

A.I

LAB TASK 3

Name: Syed Muzzamil Waseem

SID: 11067

CID: 110089

Date: 7 July 2022

QUESTION:

Implement Hill Climbing Algorithm that shows traversing down the nodes as per their heuristic value.

CODE:

```
from random import randint

tsp = [
    [923, 529, 297, 693, 907, 542, 693, 401, 280, 785],
    [272, 470, 988, 509, 592, 913, 831, 740, 858, 451]
]

def randomSolution(tsp):
    cities = list(range(len(tsp)))
    solution = []
    for i in range (len(tsp)):
        randomCity = cities[randint(0, len(cities) -1 )]
        solution.append(randomCity)
        cities.remove(randomCity)
    return solution

def routeLength(tsp, solution):
    routeLength = 0
    for i in range(len(solution)):
        routeLength += tsp[solution[i-1]][solution[i]]
    return routeLength

def getNeighbours(solution):
    neighbours = []
    for i in range(len(solution)):
        for j in range(i + 1, len(solution)):
            neighbour = solution.copy()
```

Submitted to: Sir Ramzan Ali

```

        neighbour[i] = solution [j]
        neighbour[j] = solution[i]
        neighbours.append(neighbour)
    return neighbours

def getBestNeighbour(tsp, neighbours):
    bestRouteLength = routeLength(tsp, neighbours[0])
    bestNeighbour = neighbours[0]
    for neighbour in neighbours:
        currentRouteLength = routeLength(tsp, neighbour)
        if currentRouteLength < bestRouteLength:
            bestRouteLength = currentRouteLength
            bestNeighbour = neighbour
    return bestNeighbour, bestRouteLength

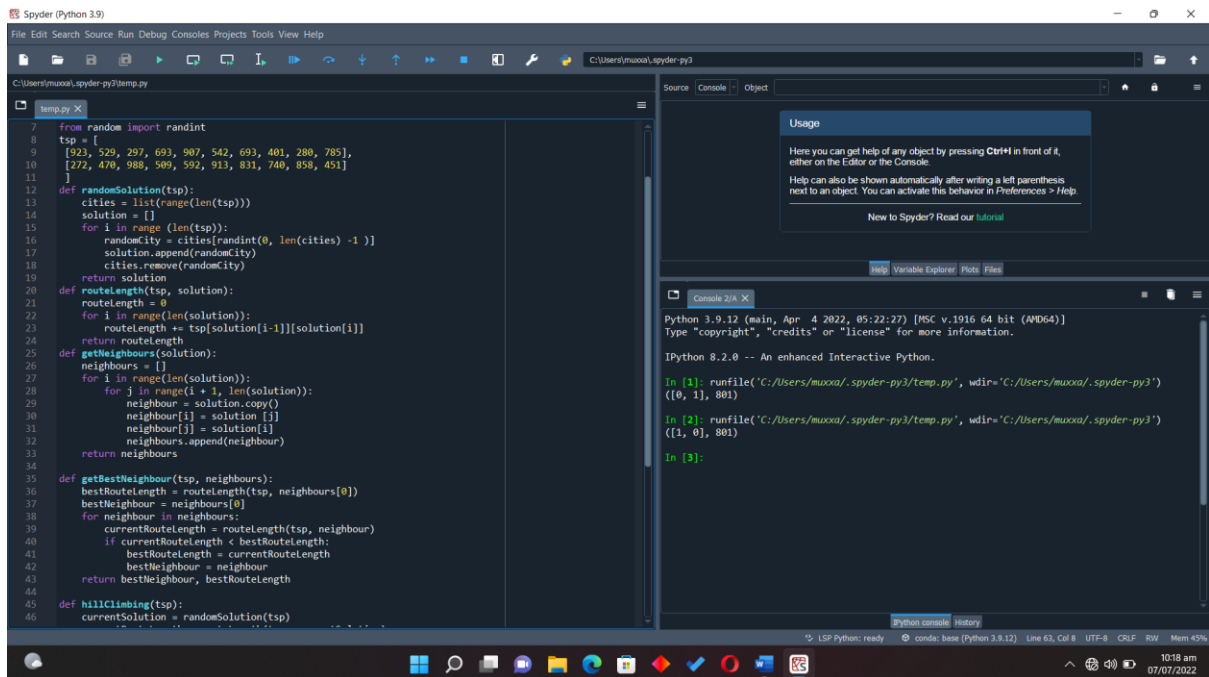
def hillClimbing(tsp):
    currentSolution = randomSolution(tsp)
    currentRouteLength = routeLength(tsp, currentSolution)
    neighbours = getNeighbours(currentSolution)
    bestNeighbour, bestNeighbourRouteLength = getBestNeighbour(tsp, neighbours)
    while bestNeighbourRouteLength < currentRouteLength:
        currentSolution = bestNeighbour
        currentRouteLength = bestNeighbourRouteLength
        neighbours = getNeighbours(currentSolution)
        bestNeighbour, bestNeighbourRouteLength = getBestNeighbour(tsp, neighbours)
    return currentSolution, currentRouteLength

def main():
    tsp = [
        [923, 529, 297, 693, 907, 542, 693, 401, 280, 785],
        [272, 470, 988, 509, 592, 913, 831, 740, 858, 451]
    ]
    print(hillClimbing(tsp))

if __name__ == "__main__":
    main()

```

OUTPUT:



