Hamiltonian Path Application

Travelling Salesman Problem

Project By:
Harshit Khariwal - 15D070026
Anmol Gupta - 15D070025
Kavya Prudhvi - 15D070014
Mohak Sahu - 15D070047

Problem Statement

A salesman wants to visit a number of cities which are connected. The distance between any two cities is known. Which is the order in which the salesman should visit the cities so that he minimizes the total distance we walks? The problem is to find Hamiltonian path with minimum weight.

An equivalent formulation in terms of graph theory is: Given a complete weighted graph (where the vertices would represent the cities, the edges would represent the roads, and the weights would be the cost or distance of that road), find a Hamiltonian path with the least weight.

Algorithm

Insight into algorithm for finding a Hamiltonian Path:

A Hamiltonian Path in a graph having N vertices is nothing but a permutation of the vertices of the graph [v1, v2, v3, ...vN-1, vN] , such that there is an edge between vi and vi+1 where $1 \leq i \leq N-1$. So it can be checked for all permutations of the vertices whether any of them represents a Hamiltonian Path or not.

1.We have introduced weighted directed edges between the cities.

2.We have implemented the same in PyGame and the result of which is shown in the video in the zip file.

Basic outline of Code

The program takes a graph as an input in the form a dictionary whose keys are the vertices and the values are the corresponding neighboring vertices.

It identifies the edges in the graph, and subsequently asks for the respective weights of each.

On inputting the edges, it finds all the possible Hamiltonian paths in the code, and then chooses the smallest one of them.

Steps to run the code

1. Run the code in command prompt by writing: python “AdvancedTSP.py”.
   It asks the user to input a dictionary in ‘vertices : [all its neighbouring vertices] form’.

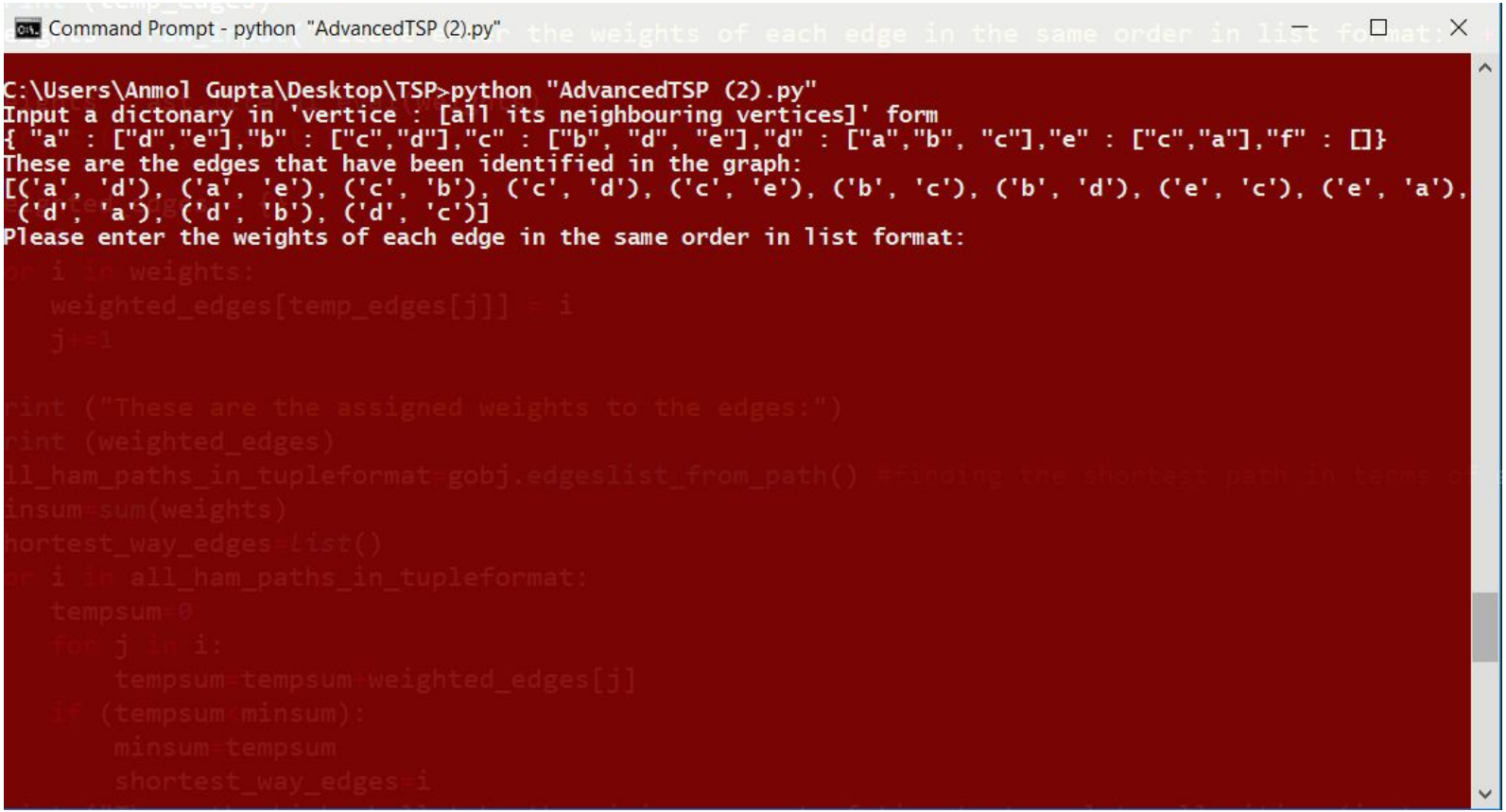
2. Enter the Sample input of dictionary. An example of the same is:
   a. { "a" : ["d","e"],"b" : ["c","d"],"c" : ["b", "d", "e"],"d" : ["a","b", "c"],"e" : ["c","a"]}

