University of Dhaka
Department of Computer Science and Engineering

CSE-4111: Artificial Intelligence Lab
Assignment 1
4th Year 1st Semester, 2020

Implementation and Performance comparison of Classical Search Algorithms Using N-Puzzle Problem

Submitted By:
Sree Sowmik Kumar Sarker
Roll: JN-054

Submitted To:
Dr. Muhammad Ibrahim
and
Dr. Md. Mosaddek Khan
February 27, 2020

Introduction

In this problem we have solved N-puzzle problem with various algorithms named: breadth first search (BFS), uniform cost search (UCS), depth limited search (DLS), iterative deepening depth first search (IDS), greedy best first search (GBFS), and A* search. To solve this problem we have used “Python” as programming language. For these algorithm, we have chosen the approaches according to the text book (Artificial Intelligence A Modern Approach) and also used many data structures. Typically, N-puzzle problem uses to compare the efficiency of classical search algorithms of Artificial Intelligence (AI). Here, the efficiency of these algorithms is measured by performance matrix that includes path cost to find the goal state from the initial state. Time required to run the algorithm and memory uses that is measured by node create count are also measured.

Result

The comparison of the result of those algorithms can easily be perceived from the graph plot. Here, the figures of the graph plot are given below-

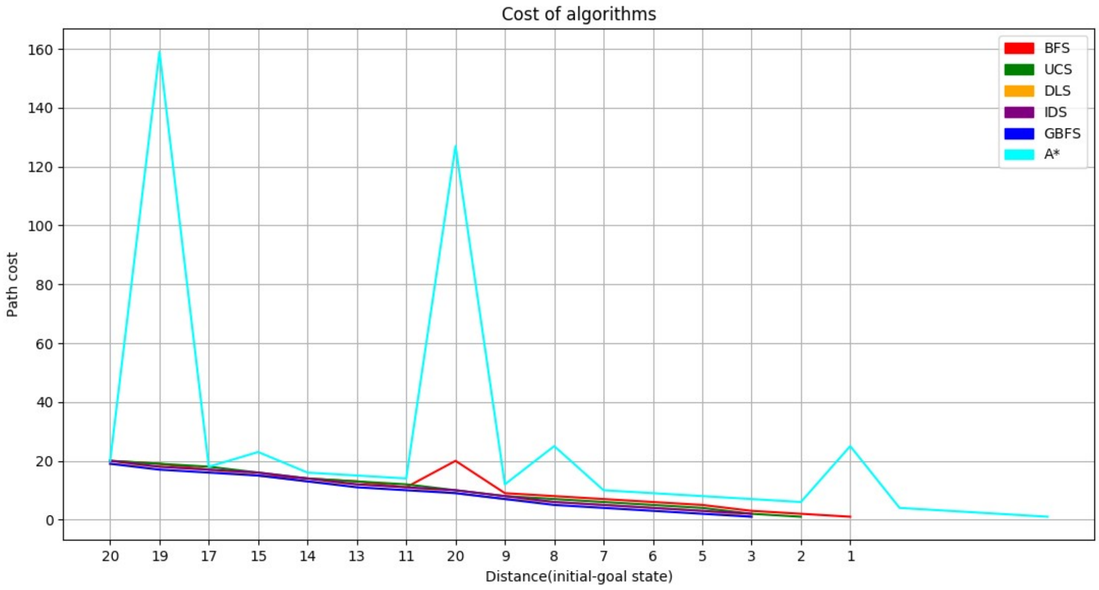

Figure:1

In figure 1,path cost of those different algorithms is compared. The result is plotted with respect to optimal result. For solving the puzzle in optimal move (various depth) the required move is calculated and plotted for all the six algorithms. From the result we can see,for BFS, UCS,IDS, A* graphs are same as they give optimal solutions. For DLS it varies near to the limit(with is set 20 for this result) and for GBFS it varies within a long range.

