**Summary of results for sample test runs:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Initial state provided | Select data | Breadth-First Search | A\* with Heuristic one | A\* with Heuristic two |
| 706128345 | Number of moves | 23 | 23 | 23 |
| Nodes explored including root node | 81254 | 13085 | 1258 |
| Time (nanoseconds) | 445692117392 | 15650876856 | 242998704 |
| 841720365 | Number of moves | 17 | 17 | 17 |
| Nodes explored including root node | 11929 | 929 | 124 |
| Time (nanoseconds) | 7144479878 | 141379938 | 30985811 |
| 345012867 | Number of moves | 23 | 23 | 23 |
| Nodes explored including root node | 83305 | 13462 | 1622 |
| Time (nanoseconds) | 307432611222 | 15296518174 | 280652219 |
| 630852714 | Number of moves | 26 | 26 | 26 |
| Nodes explored including root node | 151850 | 37706 | 4281 |
| Time (nanoseconds) | 856659312772 | 165622214679 | 1300416321 |

In all three versions of the search, the number of moves required to solve the 8-puzzle was always the same. In all test runs, BFS explored the highest number of nodes, A\* search with heuristic one explored the second highest number of nodes and A\* search with heuristic two explored the least number of nodes. For all test cases, A\* search with heuristic two ran the fastest, A\* search with heuristic one ran the second fastest and BFS ran the slowest (with one execution even taking 14.2 minutes). We can rank the performance of each algorithm from best to worst in the following order: A\* search with heuristic two, A\* search with heuristic one and Breadth-First search.