3. It identifies and display the edges in the graph, and subsequently asks for the respective weights of each. Care has to be taken to enter the weights of edges (i,j) and (j,i) as same, otherwise it will return a faulty output.
   A sample input of weights (corresponding to the sample input of vertices provided above) is:
   [1,2,3,4,5,3,6,5,2,1,6,4]

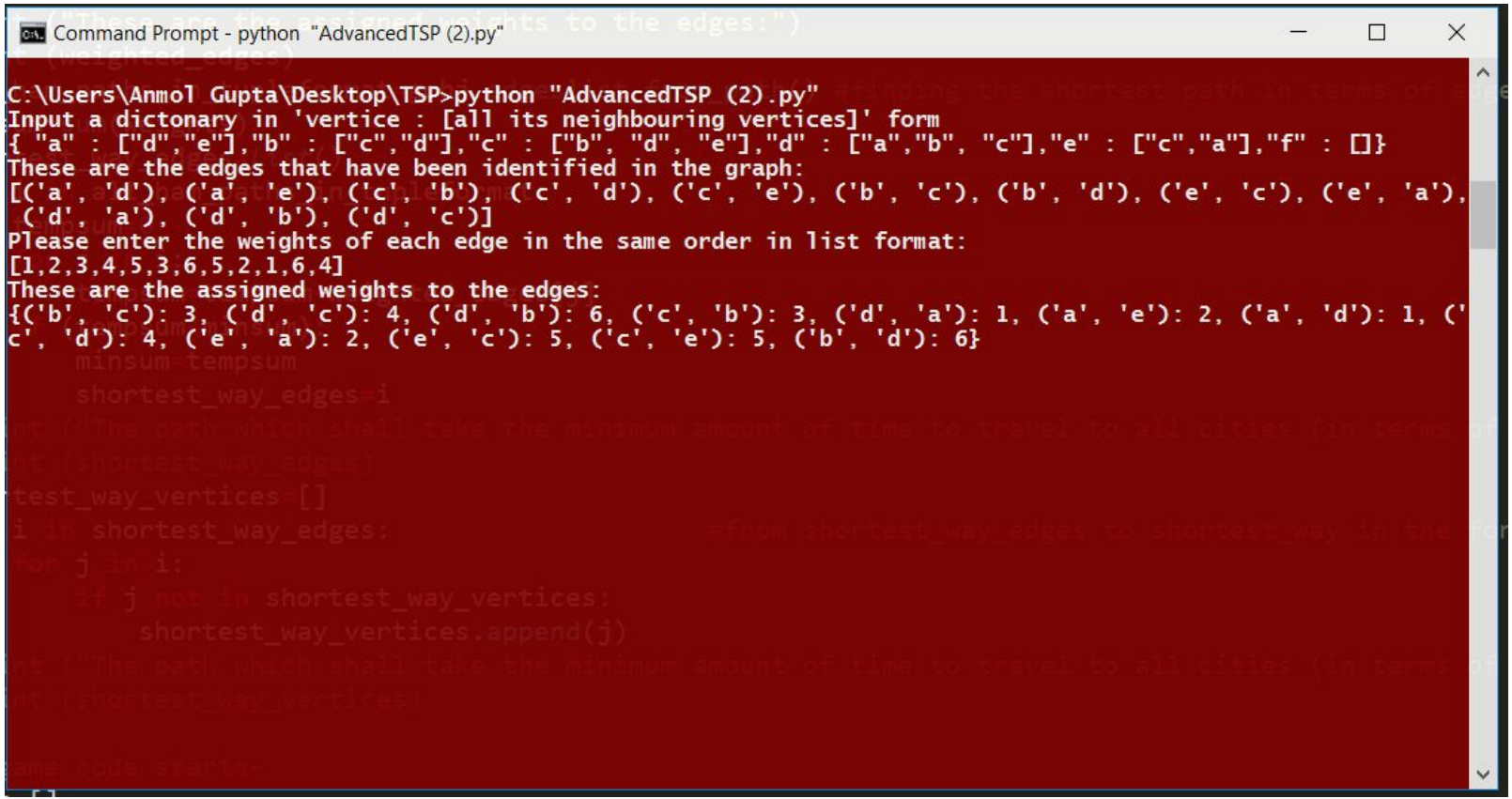
4. This displays the path which shall take the minimum amount to travel all the cities in term of edges and in the term of vertices separately.

