

```

import random

def randomSolution(tsp):
    cities=list(range(len(tsp)))
    solution=[]
    for i in range(len(tsp)):
        randomCity=cities[random.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()
            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

tsp = [
    [0,400,500,300],
    [400,0,300,500],
    [500,300,0,400],
    [300,500,400,0]
]

```

```
print(hillClimbing(tsp))  
  
([2, 3, 0, 1], 1400)
```

```
tsp = [[0,1,2,3,4,5,6,7,8,9],  
       [1,0,2,3,4,5,6,7,8,8],  
       [2,2,0,3,4,5,6,7,7,7],  
       [3,3,3,0,4,5,6,6,6,6],  
       [4,4,4,4,0,5,5,5,5,5],  
       [5,5,5,5,5,0,4,4,4,4],  
       [6,6,6,6,5,4,0,3,3,3],  
       [7,7,7,6,5,4,3,0,2,2],  
       [8,8,7,6,5,4,3,2,0,1],  
       [9,8,7,6,5,4,3,2,1,0]]  
print(hillClimbing(tsp))
```

```
([2, 0, 1, 3, 4, 6, 8, 9, 7, 5], 30)
```