Assignment 4
The Circumnavigations of Denver Long

DESIGN DOCUMENT

Zack Traczyk
CSE13S - Spring 2021

Due: May 2nd at 11:59 pm

1   Objective

The program (tsp) calculates the shortest route between a given vertices. The program can optionally read and write to a file.

2   Given

- Header files for stack, path, and graph

- Pseudocode for recursive search

3   Test Harness

Command arguments:

- h : Command line options

- v : Verbose printing; prints all Hamiltonian paths found as well as total number of recursive calls to dfs()

- u : Specifes the graph to be undirected

- i : infile; the input file containing the cities and edges of a graph (default should be stdin)

- o : outfile; the output file to print to (default is stdout)

3.1 Parse

No need to use a set, just set defaults than change on parse.

After parsing program arguments, input file is parse. The first line grabs number of iterations to store string pointers of strings on following lines.

Next a graph is constructed and the following vertices are stored.

3.2 Execute

Calls ADT in search.c passing the parsed array of string pointers and graph.

4 Algorithm implementation

4.1 Stack, Path, Graph

Given the pseudocode stack, path, and graph implementation is straight forward.

4.2 Search

search.c contains two functions: find adjacent edges given a vertex, and recursively find a Hamiltonian path (DFS).

Adjacent edges are found by finding seeing what edges exist in given row (vertex) in graph.

ADT uses the psedocode in the lab doc.