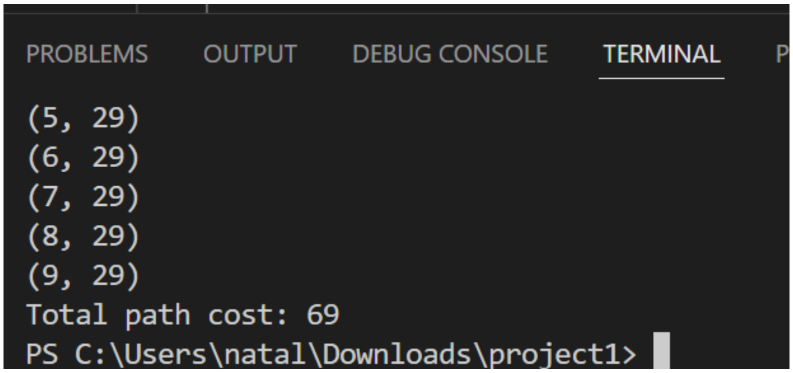




```
(7, 20)
(6, 20)
(5, 20)
(5, 21)
(5, 22)
(5, 23)
(5, 24)
Total path cost: 192
```

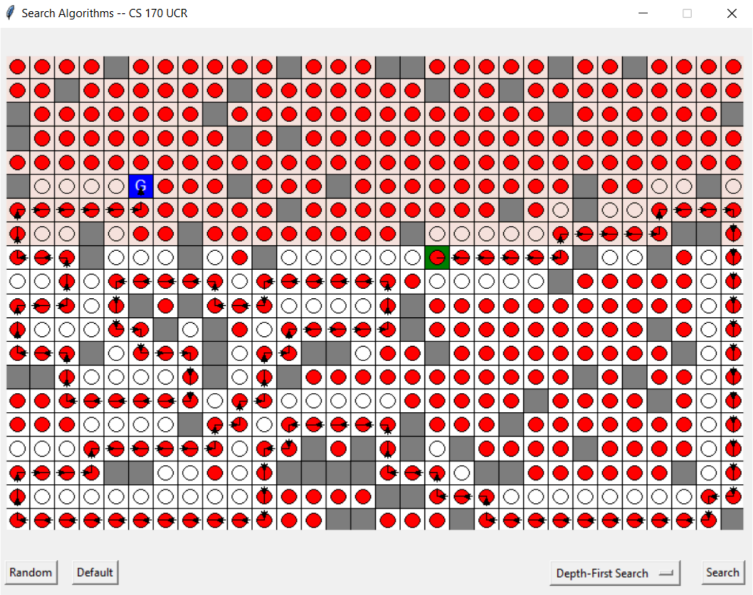
DFS Random

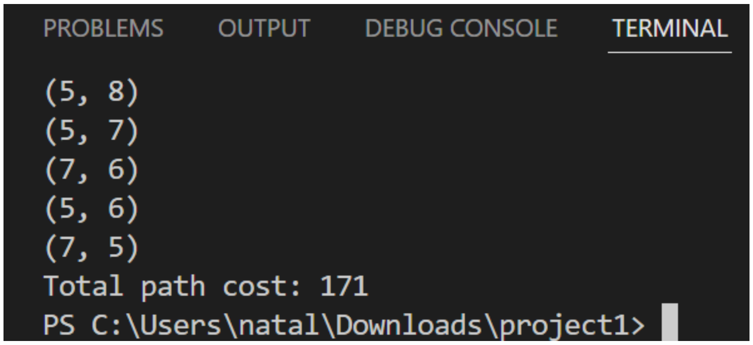




```
(5, 29)
(6, 29)
(7, 29)
(8, 29)
(9, 29)
Total path cost: 69
PS C:\Users\natal\Downloads\project1>
```

