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

Lab 9: Best Of

Due: 6:00 PM, Wednesday, November 11, 2015

For this lab, you will be implementing an interesting ordered structure and using it in a number of contexts.

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 "Lab9" (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 "`lab9.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 "`lab9.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:

Bailey Problem 10.10, p. 246. (2 points)

? LA Question 2:

Bailey Problem 10.12, p. 246. (3 points)

LA Question 3:

Consider the following three options:

1. a standard `Vector`
2. a `MyVector` from the comparator lab, which is capable of sorting its contents on demand (by calling the `sort` method)
3. an `OrderedVector`, which might be thought of as a `MyVector` that automatically sorts during/after each modification to its contents

Given a set of data which is modified periodically and printed in order periodically, describe briefly the circumstances (relative frequency of modifications and printouts) where each of the above three approaches might be beneficial. (5 points)

Practice Program

Practice Program:

Write a Java application `SyracuseSeq.java` that generates a *Syracuse Sequence* for a given value of n. This sequence will be used in some of the lab questions (see thought question 3 on p. 276 of Bailey) following the programming assignment. Your program should take a number as a command-line parameter and print the values in Syracuse sequence starting at that value until it reaches 1, and then print out the length of the sequence. (6 points)

Programming Assignment

Your programming assignment is to complete the laboratory in Section 11.4 of Bailey – computing the "Best Of" a collection of values by some metric (the `compareTo` method of `Comparable` objects).

First, carefully read the lab description in the text. Before you start coding, think carefully about the `BestOf` class. What does it extend and/or implement? What instance variables are needed? Do you plan to store the contents of your `BestOf` in order? Why or why not?

Your `BestOf`'s `main` method should run one or more simple test cases for your structure. You may use the one described at the top of p. 276 or devise your own test cases.

Question 1:

Answer thought question 1 from Bailey p. 276. (2 points)

Question 2:

Answer thought question 2 from Bailey p. 276. (3 points)

Rather than answering thought question 3 as is, we will expand it a bit.

Write a program as a `main` method to a class `Syracuse` in a file `Syracuse.java` that gen-

erates the Syracuse sequences for all starting values from 1 up to n, given as a command-line parameter. Your program should track and report the following:

1. The top k (a second command-line parameter) intermediate values encountered at any point in the sequences. At the end it should report the k values, and for each, print out the value, the n that generated it, and the position in the sequence for n where the large value occurred.
2. Find the top k longest sequences. At the end it should report the length of the k longest sequences and the n that generated each.

You should maintain two `BestOf`s for this program, one responsible for each above the parts above. Should you need to use an extra class to keep track of information for the Syracuse sequence generation, you can declare a non-public class inside of `Syracuse.java` that can be used only within that Java file. See the `MazeRunner` program for examples of these non-public classes.

The intermediate values you will generate can get quite large, so to be safe, use values of type `long` and `Long` instead of `int` and `Integer` as you generate your sequences.

? Question 3:
What are the top 25 intermediate values encountered at any point in the sequences up to $n = 10,000$? Include the n that generated the value and the position in that n's sequence. (2 points)

? Question 4:
What are the lengths of the top 25 longest sequences for values of n up to 10,000? Include the value of n that generated each. (2 points)

Expanding and Using `BestOf`

Now that you have a working `BestOf` structure, let's use it to compute some things about the mapping data we saw from an earlier lab. You wrote a program that included a `Waypoint` class that held information from the `.gra` files linked from `http://courses.teresco.org/chm/graphs.html`.

What you need to do:

1. Enhance your `Waypoint` implementation to implement the `Comparable` interface. There is no "natural" ordering here, so just have your `compare` method order `Waypoints` in alphabetical order by waypoint name.
2. Enhance your `BestOf` implementation as indicated in thought question 4 to use a `Comparator` to compare objects for inclusion in the structure.
3. Write `Comparator` classes that compare `Waypoint` objects in the following ways: increasing latitude (northernmost), decreasing latitude (southernmost), increasing longitude (easternmost), decreasing longitude (westernmost), alphabetical order by waypoint name, order by length of waypoint name.

4. Write a program `WaypointBest` that has a `main` method that takes 2 command-line parameters: a `.gra` file name, and a number n, and reports the best n waypoints in the file using each of the 6 `Comparators` from the previous item.

Note: for this section, all except the `BestOf` enhancement will be graded as a practice program.

Submitting

Before 6:00 PM, Wednesday, November 11, 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 Lab9 Submission".

Grading

This assignment is worth 80 points, which are distributed as follows:

Feature	Value	Score
LA Question 1 (10.10)	2	
LA Question 2 (10.12)	3	
LA Question 3 (ordered structure options)	5	
`SyracuseSeq.java` correctness	6	
Basic `BestOf` correctness (no `Comparators`)	15	
`BestOf` test(s) in `main` method	2	
`Syracuse` `main` method correctness	7	
`BestOf`/`Syracuse` program design	3	
`BestOf`/`Syracuse` program style	3	
`BestOf`/`Syracuse` program documentation	5	
Question 1 (thought question 1)	2	
Question 2 (thought question 2)	3	
Question 3 (top 25 numbers)	2	
Question 3 (top 25 sequence lengths)	2	
`BestOf` using `Comparators`	5	
`Waypoint` that implements `Comparable`	2	
6 `Waypoint` `Comparators`	6	
`WaypointBest` correctness	7	
Total	80	