The screenshot shows the Spyder Python IDE interface. The left pane displays a Python script named `temp.py` with the following code:

```
7 from random import randint
8 tsp = [
9     [922, 529, 297, 693, 907, 542, 693, 401, 280, 785],
10    [272, 470, 988, 509, 592, 913, 831, 740, 858, 451]
11 ]
12 def randomSolution(tsp):
13     cities = list(range(len(tsp)))
14     solution = []
15     for i in range(len(tsp)):
16         randomCity = cities[randint(0, len(cities) - 1)]
17         solution.append(randomCity)
18         cities.remove(randomCity)
19     return solution
20 def routeLength(tsp, solution):
21     routeLength = 0
22     for i in range(len(solution)):
23         routeLength += tsp[solution[i-1]][solution[i]]
24     return routeLength
25 def getNeighbours(solution):
26     neighbours = []
27     for i in range(len(solution)):
28         for j in range(i + 1, len(solution)):
29             neighbour = solution.copy()
30             neighbour[i] = solution[j]
31             neighbour[j] = solution[i]
32             neighbours.append(neighbour)
33     return neighbours
34 def getBestNeighbour(tsp, neighbours):
35     bestRouteLength = routeLength(tsp, neighbours[0])
36     bestNeighbour = neighbours[0]
37     for neighbour in neighbours:
38         currentRouteLength = routeLength(tsp, neighbour)
39         if currentRouteLength < bestRouteLength:
40             bestRouteLength = currentRouteLength
41             bestNeighbour = neighbour
42     return bestNeighbour, bestRouteLength
43 def hillClimbing(tsp):
44     currentSolution = randomSolution(tsp)
```

The right pane shows the IPython console with the following output:

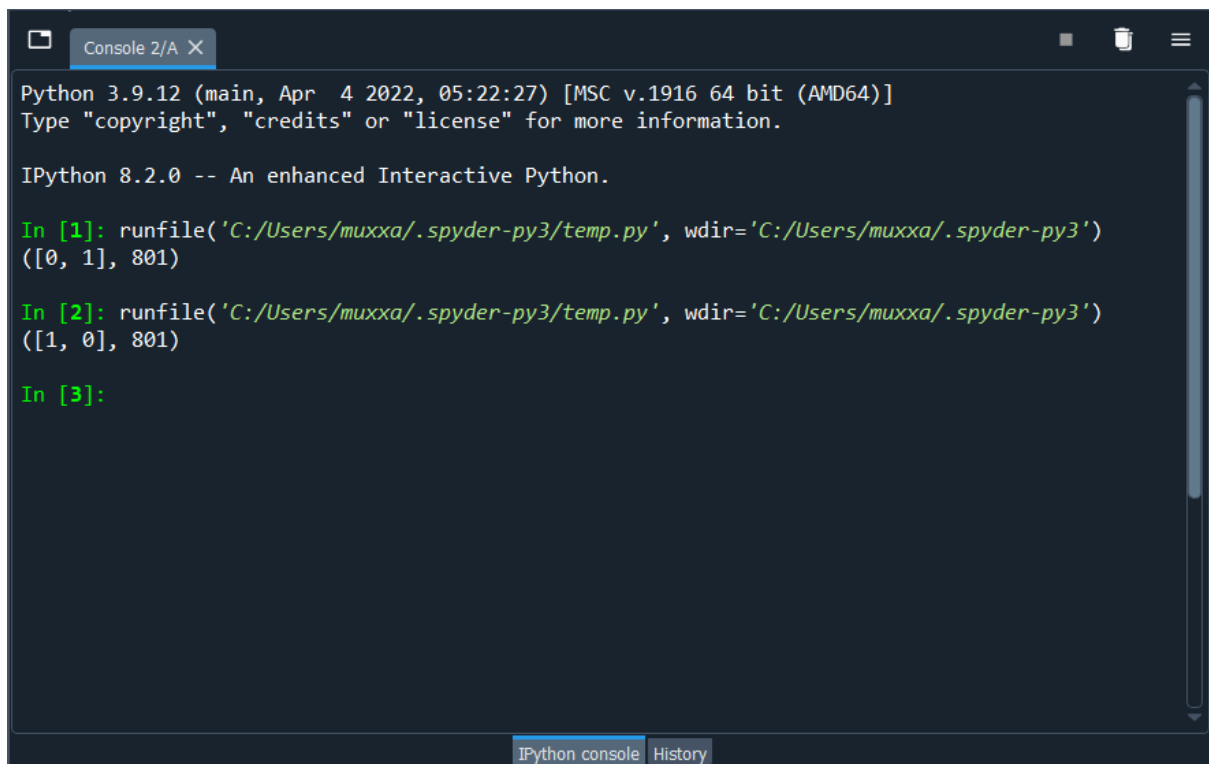
```
Python 3.9.12 (main, Apr 4 2022, 05:22:27) [MSC v.1916 64 bit (AMD64)]
Type "copyright", "credits" or "license" for more information.

IPython 8.2.0 -- An enhanced Interactive Python.

In [1]: runfile('C:/Users/muxxa/.spyder-py3/temp.py', wdir='C:/Users/muxxa/.spyder-py3')
([0, 1], 801)

In [2]: runfile('C:/Users/muxxa/.spyder-py3/temp.py', wdir='C:/Users/muxxa/.spyder-py3')
([1, 0], 801)

In [3]:
```



This is a close-up view of the IPython console window. It displays the same output as the previous screenshot, showing the execution of the `temp.py` script twice. The first execution returns `([0, 1], 801)` and the second returns `([1, 0], 801)`. The console title bar indicates it is 'Console 2/A'.

Submitted to: Sir Ramzan Ali