ENPM 661 – Planning for Autonomous Robots

Project 1

Abhinav Modi

116346849

Problem Statement

Solve a 8-Piece Puzzle using Tree.

Compiling the Code

The .zip file consists of a C++ code “8puzzle.cpp” which consists the main() for running the code with the header files:

#include<vector>

#include<iostream>

#include<fstream>

#include<algorithm>

#include<string.h>

#include<stdio.h>

#include<stdlib.h>

#include"puzzle.h"

The required helper functions for solving the puzzle are provided in the “puzzle.h” file. To compile the code open the required directory in the Terminal and type:

$ g++ 8puzzle.cpp -o puzzle.exe

Running the Code

Once the code is compiled, run the following command for execution

$ ./puzzle.exe 2

* This will solve the puzzle for the initial state(column-wise): {7, 0, 8, 5, 3, 1, 4, 2, 6} and goal state(column-wise){1, 4, 7, 2, 5, 8, 3, 6, 0}.
* To execute the code for a different initial state(for eg., {1, 4, 7, 2, 5, 8, 3, 6, 0}), than provided run the previous command as follows:

$ ./puzzle.exe 3 147258306

* To execute the code for both a different initial state(for eg., {1, 4, 7, 2, 5, 8, 3, 6, 0}) and a different goal state than {1, 4, 7, 2, 5, 8, 3, 6, 0}, then use the above command as:

$ ./puzzle.exe 4 <start state> <goal state>

**Note**: There is also a bash script in the folder which runs the **matlab graphic** showing the puzzle being solved. The script is run **automatically** once the puzzle is solved by the code and the required text files generated. Once the code completes type “**exit**” on the terminal to close the program.

In addition to the check for Goal state, I have also included checks in the program to check for the uniqueness of each node and Solvability of the given initial condition if the end goal is fixed:{1, 4, 7, 2, 5, 8, 3, 6, 0}. Feel free to look through them.