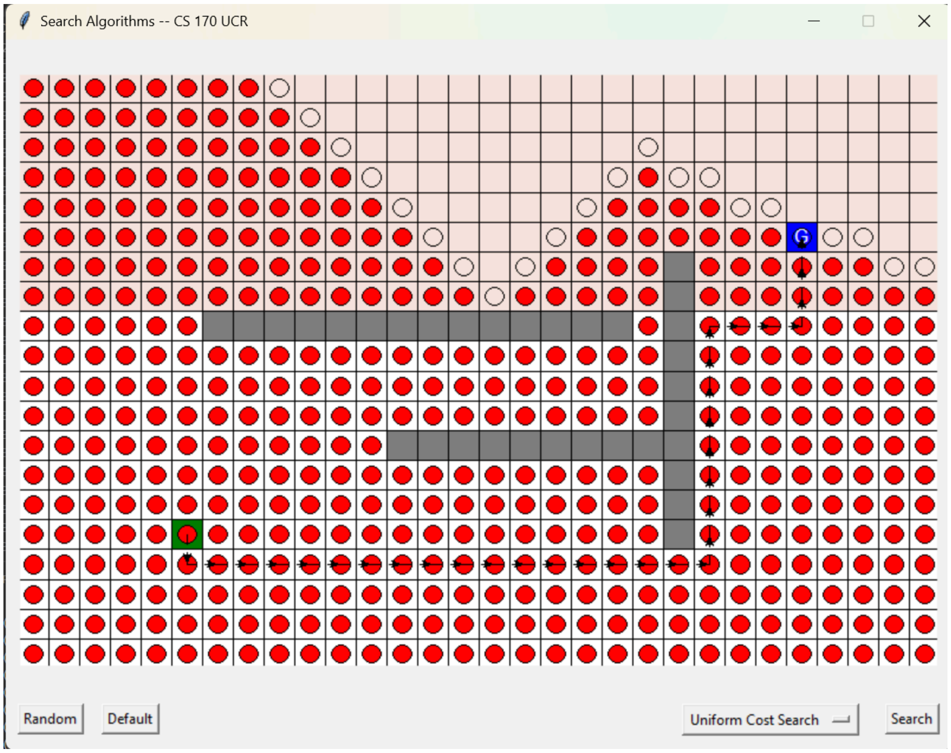
DFS Random:

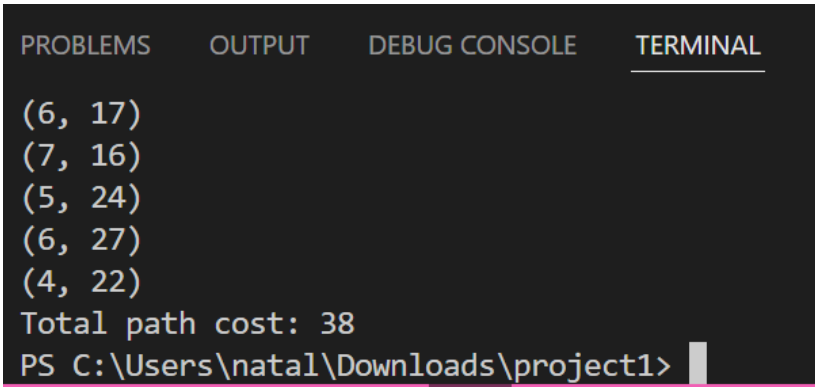




```
(5, 8)
(5, 7)
(7, 6)
(5, 6)
(7, 5)
Total path cost: 171
PS C:\Users\natal\Downloads\project1>
```

