

AI Project 1

Eric Dockery
Computer Engineering and Computer Science
Speed School of Engineering
University of Louisville, USA
eadock01@louisville.edu

Introduction:

For this assignment, we are asked to solve the traveling salesman problem using a brute force method. This task asked us to generate all permutations of a set of numbers, trace a Hamiltonian path through an undirected graph, and to find the minimum cost solution to a traveling salesperson problem. This assignment is to show the exponential growth of a traveling salesperson problem and the unrealistic approach to solve these problems using brute force computation

Approach:

The data sets given to test our code had an extra 6 lines that needed to be removed and stripped. My solution prompted the user for the tsp file then striped the file and sorted the information that was needed for the solution by storing that data in an array (list in python.) Once the data was sorted the program called a permutationPrep function.

This permutationPrep function takes the array of values and preforms the permutation on each Content in the Permutation Function. The Permutation program takes the list and removes one element at a time from the list and preforms the necessary switches of the list in order to generate the last value of the permutation list. The program then runs a for loop for each end value and moves it throughout the list with recursion. The list is solved by picking the first element in the list removing it and calling permutation for the rest of the elements. This pattern continues until there is only one element left in the list and it adds that last element the beginning in a reversed order. This happens with each element being placed first in the sequence, then second, then third, etc. This goes through each elements permutation in the function one at a time solving the problem with recursion. Due to the exponential growth potential of the program the function removes each instance in memory as it finishes with each group of permutations.

When each permutation is calculated it is returned to the permutationPrep for loop where the distance formula is calculated for each permutation. That information is stored in the testcost variable. This variable is tested against a mincost variable that is given the arbitrary value of 1111111111111111. If the testcost is less than the mincost then the mincost is set to testcost and that p value is stored in an array. After that instance in the for loop is finished the p value is deleted re-allocating more memory.

At the end of the permutationPrep function the program calls a function called plotPoints where the values given are stored on a GUI and the shortest path is drawn along with the shortest distance calculation.

3. Results:

3.1 Data:

The data that was used for this assignment was generated using Concorde. The format for the data was:

```
NAME: concorde4
TYPE: TSP
COMMENT: Generated by CCutil_writetsplib
COMMENT: Write called for by Concorde GUI
DIMENSION: 4
EDGE_WEIGHT_TYPE: EUC_2D
NODE_COORD_SECTION
1 87.951292 2.658162
2 33.466597 66.682943
3 91.778314 53.807184
4 20.526749 47.633290
```

3.2 Results:

```
>>>
```

What is the file name? Random4.tsp

The shortest Path is this permutation:

```
[[ '1', '87.951292', '2.658162'], [ '4', '20.526749', '47.633290'], [ '2', '33.466597', '66.682943'], [ '3', '91.778314', '53.807184'], [ '1', '87.951292', '2.658162']]
```

The minimum cost is:

215.086

```
>>> ===== RESTART
=====
```

```
>>>
```

What is the file name? Random5.tsp

The shortest Path is this permutation:

```
[[ '1', '31.045752', '75.527426'], [ '2', '42.565359', '77.004219'], [ '5', '53.104575', '90.717300'], [ '3', '64.705882', '76.160338'], [ '4', '50.735294', '33.333333'], [ '1', '31.045752', '75.527426']]
```

The minimum cost is:

139.134

```
>>> ===== RESTART
=====
```

```
>>>
```

What is the file name? Random6.tsp

The shortest Path is this permutation:

```
[[ '1', '30.147059', '79.746835'], [ '2', '45.996732', '78.270042'], [ '3', '61.192810', '78.270042'], [ '4', '60.049020', '48.734177'], [ '5', '43.545752', '49.156118'], [ '6', '30.228758', '51.476793'], [ '1', '30.147059', '79.746835']]
```

The minimum cost is:

118.969

```
>>> ===== RESTART
=====
```

```
>>>
```

What is the file name? Random7.tsp

The shortest Path is this permutation:

```
[[ '1', '29.656863', '81.223629'], [ '2', '29.003268', '69.831224'], [ '7', '27.042484', '59.071730'], [ '3', '21.486928', '67.088608'], [ '6', '25.571895', '72.573840'], [ '5', '24.019608', '78.691983'], [ '4', '20.915033', '86.497890'], [ '1', '29.656863', '81.223629']]
```