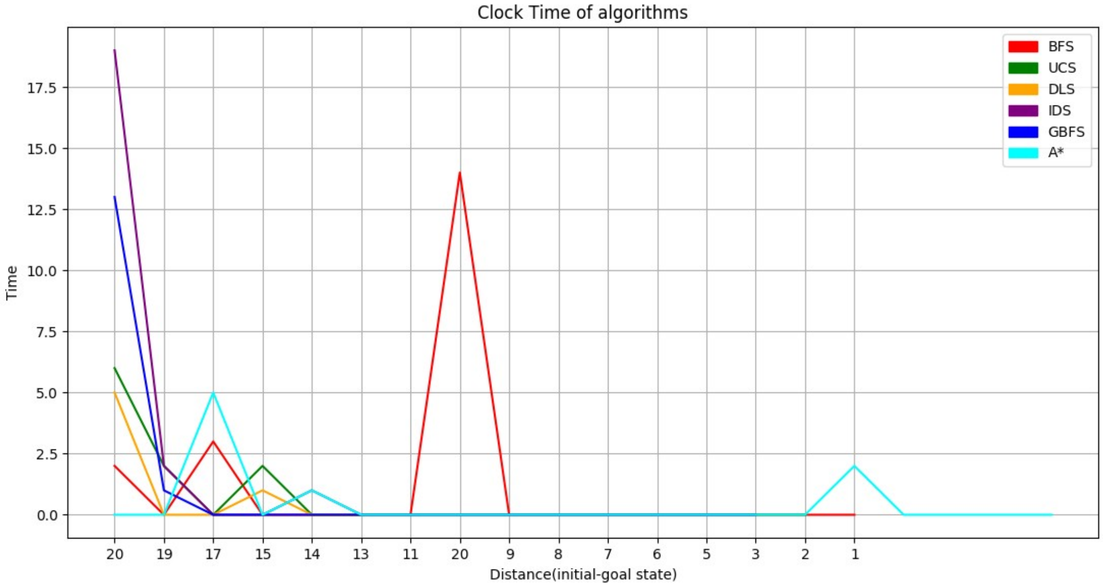

Figure: 2

In figure 2, we can see the comparison of clock cycle between those six algorithms which is measured by the total time required to complete a particular algorithm. For this measurement informed(heuristic) search strategies For GBFS and A* it requires much more smaller time then others. In these measurement GBFS and A* give the best results and DLS gives the worst result.

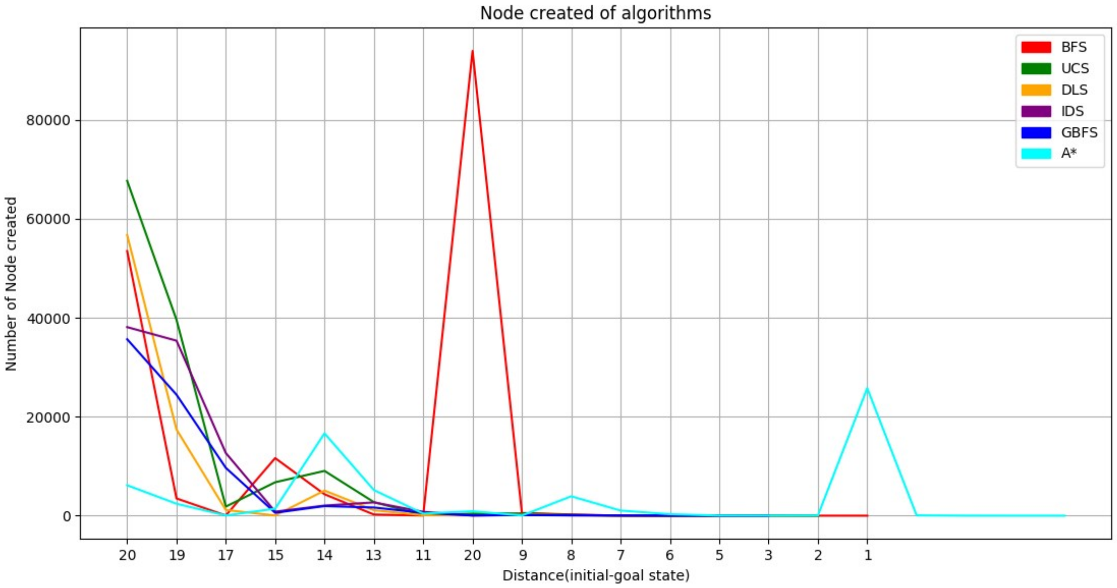

Figure: 3

In figure 3, memory performance is compared that is measured by the number of total node created to find goal state from initial state. That’s why it requires more space. In these measurement GBFS and A* give the best results and DLS gives the worst result.

Challenges

Tab error or indentation error is the most common problem I faced to much time during the implementation.

To implement these algorithms (BFS,UCS,DLS,IDS,GBFS and A*) , handling data structure was a big challenge. To implement different state of a node was also challenging. There was also many problems while working with offline mode to collect data and plot the graph.