CS 4613 **Fall 2021**
Project 1: 11-Puzzle Problem **E. K. Wong**

Total # points = 100.

Project Description: Implement the *A** search algorithm with *graph search* for solving the 11-puzzle problem as described below. Use $h(n) =$ *sum of Manhattan distances of the tiles from their goal positions* as heuristic function in the *A** search and also try *weighted A** search with *W* = 1.2 and 1.4*;* i.e., let $f(n) = g(n) + W \times h(n)$ for $W = 1.0, 1.2$ and 1.4. Your program will read in the initial and goal states from an input file and then generate an output file that contains the solution. Let *W* be a parameter that you can enter interactively using your keyboard. [Reminder: *graph search* does not allow repeated states.]

11-puzzle problem: On a 3 x 4 board (as shown below) there are 11 tiles numbered from 1 to 11 and a blank position. A tile can slide into the blank position if it is horizontally or vertically adjacent to the blank position. Given a start board configuration and a goal board configuration, find a move sequence with a minimum number of moves to reach the goal configuration from the start configuration.

1	2	3	4
10	11		5
9	8	7	6

You can work on the project by yourself or form a team of two students to work on the project. You can discuss with your classmates on how to do the project but everyone or every team is expected to write their own code and submit their own project.

Input and output file formats: Your program will read in the initial and goal states from a text file that contains 7 lines as shown in Figure 1 below. Lines 1 to 3 contain the tile pattern for the initial state and lines 5 to 7 contain the tile pattern for the goal state. Line 4 is a blank line. *n* and *m* are integers that range from 0 to 11. Integer 0 represents the blank position and integers 1 to 11 represent tile numbers.

Your program will produce an output file that contains 13 lines as shown in Figure 2 below. Lines 1 to 3 and lines 5 to 7 contain the tile patterns for the initial and goal states as given in the input file. Lines 4 and 8 are blank lines. Line 9 is the *W* value that you have entered interactively using the keyboard. Line 10 is the depth level *d* of the shallowest goal node as found by your search algorithm (assume the root node is at level 0.) Line 11 is the total number of nodes *N* generated in your tree (including the root node.) Line 12 contains the solution that you have found. The solution is a sequence of actions (from root node to goal node) represented by the A's in line 12, separated by blanks. Each A is a character from the set {L, R, U, D}, representing the left, right, up and down movements of the blank position. Line 13 contains the *f*(*n*) values of the nodes along the solution path from the root node to the goal node, separated by blanks. There should be *d* number of A values in line 12 and *d*+1 number of *f* values in line 13.

Testing your program: Three input files will be provided on *Brightspace* for you to test your program (All three input files are solvable; i.e., they have solutions.) For each input file, produce an output file for each of the weight value $W = 1.0, 1.2$ and 1.4.

Recommended languages: Python, C++/C and Java. If you would like to use a language other than these three, send me an email first.

CS 4613 **Fall 2021**
Project 1: 11-Puzzle Problem **E. K. Wong**

Submit on *Brightspace* by the due date:

1. Your source code file. Put comments in your source code to make it easier for someone else to read your program. Points will be taken off if you do not have comments in your source code.
2. The nine output files generated by your program for test input files 1 to 3. Name your output files output1a.txt, output1b.txt, output1c.text, output2a.txt, …, and output3c.txt.
3. A PDF file that contains instructions on how to run your program. If your program requires compilation, instructions on how to compile your program should also be provided. Also, copy and paste your output files and your source code onto the PDF file (to make it easier for us to grade your project.) This is in addition to the source code file and output files that you have to hand in separately, as described in (1) and (2) above.

If you work in a team of two, only one partner needs to submit but please write both partners' names on the source code and the PDF report.

```
n n n n
n n n n
n n n n

m m m m
m m m m
m m m m
```

Figure 1. Input file format (7 lines.)

```
**************************
n n n n
n n n n
n n n n

m m m m
m m m m
m m m m

W
d
N
A A A A A A A ……
f f f f f f f ……..
```

Figure 2. Output file format (13 lines.)