

7 Mapping the World.

Introduction

We want to build a map program that records the streets in a city (as well as the intersections) and provides directions for a visitor.

We can get the information we need from various online resources. The most accurate representation of the locations that can easily be correlated with geographic maps is the knowledge of the latitude and longitude of the location. To get enough accuracy, we record each of these pieces of information as degrees, minutes, and seconds - with the first two being integers, and the third one represented as a `double`.

The first problem concerns the representation of a location and producing a travel advice for a travel from one location to another, such as "travel one mile North", or "at XYZ intersection turn left", or "arrive at destination".

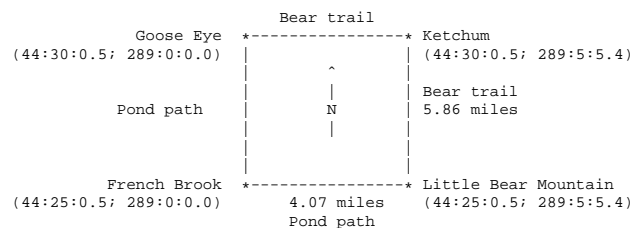
The second problem examines one way that can be used to represent a route from the origin to destination.

The third problem is concerned with the representation of the information about the map that will be used to generate the travel directions.

7.1 Problem: Representing the Map Elements

A location is given by latitude and longitude. Both latitude and longitude are given by degrees, minutes, and seconds. Minutes are whole numbers in the range from 0 to less than 60. Seconds are doubles in the range from 0 to less than 60. The latitude for our maps is in the range from 24 to less than 50 (the locations in the continental USA.) The longitude for our maps is in the range from 220 to 300 — again the continental USA - measured from the Greenwich meridian going East around the Earth, so that New York is at 286 and Boston is at 289.

Here is some sample data that you should use in your examples (of course, you shall add your own examples as well):



- A. Design the classes that represent a location on our map. Make sure that the constructor does not accept invalid data and throws a *RuntimeException* if that happens.
- B. Design the method that converts the distance from one latitude to another into a distance in miles.
- C. Design the method that converts the distance from one longitude to another into a distance in miles.
- D. Design the method that produces the angle that represents the direction of travel from one location to another one measured from North in a clockwise direction.
- E. An intersection on a map is given by its name and its location. A street segment has a name, the starting intersection, and the ending intersection. Design classes to represent intersections and street segments.
- F. Design the method that produces the travel advice for a travel along one street segment.

Note: Design first the method for the advice that specifies the distance, then the advice that gives the direction, then combine the two. For example

“Travel along Bear trail North 5.1 miles to Ketchum”

Note: It is sufficient to give the directions as North, East, South, or West.

- G. Design the method that tells the traveler which way to turn when moving from one street segment to another, e.g.:
“At French Brook turn left”

■

7.2 Problem: Representing the Routing

It is clear that a list of street segments can be used to represent a route from the origin to the destination.

- A. Design the classes to represent a route. Make sure that it is constructed in such was that the ending intersection of a street in the route is the starting intersection of the following street.
- B. Design the method that computes the total distance of the route.
- C. Design the method that produces a `String` that represents the travel advice along this route. Use `"\n"` to separate the lines.

For example one route Ketchum to Goose Eye to Little Bear Mountain would result in:

Total distance from Ketchum to Little Bear Mountain is 14.0 miles.

From Ketchum travel West on Bear Trail 4.07 miles to Goose Eye.

At Goose Eye turn left.

From Goose Eye travel South on Pond Path 5.86 miles to French Brook.

At French Brook turn left.

From French Brook travel East on Pond Path 4.07 miles to Little Bear Mountain.

Arrive at destination at Little Bear Mountain.

■

7.3 Problem: Representing Whole Maps of Data

The interface *IMapping* defines an ADT (Abstract Data Type) that represents pairs of data elements — a key and a value. The user can find the value in the mapping by providing the key to search for.

```
// to represent key-value collection of data
interface IMapping{
    // to add a key-value pair to this mapping
    IMapping add(String key, Object value);

    // does this mapping contain the value
    // corresponding to the given key?
    boolean contains(String key);

    // produce the value in this mapping
    // corresponding to the given key
    // throw a RuntimeException if the key is not found
    Object find(String key);

    // count the number of key-value pairs in this mapping
    int count();
}
```

- A. Design two implementations of the *Imapping* ADT — one as a list of *Objects* and one as a binary search tree of *Objects*.
- B. Use these two implementations to make examples of mappings of intersections. (Yes, it is OK to use the name of the intersection as its key — and to include it in the mapping 'twice'.)
- C. Use these two implementations to make examples of mappings of street segments. Your key can be a concatenation of the names of the starting and of the ending intersections.
- D. A city map data consists of the intersections and street segments. Design the class(es) needed to represent a city map.
- E. Design the method that determines whether a given route is a valid route in this city - the names of the intersections and streets are correct, and the street segments indeed exist on the map.

■

7.4 Writing Problem

Writing assignments are separate from the rest of the assignment for the week. You should work on this assignment alone, and submit your work individually on the due date for the rest of the homework. The answer should be about two paragraphs long – not to exceed half a page or 300 words.

The Big Brother is watching you! Satelites and street-corner cameras can tract your every move. The phone company has records of all the phone calls you made from your cell phone - and where were you when you made them. The credit card companies know about every dollar you spend. Read at least two articles that address this problem. Add your opinion whether this is something to worry about. ■