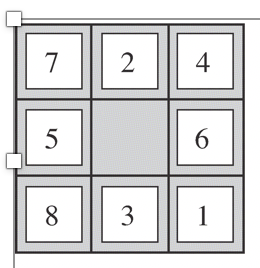
# **CPSC 4420/6420: Artificial intelligence**

# Assignment 1- Due: Sep 3, 2021 @11:59 pm

# Name: Thomas Griffith



For the above puzzle shown here, develop a Python program that

1. Lists all states [No need to submit the output (the list of states) for this part, and just submit the code, since the list will be very long !!!]

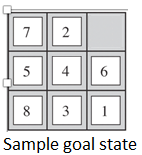
It works in code just type state and then hit a

1. Gets the current state and the action (moving up:1, down:2, left:3, right:4) as input, and returns the resulting state. Represent the blank spot with “0” and use one of the following naming formats for states

* Represent each state with a sequence of numbers from left to right and top to bottom. Ex. Use 7-2-4-5-0-6-8-3-1 for the state shown above
* Represent each state by a 9-digit integer number. Like show the above state by 724506831

Ex: Input (Current state: 724506831, Action: 3) should give output state: 724056831

It works in code just type state and then hit b

1. Suppose that the goal is to arrange the numbers so that the resulting 3-digit numbers created by each row are divisible by 3. For instance, 7-2-0-5-4-6-8-3-1 is a goal state because 720, 546, and 831 are divisible by 3. Write a program that prompts the user to receive an arbitrary initial state, and then performs random actions to reach the goal state. Show the sequence of actions and the sequence of states.

It works in code just type state and then hit c

1. Suppose that the goal is arranging the blocks in numerical order as shown below

Develop a Breadth First Search (BFS) algorithm and show the results. Present the sequence of states and moves, starting from the initial state. How many moves (actions) did it take to reach the goal state?



It works in code just type state and then hit d

Also, It takes a really really long time so I ran it with out the path using state 724506831 and got

5total Steps 88308

Done [final state, steps for plan]

['012345678', 48404]

So if you need to comment out path to see if mine works please do so

1. Repeat part (D) using a Depth-First Search (DFS). How many moves (actions) did it take to reach the goal state?

Which algorithm found the solution with less moves? Explain your observation.

It works in code just type state and then hit e

Also, It takes a really really long time so I ran it with out the path using state 724506831 and got

total Steps 171711

done

['012345678', 26]So if you need to comment out path to see if mine works please do so

1. Repeat Part (D), if the goal is ordering the numbers clockwise around the blank space, with the given initial state, as shown below.

It works in code just type state and then hit f



1. Implement a unit cost search (UCS), if the goal is achieving the final state in part F from an arbitrary initial state, if we have the following costs for different moves

E1) All moves have unit cost

E2) Up (Cost=1), Down (Cost=1) Left (Cost=2) Right (Cost=0.5)

Present the sequences of moves and states for each option. How many actions are used to achieve the solution for each option? Explain your observation.

Did not finish it in time, may submit it a day late if I get it by then