

```

from random import *

import random

import numpy

import copy


countCities = 20;

# 2D Array
cities = numpy.zeros(shape=(20,20))

# tour
hypothesis = [int]*countCities
visitedCities = []
saveState = []


threshold = 2
lastFitness = 0
trials = 0
cityIndex = 1


# calculates fitness based on the difference between the distances
def getFitness(fitness, hypothesis, saveState, cities):
    oldDistance = getDistance(cities, saveState)
    print("Old Distance ",oldDistance,"km")
    print("")
    newDistance = getDistance(cities, hypothesis)
    print("New Distance ",newDistance,"km")
    print("")
    if(oldDistance > newDistance):
        fitness += 1
    elif(oldDistance < newDistance):
        fitness -= 1

```

```
return fitness
```

```
# choose random City at position cityIndex
```

```
def doRandomStep():
```

```
    global visitedCities
```

```
    global saveState
```

```
    global hypothesis
```

```
    if(len(visitedCities) >= countCities):
```

```
        visitedCities.clear()
```

```
        visitedCities.append(0)
```

```
    randomNumbers = list(set(saveState) - set(visitedCities))
```

```
    randomStep = random.choice(randomNumbers)
```

```
    visitedCities.append(randomStep)
```

```
    hypothesis.remove(randomStep)
```

```
    hypothesis.insert(cityIndex,randomStep)
```

```
# next city
```

```
def increment():
```

```
    global cityIndex
```

```
    global visitedCities
```

```
    if (cityIndex < countCities - 2):
```

```
        cityIndex += 1
```

```
    else:
```

```
        visitedCities.clear()
```

```
        cityIndex = 1
```

```
# calculates distance from tour
```

```
def getDistance(cities, hypothesis):
```

```
    distance = 0
```

```
    for i in range(countCities):
```

```
        if (i < countCities-1):
```

```

        distance += cities[hypothesis[i]][hypothesis[i+1]]

        print("[",hypothesis[i],"]",distance,"km ",end="")
    else:
        print("[",hypothesis[i],"]")

    return distance

if __name__ == '__main__':

    for i in range(countCities):
        hypothesis[i] = i
        for j in range(countCities):
            if (j > i):
                cities[i][j] = randint(1,100)
            elif(j < i):
                cities[i][j] = cities[j][i]

    print("=== START ===");
    while(lastFitness < threshold):

        print("_____")
        saveState = copy.deepcopy(hypothesis)
        doRandomStep()
        currentFitness = getFitness(lastFitness, hypothesis, saveState, cities)
        print("Old fitness ",lastFitness)
        print("Current fitness ",currentFitness)

        if (currentFitness > lastFitness):
            lastFitness = currentFitness
        elif(currentFitness < lastFitness):
            hypothesis = copy.deepcopy(saveState)

```

```
if(trials < 3):  
    increment()  
else:  
    trials = 0  
visitedCities.append(saveState[cityIndex])
```