Computer Science 501
Data Structures & Algorithms
The College of Saint Rose
Fall 2015

Lab 2: Practice with `Vector`s

Due: 6:00 PM, Wednesday, September 16, 2015

This week, you will gain some experience developing and using `Vector` and similar data structures.

You may work alone or in a group of 2 or 3 on this lab. Only one submission per group is needed.

Getting Set Up

To get your BlueJ environment set up for this week's lab assignment, start BlueJ and choose "New Project" from the "Project" menu. Navigate to your folder for this course and choose the name "Lab2" (no spaces) for the project.

Create a document where you will record your answers to the lecture assignment and lab questions. If you use plain text, call it "`lab2.txt`". If it's a Word document, you can call it whatever you'd like, but when you submit, be sure you convert it to a PDF document "`lab2.pdf`" before you submit it.

Lecture Assignment Questions

We will usually discuss these questions at the start of class on the lab due date, so no credit can be earned for late submissions of lecture assignment questions.

? LA Question 1:

Suppose you have 22 shoes in a basket: 5 identical pairs of sneakers, 4 identical pairs of sandals, and 2 identical pairs of dress shoes. You select shoes from the basket in the dark, hoping to get a matching pair, and can check them only after a selection has been made. What is the smallest number of shoes you need to select where you could possibly have at least one matching pair? What is the smallest number you would need to select to have at least one matching pair? Note that a "matching pair" consists of a left and right shoe of the same type (*i.e.*, a left and right sneaker). (4 points)

? LA Question 2:

Suppose you have 5 distinct pairs of gloves, but lose 2 gloves. You are left with either 3 (worst case) or 4 (best case) complete pairs. Assuming that the probability of disappearance for each of the 10 gloves is the same, find the probability of the best-case scenario; the probability of the worst-case scenario; the number of pairs you should expect in the average case. (4 points)

LA Question 3:
Bailey Problem 3.8, p. 65. A formal mathematical proof is not necessary, but be convincing that you understand why the total number of elements copied is proportional to n^2. (2 points)

Practice Programs

Practice Program:
Write the class described in Bailey Problem 3.6, p. 65. You may limit your implementation to include a default constructor that creates a `BitVector` with slots initially for 10 `boolean` values, a second constructor that takes a parameter specifying the number of slots, and the following public methods: `add` at the end, `add` at a given position, `contains`, `get`, `indexOf`, `clear`, `remove` (by position), `set`, `size`, and `toString`. Include a `main` method that thoroughly tests your class and all of its constructors and methods. (25 points)

Question 1:
What are the advantages and disadvantages of using a `BitVector` as you implemented it as compared to using a `Vector` or `ArrayList` that stores boolean values. (3 points)

Programming Assignment

For this week's programming project, you will begin working with some real world data derived from highway systems. This same data will be used in other lab assignments later this semester.

A big advantage of working with this kind of data is that it has a connection to reality, and that we can visualize the data and the results of our manipulations of that data with the Google Maps API. This data is collected by the Clinched Highway Mapping (CHM) Project (`http://cmap.m-plex.com/`). I have taken some of the data from the CHM collaborators and converted into a format that is more convenient for us to load into a graph structure and use. Much more about the project is available at `http://courses.teresco.org/chm/`, but everything you need to know should be on this sheet.

The Data

The data is in "`.gra`" files which have the following format:

- The first line consists of two numbers: the number of vertices, $|V|$, (we'll call them "waypoints") and the number of edges, $|E|$, (road segments that connect adjacent waypoints).
- The next $|V|$ lines descibe the waypoints. Each line consists of a string describing a waypoint (its "label"), followed by its latitude and longitude as floating-point numbers.
- The last $|E|$ lines describe the road segments. Each line consists of two numbers specifying the waypoint numbers (0-based and in the order read in from this file) connected by this road segment, followed by a string with the name of the road or roads that form this segment.

You can find a few dozen example graph files linked from `http://courses.teresco.org/chm/graphs.html`. For example, `usai.gra` describes the entire U.S. Interstate Highway system. `canyt.gra` describes a much smaller system: the territorial highway system in the Yukon. The links of most interest to you are the "download" and "view" links.

Over the course of the semester, you will develop a Java program or programs that can read in graph data, store it appropriately in memory, and perform a variety of operations on that data.

For this week, you will work only with waypoint data and ignore the road segments. Your tasks:

1. Develop a class `Waypoint` that represents that data for a single waypoint. It should include fields for the waypoint name and its latitude and longitude values, a constructor, accessors for the three components, and appropriate `equals` and `toString` methods.

2. Develop a class `WaypointLoader` that has a `main` method that takes the name of an input `.gra` file as a command-line parameter, and reads all of the waypoints from that file into a `Vector` of `Waypoint` objects. It should then enter an input loop where the user is repeatedly prompted for a string, and the program prints out all waypoints in the `Vector` whose labels contain the entered string as a substring, ignoring case.

Submitting

Before 6:00 PM, Wednesday, September 16, 2015, submit your lab for grading. There are two things you need to do to complete the submission: (i) Copy your file with the answers to the lecture assignment and lab questions into your project directory. Be sure to use the correct file name. If you prepared your answers in Word, export to a PDF file and submit that. (ii) Email a copy of your lab (a `.7z` or `.zip` file containing your project directory) to *terescoj@strose.edu*. Please use a meaningful subject line such as "Joe Student Lab2 Submission".

Grading

This assignment is worth 90 points, which are distributed as follows:

Feature	Value	Score
Lecture Assignment Q1	4	
Lecture Assignment Q2	4	
Lecture Assignment Q3	2	
`BitVector` fields	3	
`BitVector` resizes as needed	3	
`BitVector` constructor(s)	2	
`BitVector` other methods	12	
`BitVector` main method with tests	5	
Lab question	3	
`Waypoint` fields	3	
`Waypoint` constructor	3	
`Waypoint` accessors	3	
`Waypoint` `equals`	2	
`Waypoint` `toString`	2	
`WaypointLoader` command-line param	3	
`WaypointLoader` load waypoints into a `Vector`	10	
`WaypointLoader` interactive loop	6	
`WaypointLoader` print all matching waypoints	10	
Comments	6	
Naming conventions	3	
Formatting	1	
Total	90	