Jared Harmon

Programming Assignment 1

My BFS takes a LONG time to complete due to the inefficiencies in the algorithm. All of my trials with randomly generated boards took over 20 mins to run. Time approximated to mins.

|  |  |  |
| --- | --- | --- |
| Board | # of steps | Completion time |
| #1 | 1,221,284 | 22:15 |
| #2 | 1,235,288 | 22:18 |
| #3 | 1,001,292 | 20:54 |

My A\* is obviously not working properly as I am taking longer to finish the solve. I think it has something to do with my competitor/heuristic function not working properly. I do think that my implementation is accurate with the use of a priority queue that adds nodes based on number of misplaced tiles.

|  |  |  |
| --- | --- | --- |
| Board | # of steps | Completion time |
| #1 | 1,431,586 | 24:35 |
| #2 | 1,398,985 | 23:57 |
| #3 | 1,321,129 | 22:46 |

If implemented correctly my A\* should be WAY faster almost instant. With that assumption BFS is a wildly inefficient algorithm. The computer that this was tested on is very high spec for a home PC. The important take away is that smart algorithms like A\* need to be implemented for AI to have any chance at solving anything more complicated in a reasonable amount of time.