The minimum cost is:

63.863

>>> ===== RESTART
=====

>>>

What is the file name? Random8.tsp

The shortest Path is this permutation:

[['4', '20.526749', '47.633290'], ['5', '9.006012', '81.185339'], ['2', '33.466597', '66.682943'], ['3',
'91.778314', '53.807184'], ['7', '77.181310', '31.922361'], ['1', '87.951292', '2.658162'], ['6',
'20.032350', '2.761925'], ['8', '41.059603', '32.578509'], ['4', '20.526749', '47.633290']]

The minimum cost is:

310.982

>>> ===== RESTART
=====

>>>

What is the file name? Random9.tsp

The shortest Path is this permutation:

[['6', '33.373494', '45.811518'], ['7', '30.481928', '55.497382'], ['1', '21.024096', '75.654450'], ['8',
'38.192771', '78.010471'], ['4', '37.048193', '74.345550'], ['9', '42.168675', '67.931937'], ['2',
'57.771084', '73.298429'], ['5', '52.108434', '47.905759'], ['3', '41.566265', '37.303665'], ['6',
'33.373494', '45.811518'],]

The minimum cost is:

131.028

>>> ===== RESTART
=====

>>>

What is the file name? Random10.tsp

The shortest Path is this permutation:

[['1', '22.549020', '89.029536'], ['2', '23.039216', '81.434599'], ['7', '25.245098', '67.721519'], ['6',
'23.774510', '59.704641'], ['8', '30.065359', '66.244726'], ['5', '38.071895', '60.759494'], ['9',
'36.029412', '70.886076'], ['10', '49.264706', '71.940928'], ['4', '40.277778', '80.379747'], ['3',
'30.392157', '79.324895'], ['1', '22.549020', '89.029536'],]

The minimum cost is:

106.786

>>> ===== RESTART
=====

>>>

What is the file name? Random11.tsp

The shortest Path is this permutation:

[['1', '20.016340', '84.599156'], ['6', '39.869281', '73.839662'], ['10', '53.104575', '68.354430'],
['11', '63.888889', '61.814346'], ['8', '89.460784', '83.966245'], ['9', '78.022876', '29.746835'], ['7',
'48.202614', '38.396624'], ['5', '27.369281', '54.008439'], ['3', '14.215686', '46.835443'], ['4',
'11.192810', '63.502110'], ['2', '20.343137', '72.573840'], ['1', '20.016340', '84.599156']]

The minimum cost is:

252.684

>>> ===== RESTART
=====

>>>What is the file name? Random12.tsp

[['1', '25.816993', '74.261603'], ['8', '28.676471', '76.160338'], ['2', '32.352941', '77.426160'], ['3',
'34.477124', '73.839662'], ['12', '29.820261', '65.822785'], ['4', '35.702614', '59.282700'], ['9',
'31.862745', '55.274262'], ['5', '29.084967', '52.109705'], ['10', '25.571895', '53.164557'], ['6',
'21.650327', '58.016878'], ['7', '22.467320', '64.556962'], ['11', '24.183007', '69.198312'], ['1',
'25.816993', '74.261603'],]

The minimum cost is:

66.804

>>>

4. Discussion:

These results ran very poorly. Due to the exponential growth of the program as you increase the points to be plotted, the program significantly declined in performance.

Run Times:

Random4.tsp : less than 1 second

Random5.tsp: less than 1 second

Random6.tsp: less than 1 second

Random7.tsp: approximately 1 seconds

Random8.tsp: approximately 2 seconds

Random9.tsp: approximately 17 seconds

Random10.tsp: approximately 148 seconds

Random11.tsp: approximately 46 minutes

Random12.tsp: Over 6 hours

5. References:

For this assignment I based my permutation strongly on the `itertools.permutation` command in python.

<https://docs.python.org/2/library/itertools.html#itertools.permutations>