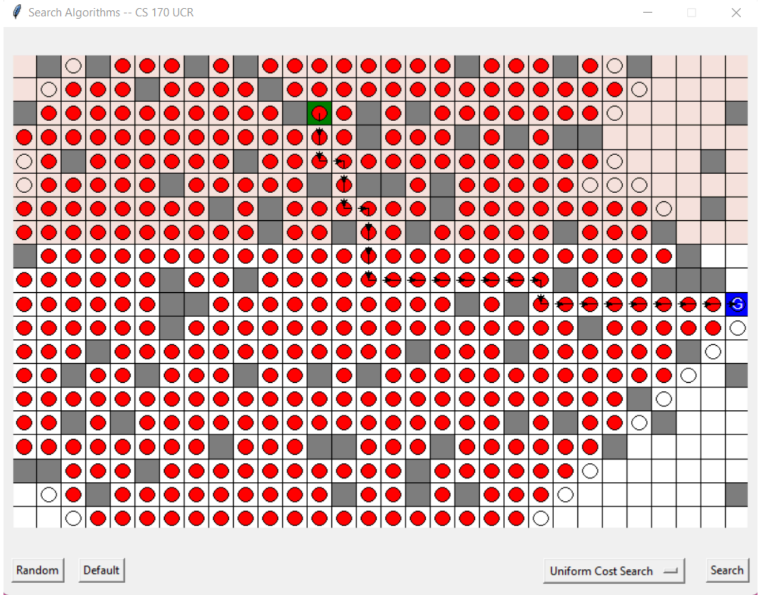
UCS Given

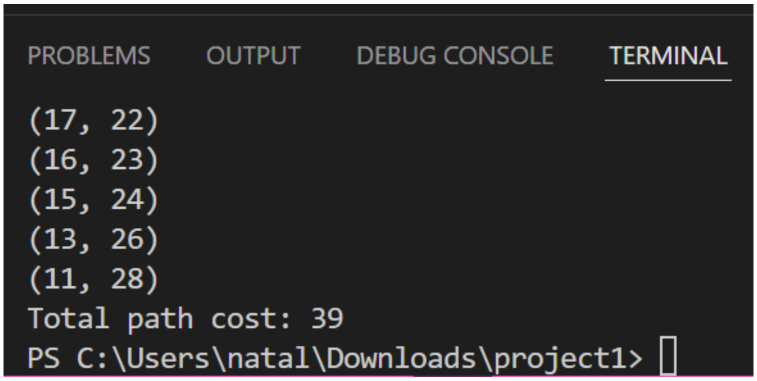




```
(6, 17)
(7, 16)
(5, 24)
(6, 27)
(4, 22)
Total path cost: 38
PS C:\Users\natal\Downloads\project1>
```

