CIS 263 Project 2

Name: _____

Due Date

- at the start of class on Wednesday, July 30.

Before Starting the Project

- Read the entire project description before starting. This is an individual assignment.

Learning Objectives

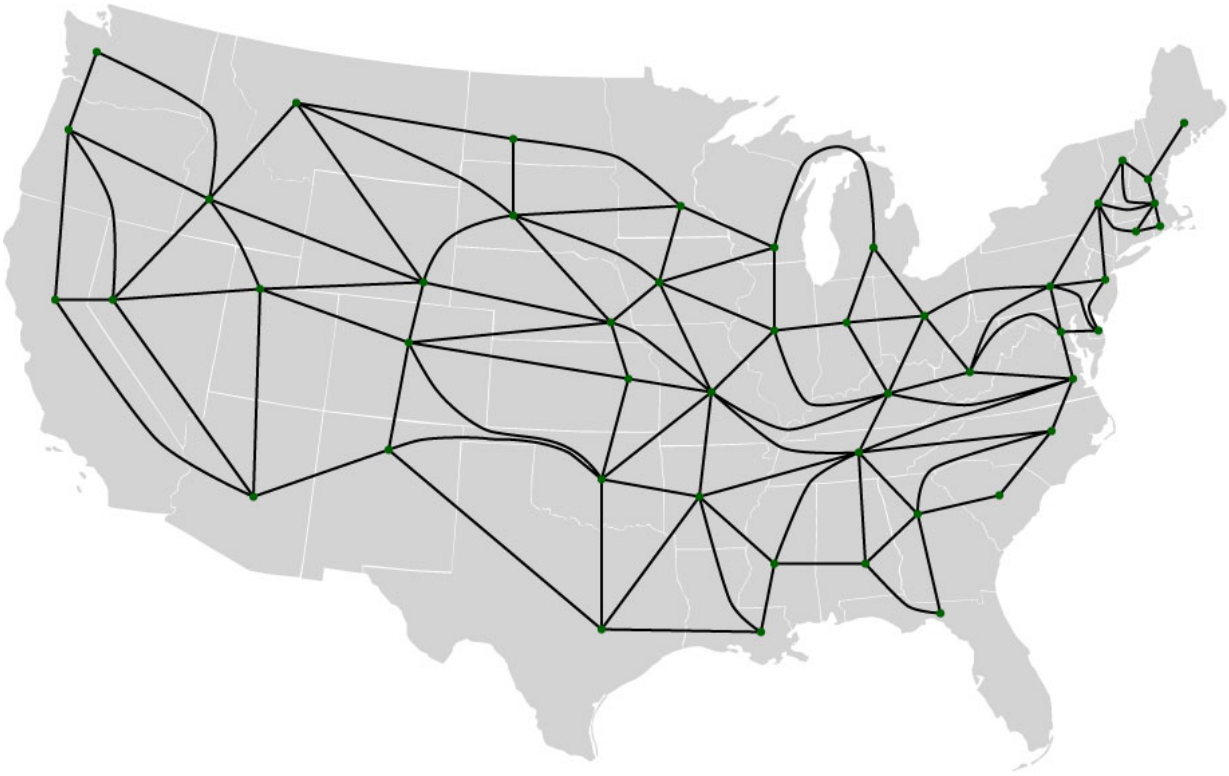
After completing this project you should be able to implement graph algorithms.

Rubric

25 pts commented and readable code	_____
25 pts elegant source code	_____
25 pts concise design	_____
25 pts results during testing	_____

Background

Tom's Tourism would like to sell large signs to the U.S. government to be placed at the intersection point where 3 or 4 states come together. The provided graph has an edge between states if it is possible to drive from one state's capital to a neighboring state's capital without going through a 3rd state. Tom has hired you to determine how many 3 and 4 state signs he can sell. He plans to later expand his business to intersections between countries all around the world and wants an algorithm to compute the sign locations.



Step 1: Gather data and store the graph

- Download the graph from blackboard (edge list notation). Assume your program will be tested with at least this file.
- Read the graph into your program and store it using any structures needed.

Step 2: Functionality

- Write functions as needed to determine 1) how many tri-state combinations there are, 2) print a list of each tri-state intersection, 3) how many four-state combinations there are, and 4) print a list of each four-state area.

Step 3: Testing

- Provide the output needed to answer all four questions.
- State of any observations you have to share with Tom based on these results.
- State if your algorithm will work correctly with country level data if provided a similar graph (e.g., for Africa, South America, Europe, or Asia).

Step 4: Bundle your program

- Bundle your program and turn in all files required to run your program. Also, include the testing output and your comments from Step 3 of your program. Print all of these files and turn them in at the start of class on the due date (along with the first page of this description). Upload any source code to blackboard.
- Ensure you include suitable documentation for your 1) overall project and 2) source code. Include an appropriate comment block at the top of your program.

Grading Criteria

- There is a 50% penalty on programming projects if your solution does not execute or generates errors.
- There is a 50% penalty for not turning in a hardcopy (code and 1st page of this document) and softcopy (zipped if needed) to blackboard.
- Any options/approaches/requirements not specified in this document are left for your own decision making, in keeping with the spirit of the assignment.

Late Policy

Projects are due at the START of the class period and not accepted later. Not turning in the hard copy or soft copy by the due date is considered a late/missing project unless PRIOR arrangements are made.