# CSCI 4450/8456 Artificial Intelligence

# Fall 2018: Homework 1

Instructor: Raj Dasgupta

**Given on: August 30, 2018 (Tuesday)**

**Due on: September 11, 2018 (Tuesday), 11:59 PM – submissions on loki**

**Maximum Points: 70**

Submission Instructions

You need to submit your source code that you have written for your assignment, and a README.txt file that gives the commands to compile and run your program for each question. Submissions are via loki only. DO NOT SUBMIT ON BLACKBOARD.

**It is your responsibility to ensure that your program compiles and runs properly on loki using the commands in your README file.**

1. Login to loki. You will be logged into your home directory
2. Create a sub-directory inside your home directory called CSCI-4450-DG-F18-A1 by using the command mkdir CSCI-4450-DG-F18-A1 from inside your home directory.
3. You must create the directory in step 2 right under your home directory and **with the exact name given above**. Otherwise, your homework will not be collected and graded.
4. Place your source code in the directory that you created in step 2 above before the assignment deadline.
5. Note that assignments will be automatically collected using a pre-scheduled script at the due date/time from the above directory names.

**Question 1 (30 points)**

A 5 puzzle is a 3 X 2 grid with 5 tiles numbered 1...5 and an empty tile as shown in the figures below. A tile next to the empty space can be moved (left, right, up or down but not diagonally) into the empty space. The objective is to have the arrangement of tiles shown in the goal state.

Random state

Goal state

4

5

3

1

2

1

4

2

5

3

For simplifying the problem the empty space is considered as tile 0. A state can be represented by the traversing the tiles row-wise (or row-major) starting from the top-left corner. So the state of the left hand figure is represented as 431502 while the goal state is 123450.

An action at each state would be one of L(left), R(right), U(up) or D(down). For example, the action left on the random state given above would move tile ‘2’ to its left into the empty space. Therefore, L(431502) would give the state 431520. Similarly, D(431502) (move tile 3 down) would give the state 401532. Notice that each state will have two possible actions if the empty space (tile 0) is at one of the corner squares and three possible actions if the empty space (tile 0) is in one of the middle squares. The cost of each action is the same.

Write a program to find a solution to the 5-puzzle starting from a given state (that you will input) using **iterative deepening search**(IDS). While inputting the start state you should keep in mind that some states are never possible or invalid because the goal state is unreachable from those states. For e.g. 132450 is an invalid state for the above start/goal state. For testing the correctness of your program you can use the start state given in the example above. The action sequence D-L-U-R-R-D-L-L-U leads to the goal state.

**Format of input and output**

Your program should first prompt the user to enter an initial state in the format XXXXXX, where each X corresponds to a digit between 0 and 5; e.g., 431502 (no spaces between digits). The output of the program should be printed on two lines –the first line should print out the goal state as “Goal State is 123450” or “Goal State is 012345”. On the second line it should print out the action sequence, each move separated by a ‘-‘ (e.g., Solution: D-R-D-D) found by your algorithm.

**Question 2 (40 points: 20 points each subpart)**

****

The 8-puzzle is comprises of tiles numbered 0..8 in a 3 X 3 grid, as shown alongside. Formulate the 8 puzzle problem in a manner similar to the 5-puzzle problem given in question 1, that is, define a representation of a state, the possible actions from each state, the goal test and the step cost.

1. Write a program to search for the goal state shown in Figure 3.4 (right), starting from a random state in the 8 puzzle using A\* search with the heuristic function h(n) = number of tiles that are not in the correct place in state ‘n’.
2. Define the Manhattan distance of a tile M(tile) = Number of moves needed to get the tile from its position in the current state to its position in the goal state. Write a program to search for a goal state starting from a random state in the 8 puzzle using A\* search using the heuristic function h(n) = Sum of Manhattan distances of each tile in state ‘n’.

For parts a) and b) assume that g(n) is the cost from the start state to state ‘n’. Use the same format of input and output given in Question 1.