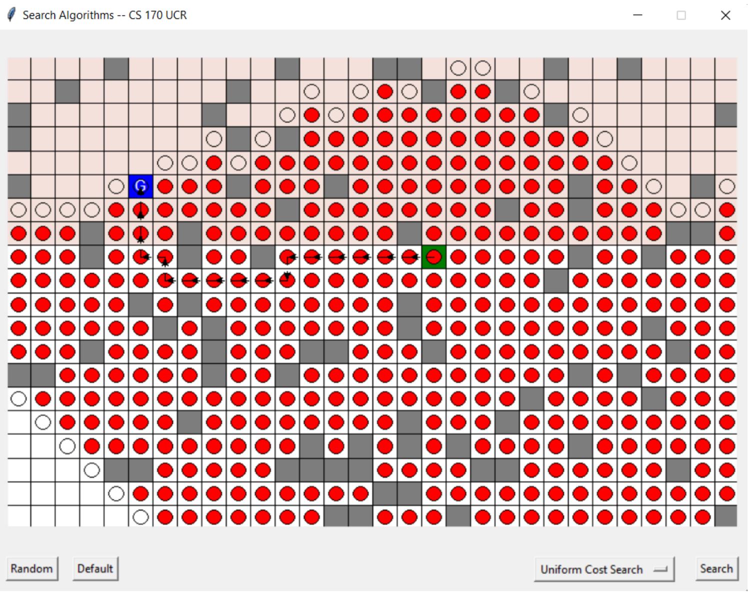
UCS Random

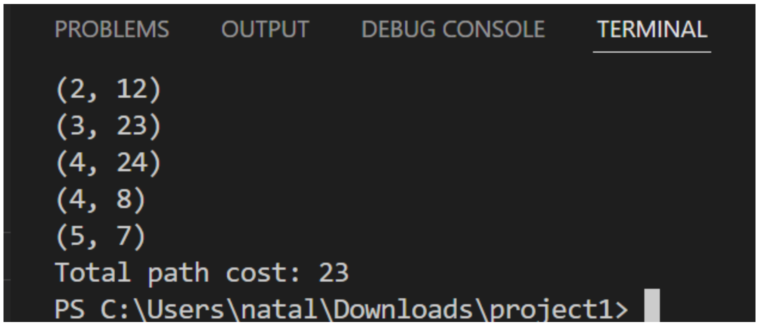




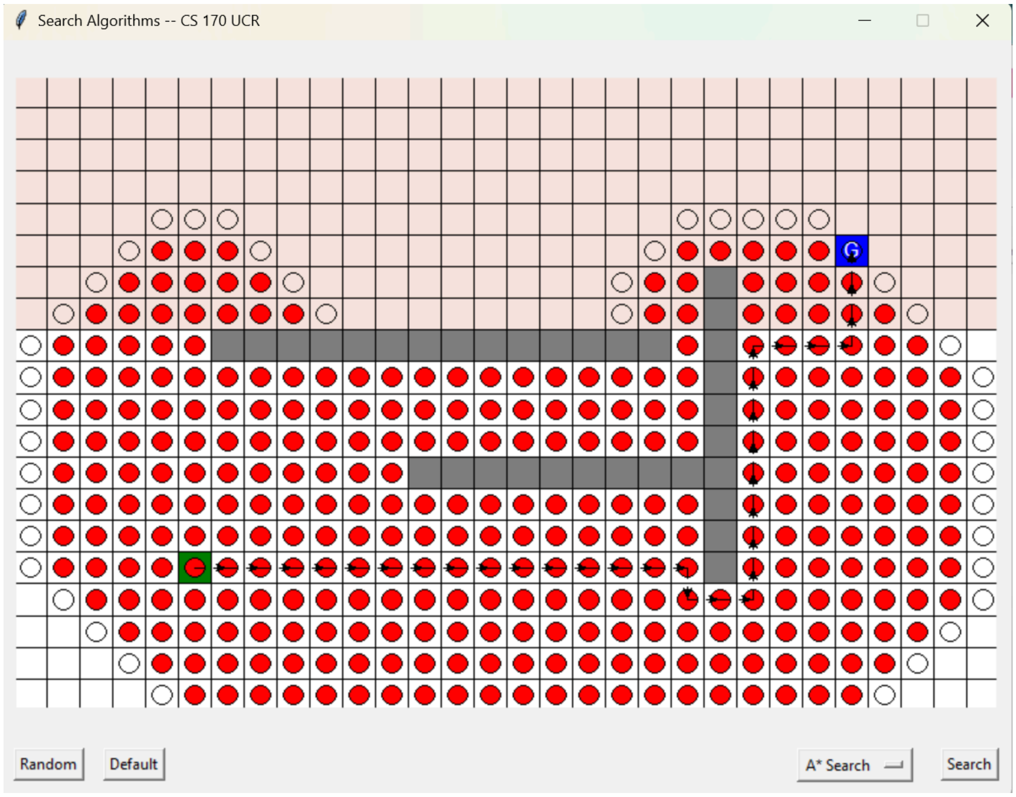
```
(17, 22)
(16, 23)
(15, 24)
(13, 26)
(11, 28)
Total path cost: 39
PS C:\Users\natal\Downloads\project1> 
```

UCS Random:

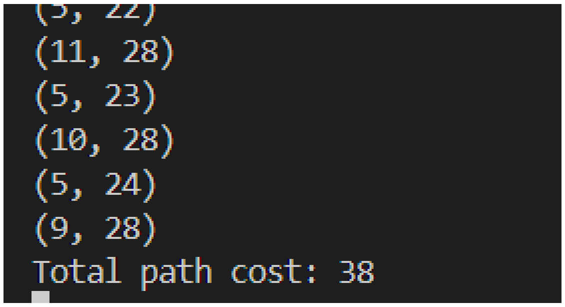




```
(2, 12)
(3, 23)
(4, 24)
(4, 8)
(5, 7)
Total path cost: 23
PS C:\Users\natal\Downloads\project1>
```

