Algorithm Course Work Report.

* On a 10x10 Grid where A is 0,0 and B is 9,9 using Manhattan the Elapsed time = 0.515 with a total cost of 18.
* On a 20x20 grid where A is 0,0 B is 19,19 using Manhattan the Elapsed time = 1.038 with a total cost of 38.
* On a 40x40 grid where A is 0,0 and B is 39,39 using Manhattan the Elapsed time = 2.041 with a total cost of 78.
* On a 80x80 grid where A is 0,0 and B is 79,79 using Manhattan the Elapsed time = 3.806 with a total cost of 158.

From analysis made I can tell the more I increase the size of my grid the more time it will spend to find path, For example on the 10x10 grid it took from A to B 0.515 which is pretty good, but the more I increase the grid the more time it need to find theq path. In the second example where my grid is 20x20 it took 1.038, in this case I double the amount of cells in the grid. I doubled again the size of grid 40x40 and now the time is 2.041, I doubled one more time and the time is 3.806.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Grid Size | Elapsed time | ratio | Log(ratio)base2 |  |
| 10x10 | 0.515 |  |  |  |
| 20x20 | 1.038 | ~2 | ~1 | (2^1=2) |
| 40x40 | 2.041 | ~2 | ~1 |  |
| 80x80 | 3.806 | ~2 | ~1 |  |
|  |  |  |  |  |
|  |  |  |  |  |

Conclusion: performance is linearithmic (O(N^1)), since the log of change converges to 1, e.g. the exponent of 2 in order to get 2.