

2 Understanding Data

Portfolio Problems

- *Classes of Data*: Exercise 2.2 on pages 16 in HtDC.
- *Classes of Data with Containment*: Exercises 3.2 and 3.3, page 26 in HtDC.
- *Unions of Classes of Data*: Exercises 4.4 and 4.6, pages 33-35 in HtDC.
- *Unions of Classes of Data with Containment (Self-Referential Data)*: Exercise 5.6, page 50 in HtDC.
- *A comprehensive example*: Complete the design of all classes and interfaces needed to represent the information about radio shows (with both commercial ads and public service ads) as described in Lab 2.

Pair Programming Assignment

Graphs

All problems in this part become one *Beginner ProfessorJ* program. Combine all examples of data into one *Examples* class.

Most of these problems ask you to translate data definitions and examples from *Beginner HtDP* language to *Beginner ProfessorJ* language, to add class diagrams that represent these data definitions, and to convert the examples of data to the *Beginner ProfessorJ* language.

Draw the class diagram for each problem as you go. The diagrams should grow as we add more classes and interfaces that refer to each other through *containment arrows* and *inheritance arrows*. Make sure you use these arrow correctly.

2.1 Problem

Design the data definition in *ProfessorJ Beginner* language for the class *CartPt* that represents a location on a *Canvas* using the x, y coordinates, where x and y are whole (non-negative) numbers. ■

2.2 Problem

Design the data definition in *ProfessorJ Beginner* language for the class *SimpleNode* that represents a node in a graph. Each node has a name (one or more characters, usually letters and digits) and a location (a cartesian point so that it can be drawn on a *Canvas*). ■

2.3 Problem

Design the data definition in *ProfessorJ Beginner* language for the class *Edge* that represents an edge in a graph. Each edge is defined by the names of the two nodes, its origin and its target node. ■

2.4 Problem

Design the data definition in *ProfessorJ Beginner* language for the classes and interfaces that represent a list of Strings.

Hint: Think of names such as *I洛String*, *Mt洛String*, and *Cons洛String*. ■

2.5 Problem

Develop the data definitions in *ProfessorJ Beginner* language for the class *Node* that includes not only the information required for the class *SimpleNode*, but also includes a list of the names of all neighbors of this node. ■

2.6 Problem

Develop the data definitions in *ProfessorJ Beginner* language that represent all information needed to draw a graph as we have done in Assignment 1. Include a class diagram. Name your class *Graph*. ■

2.7 Problem

Add to your examples two examples of graph that represent the same information as the two examples in Assignment 1 (the graph given there, and your new example graph). ■

City Map

2.8 Problem

Translate the class diagram in figure 1 into a class definition. Also create instances of the class. Include among your examples the data that represents the locations on the enclosed map (figure 2). The grid lines are 20 pixels apart. ■

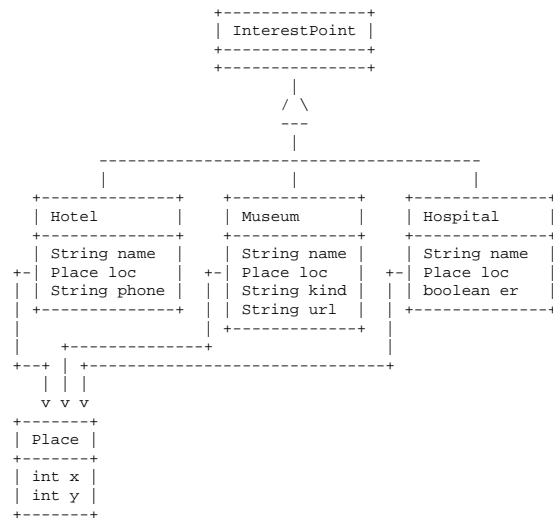
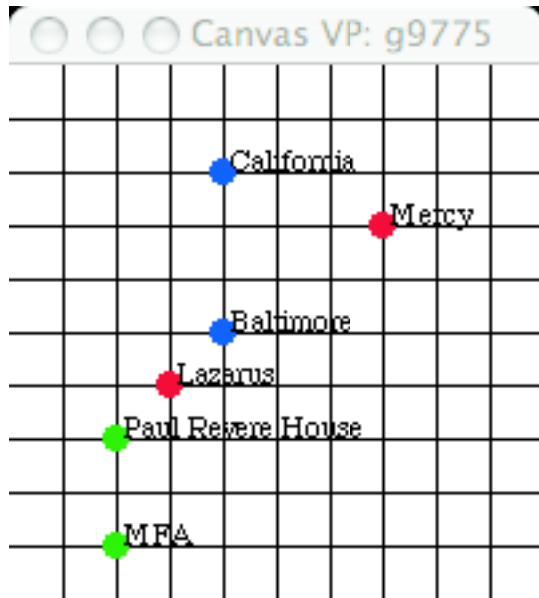


Figure 1: A class diagram for city map points of interest



Hotel California
Hotel Baltimore
Museum of Fine Arts
Paul Revere House
Mercy Hospital
Lazarus Hospital

Figure 2: A sample city map