RESULTS

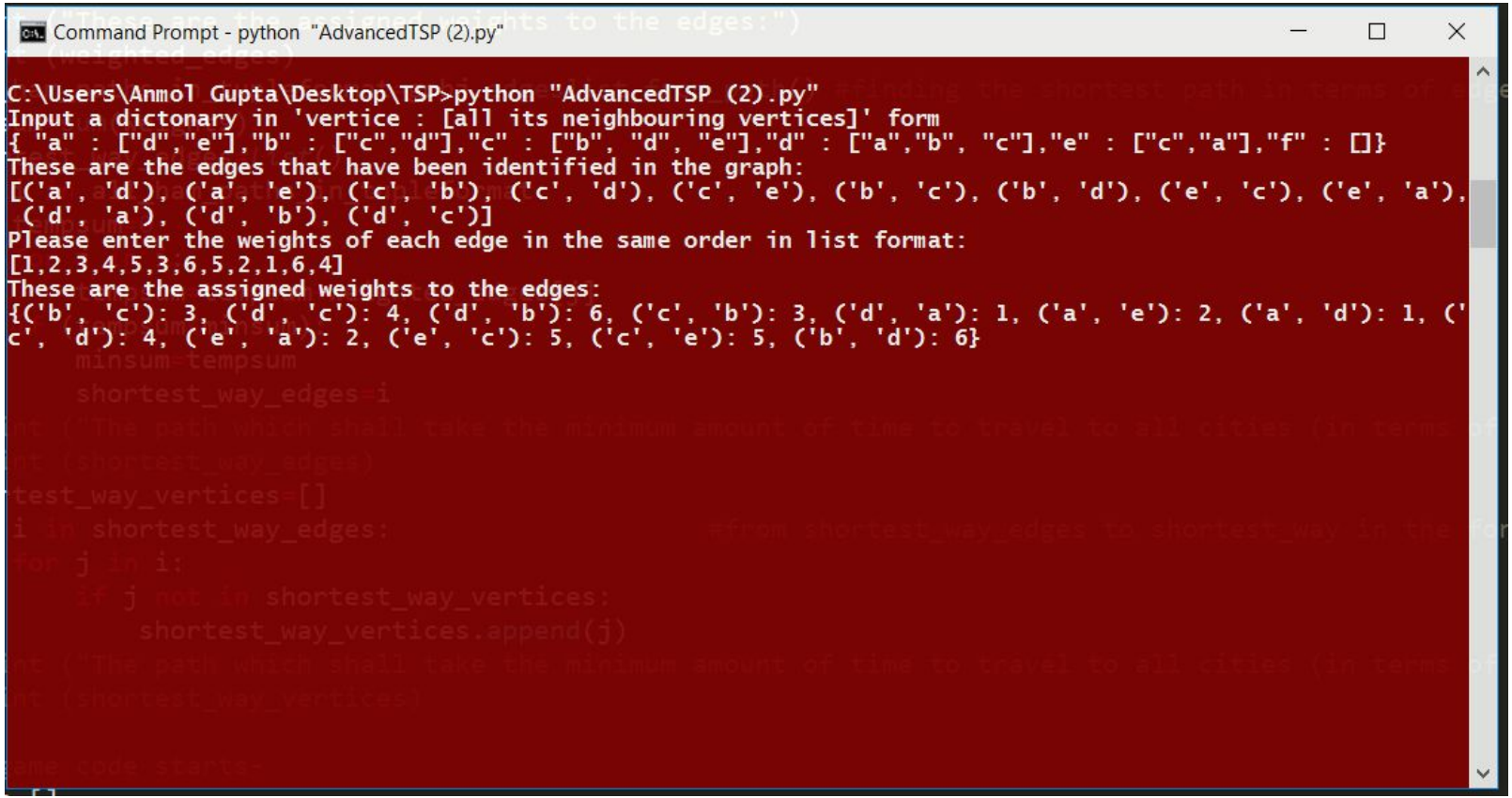
1. Identification of edges from the graph



3. Assignment of weights to edges



2. Assignment of weights to edges



References

- http://www.techeuler.com/python/usage-of-underscores-before-and-after-function-name-in-python/
- https://www.python-course.eu/graphs_python.php
- https://stackoverflow.com/questions/17264174/python-how-to-take-a-dictionary-as-input
- https://pythonprogramming.net/
- http://www.nerdparadise.com/programming/pygame/part1
- https://www.hackerearth.com/practice/algorithms/graphs/hamiltonian-path/tutorial/
- http://www.csd.uoc.gr/~hy583/reviewed_notes/euler.pdf

Work Distribution

Mohak Sahu (15D070047) -

Kavya Prudhvi (15D070014) -

Anmol Gupta (15D070025) -

Harshit Khariwal (15D070026) -