Ahsanullah University of Science and Technology (AUST)
Department of Computer Science and Engineering

Lab Report

Course No   : CSE4108

Course Title   : Artificial Intelligence Lab

Assignment No   : 02

Date of Submission   : 28-11-2022

Submitted To   : Mr. Raihan Tanvir ,Mr Shimul Paul

Submitted By:

Name   : Debopriya Deb Roy

Student ID : 190104065

Section   : B

Lab Group   : B1

Ques 1: **Program in Python to find the length of a path between two vertices of a directed weighted graph.**

Answer:

```python
def Solve_ques1():
    def findPath(start, end, cost=0):
        if start == end:
            # If the start and end are the same, print the cost
            print(str(cost) + ' ')
        else:
            # Check each edge in the weighted list
            for (i, j, w) in weightedList:
                if i == start:
                    # If the edge starts at the current start point,
                    # call findPath() recursively with the end point of the edge
                    # and the updated cost
                    findPath(j, end, cost + w)

    # Here I have defined the edges as a list of tuples
    # Each tuple contains the start and end points, and the weight of the edge
    weightedList = [('A', 'B', 10), ('A', 'C', 5),
                    ('B', 'D', 20), ('C', 'D', 15),]
    start_point = str(input('Enter Starting point: '))
    end_point = str(input('Enter Ending point: '))
    print('The length of findPath is: ')
    findPath(start_point, end_point)
Solve_ques1()
```

```
ddroy@DDR13Laptop MINGW64 ~/Desktop/Ai lab exercises (main)
$ C:/Users/ddroy/AppData/Local/Programs/Python/Python310/python.exe "c:/Users/ddroy/Desktop/Ai lab exercises/test.py"
Enter Starting point: A
Enter Ending point: D
The length of path is:
30
20
```

Ques 2 **Program in Python to find h2 (heuristic value for 8-Puzzle problem) and h3 (heuristic value for 8-Queen problem).**

Answer:

```python
def Solve_ques2b():

    # goal_state is a touple list to store the Goal State of the 8-puzzle problem.
    goal_state = [(1, 1, 1), (2, 1, 2), (3, 1, 3), (4, 2, 3),
            (5, 3, 3), (6, 3, 2), (7, 3, 1), (8, 2, 1)]
    # current_state is a touple list to store the Current State of the 8-puzzle problem
    current_state = [(1, 1, 2), (2, 1, 3), (3, 2, 1), (4, 2, 3),
            (5, 3, 3), (6, 2, 2), (7, 3, 2), (8, 1, 1)]

    # i is a variable to do the iteration in while loop which is initiated with the value 0.
    # h is a variable to store the manhattan distance which is initiated with the value 0.
    i, h = 0, 0
    while (i <= 7):
        if ((goal_state[i][1] != current_state[i][1]) or (goal_state[i][2] != current_state[i][2])):
            h += abs(goal_state[i][1] - current_state[i][1]) + \
                abs(goal_state[i][2] - current_state[i][2])
        i = i+1

    print("Manhattan distance is: ", h)

def Solve_ques2a():
    def count_collisions(position):
        collisions = 0

        # Count collisions in rows and storing it in collisions variable
        for i in range(8):
            for j in range(i + 1, 8):
                if position[i] == position[j]:
```

collisions += 1

    # Count collisions in diagonals by checking if the distance of x-axis and y-axis are same or
not.
    for i in range(8):
        for j in range(i + 1, 8):
            if abs(position[i] - position[j]) == abs(i - j):
                collisions += 1
        return collisions

    position = [6, 1, 5, 7, 4, 3, 8, 1]
    print('Total Collisions in 8-Queens: ', count_collisions(position))

Solve_ques2a()


```
ddroy@DDR13Laptop MINGW64 ~/Desktop/Ai lab exercises (main)
$ C:/Users/ddroy/AppData/Local/Programs/Python/Python310/python.exe "c:/Users/ddroy/Desktop/Ai lab exercises/190104065_Assignment2.py"
Manhattan distance is:  8
```


Solve_ques2b()

```
ddroy@DDR13Laptop MINGW64 ~/Desktop/Ai lab exercises (main)
$ C:/Users/ddroy/AppData/Local/Programs/Python/Python310/python.exe "c:/Users/ddroy/Desktop/Ai lab exercises/190104065_Assignment2.py"
Total Collisions in 8-Queens:  5
```