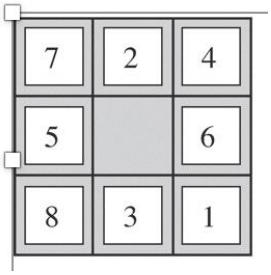
CPSC 4420/6420: ARTIFICIAL INTELLIGENCE

ASSIGNMENT 1- DUE: SEP 14, 2023 @04:59 PM

NAME:



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

(A) [5 pts] Lists (displays on screen) all states. Present blank with digit "0". [No need to submit the list of states since the list will be very long !!!]

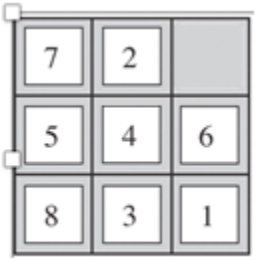
(B) [5 pts] Prints 10 randomly selected states so that no odd numbers are neighbors (like 014523876) in the state.

(C) [10 pts] Gets the current state and the action (moving up:1, down:2, left:3, right:4) as input, and returns the resulting state. You can 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

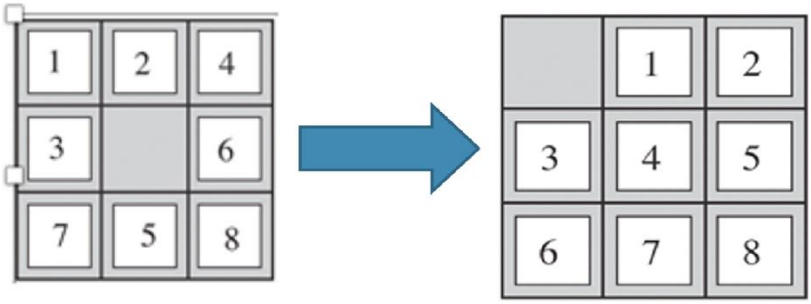
Ex: Input (Current state: 724506831, Action: 3) should give output state: 724056831

(D) [10 pts] 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 potential 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.



Sample goal state

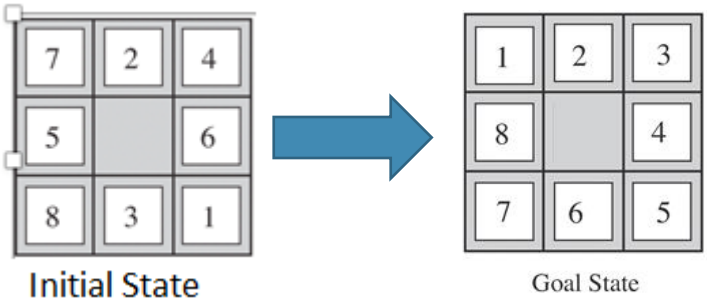
(E) [20 pts] Suppose that the goal is to arrange the blocks in numerical order as shown below. Develop a Breadth First Search (BFS) algorithm and show the results. Present the sequence of states and moves, starting from the initial state. How many moves (actions) did it take to reach the goal state?



Initial State Goal State

(F) [10 pts] 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 fewer moves? Explain your observation.

(G) [10 pts] Repeat Part (D), if the goal is ordering the numbers clockwise around the blank space, with the given initial state, as shown below.



(H) Implement a Uniform 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

[15 pts] G1) All moves have a unit cost
[15 pts] G2) Up (Cost=1.5), Down (Cost=0.5) Left (Cost=1) Right (Cost=2)
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.