

CS 330: Algorithms: Design & Practice

Lab#9: Simple Graph Algorithms

The W-O-M-A-N Puzzle – March to Washington, DC

In this lab we will learn how to represent graphs and implement some simple graph algorithms to solve a quintessential Bryn Mawr puzzle. First the puzzle:

You need to travel, by land, from the state of Washington to Washington, DC. You are only allowed to travel through states whose names begin with any of the letters in the word W-O-M-A-N. That is, you can travel through states like, Arkansas, Minnesota, Wyoming, etc. but not through California, Texas, Utah. Is it possible? If so, what would be the path?

Graphs: In class, we reviewed the basics of graphs/networks: terminology, properties, and some basic algorithms (See Chapters 5 & 6 from Cormen). You can solve the puzzle above by modeling it as a graph search problem:

Given a graph of all the states in the United States and their neighbors, you can model the problem as a graph search problem where Washington is the start state and Washington, DC the goal state. A database of states and neighbors is available in a text/csv file:

Link: <http://cs.brynmawr.edu/Courses/cs330/spring2020/USStates.csv>

Here are some of the entries in it:

```
Colorado, Arizona, Kansas, Nebraska, New Mexico, Oklahoma, Utah, Wyoming
Connecticut, Massachusetts, New York, Rhode Island
District of Columbia, Maryland, Virginia
Delaware, Maryland, New Jersey, Pennsylvania
Florida, Alabama, Georgia
Georgia, Alabama, Florida, North Carolina, South Carolina, Tennessee
Hawaii,
Idaho, Montana, Nevada, Oregon, Utah, Washington, Wyoming
Illinois, Indiana, Iowa, Michigan, Kentucky, Missouri, Wisconsin
```

Each line represents a state, followed by all its neighbors.

Task#1: In a programming language of your choosing, create a graph abstract data type ADT to input and store the above graph. The program should print out the state with the most neighbors. And then, use the graph to answer some simple queries:

```
Enter a state: Florida
```

```
Florida has the following neighbors: Alabama, Georgia
```

```
Enter a state: Idaho
```

```
Idaho has the following neighbors: Montana, Nevada, Oregon,
Utah, Washington, Wyoming
```

```
Enter a state: Hawaii
```

```
Hawaii has no neighbors.
```

Task#2: Solve the W-O-M-A-N Puzzle. Your program should first say “Yes” or “No”. If the answer is “Yes” it should print out the names of the states the path goes through to get from Washington to Washington, DC (District of Columbia). That is:

No. There is no way to get from Washington to District of Columbia.

Or

Yes. To get from Washington to District of Columbia, march as follows:

Washington, Oregon, Nevada, ..., District of Columbia

First, think about the best possible implementation of the graph ADT, the algorithm(s) you will use (depth-first or breadth-first search), and how they will be implemented. Your implementation should make judicious use of already implemented data types and structures in the library of the language of your choice (Python, or Java).

NOTE: Your program will be graded based not just on correctness, but also on choice of algorithm(s), programming style, and use of proper abstractions.

What to submit:

1. A printout of just the sample runs (as shown) from the program in Task#1.
2. A printout of the program and its output from Task#2.

Extra Credit

Implement both DFS and BFS algorithms.