Nate Sanchez

Easy:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Algorithm | Length of Solution | Cost of Solution | Number of Visited | Max Queue Size |
| BFS | 5 | 17 | 54 | 89 |
| DFS | 895 | 3957 | 910 | 1596 |
| UCS | 5 | 17 | 13 | 22 |
| GBF | 5 | 17 | 5 | 11 |
| A\*1 | 5 | 17 | 7 | 14 |
| A\*2 | 5 | 17 | 5 | 11 |

Medium:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Algorithm | Length of Solution | Cost of Solution | Number of Visited | Max Queue Size |
| BFS | 9 | 31 | 410 | 685 |
| DFS | 31723 | 141643 | 32555 | 55806 |
| UCS | 9 | 31 | 56243 | 70550 |
| GBF | 9 | 31 | 164 | 273 |
| A\*1 | 9 | 31 | 36 | 63 |
| A\*2 | 19 | 71 | 132 | 218 |

Hard:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Algorithm | Length of Solution | Cost of Solution | Number of Visited | Max Queue Size |
| BFS | 30 | 136 | 517720 | 521884 |
| DFS | 72686 | 328150 | 76101 | 126565 |
| UCS | 30 | 128 | 69341 | 86339 |
| GBF | 66 | 296 | 1926 | 3223 |
| A\*1 | 98 | 446 | 3790 | 5788 |
| A\*2 | 76 | 350 | 618 | 1028 |

There is a clear distinction on which algorithms work best and which don’t. Breadth-first search worked well in the easier solutions but performs worse when the case is less ideal. This is because the search is unguided and has to simply stumble on the solution. This could end up getting a good solution but there is no guarantee. Depth-first search functions similarly to breadth first search but performs incredibly terribly in both its solution and the cost of finding that solution. Similar to breadth-first, it relies on chance and has no guidance. However, since breadth -first effectively goes by depth, it is much better at finding the best solution. Uniform-cost search seemed to always find the best solution and performs well in that regard. However, it can be very expensive to run. On the harder puzzles, it is especially slow. Best-first, A\*1, and A\*2 are much more comparable.