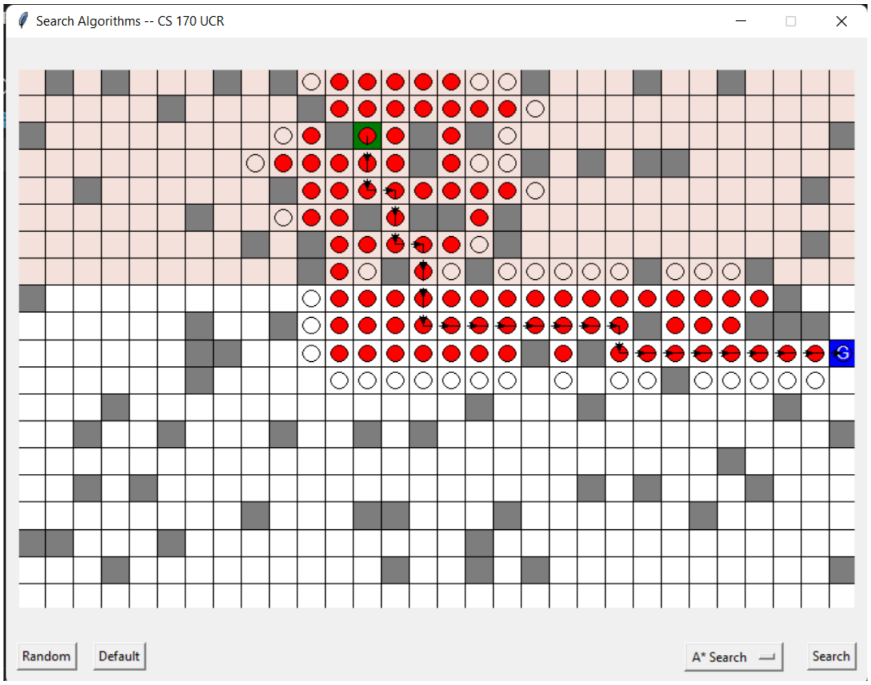


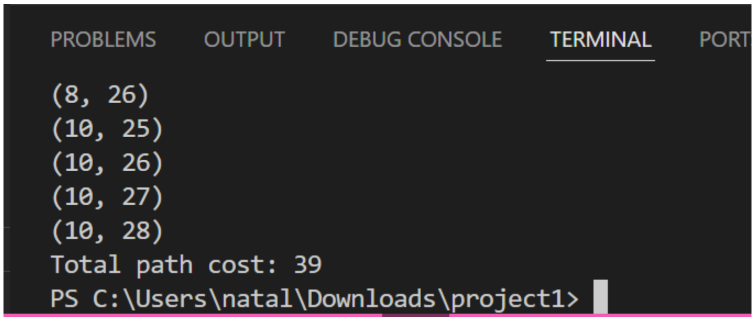
A* Given



```
(5, 22)
(11, 28)
(5, 23)
(10, 28)
(5, 24)
(9, 28)
Total path cost: 38
```

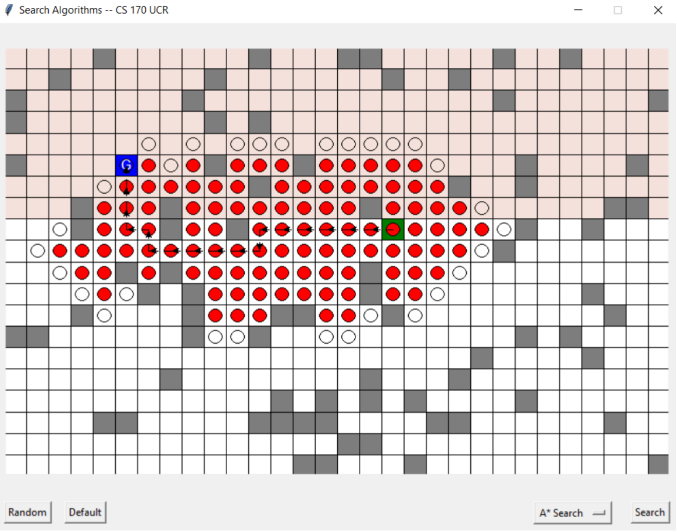
A* Random

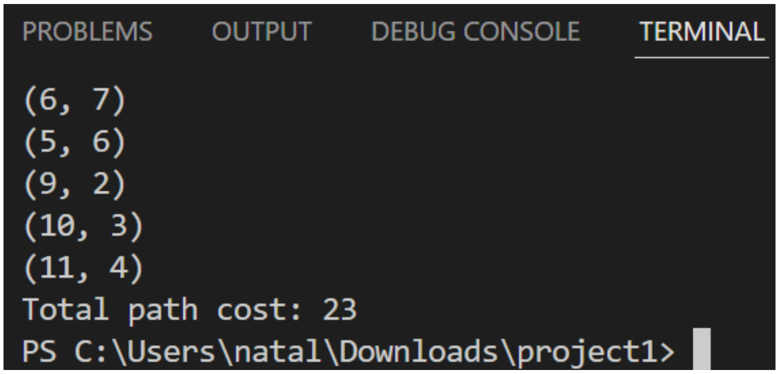




```
(8, 26)
(10, 25)
(10, 26)
(10, 27)
(10, 28)
Total path cost: 39
PS C:\Users\natal\Downloads\project1>
```

A* Random





```
(6, 7)
(5, 6)
(9, 2)
(10, 3)
(11, 4)
Total path cost: 23
PS C:\Users\natal\Downloads\project1>
```

Report questions:

1. Between BFS and DFS, both take a long time to generate and are not time efficient, and BFS generally gives a shorter path/cost than DFS. UCS and A* both have the same path costs in these cases, but as shown in the screenshots, have different expanding nodes.

Algorithm	cost	rand1	rand2
BFS	76	39	41
DFS	192	69	171
UCS	38	39	23
A*	38	39	23

2. The main difference between A* on the default and uniform cost grids is that the Manhattan distance, AKA the heuristic cost, would be the only variable considered because all squares cost the same to traverse. It would not produce the same path exactly because in the default grid, more than one path can have the same Manhattan distance but different costs because the squares at the top of the map cost more than the white ones at the bottom. To sketch the optimal uniform cost grid path for A*, we'd just need to find the most direct route/path since it has the least squares traversed.
3. On a uniform grid, BFS and UCS would produce the same optimal path. This is because UCS is basically BFS with a cost sorting property implemented via priority queue; when the costs are the same for all directions, it would just end up like BFS, which goes layer by layer.