

HW4: Java 2

Due Feb 10 by 11:59pm **Points** 30 **Submitting** a file upload

Problem 2 (15 points).

a. In this problem, you'll be working with the Food, Fruit, Apple, and Orange classes from HW 3. You should modify your Fruit class so that it implements Java's Comparable interface:

<https://www.javatpoint.com/Comparable-interface-in-collection-framework> 

(<https://www.javatpoint.com/Comparable-interface-in-collection-framework>)

A piece of fruit should be considered greater if its ripe Date is later.

You should also create a main function and/or class that creates an ArrayList of Fruit, prints out its contents in a nice format, sorts it, and prints out its contents again. Note: Implementing Comparable means you can use the Java built-in sort function.

You may wish to refer to the documentation for the Date class:

<https://docs.oracle.com/javase/7/docs/api/java/util/Date.html> 

(<https://docs.oracle.com/javase/7/docs/api/java/util/Date.html>)

For this problem, submit your Fruit.java file and your main file. The main file should be named HW4P2aLastname.java where Lastname is your last name.

b. In this problem, you'll work with the SortedList class from HW3. You should modify your SortedList class so that it implements Java's Iterable interface:

<https://docs.oracle.com/javase/8/docs/api/java/lang/Iterable.html> 

(<https://docs.oracle.com/javase/8/docs/api/java/lang/Iterable.html>)

You should also create a main function and/or class that creates a SortedList, adds elements to it, and uses a foreach loop to print out its contents.

Hint: it may be helpful to know that you can pass an array into the Arrays.stream() method to convert it into a Stream. Note that Stream inherits from BaseStream.

For this problem, submit your SortedList.java file and your main file. The main file should be named HW4P2bLastname.java where Lastname is your last name.

Problem 3 (10 points) In this problem, you'll be working with the Food, Fruit, Apple, and Orange classes from HW 3. Translate these classes into C++, and add the following two classes:

Vegetable, which inherits from Food (**Data:** expiration:Date, **Methods:** Vegetable(), Vegetable(n:String, e:Date), expireTime():Date)

Tomato, which inherits from Vegetable and Fruit (**Data:** None, **Methods:** Tomato(), Tomato(e:Date, r:Date, n:String))

Add print statements to all constructors that show which class's constructor has been called. You should ensure that the Food constructor is called only once when a Tomato object is created.

Also include a main function that creates a Tomato object and calls all of its member functions.

Note that C++ doesn't have a built-in Date type like Java, so instead you may use whatever type you like to represent the date; you can use functionality from ctime, create your own Date type, etc.

For this problem, submit your Food.h, Fruit.h, Apple.h, Orange.h, Vegetable.h, and Tomato.h files (please write the entire class in the header file to control the explosion of files) and your main file. The main file should be named HW4P3Lastname.cpp where Lastname is your last name.

Problem 4 (5 points). Show that this CFG is ambiguous. What is the problem with a grammar being ambiguous if we're trying to use it to represent a programming language?

$A \rightarrow A < A \mid A > A \mid A != A \mid A == A \mid (A) \mid N$

$N \rightarrow 0 \mid 1 \mid 2 \mid 3 \mid 4 \mid 5 \mid 6 \mid 7 \mid 8 \mid 9$

Please name the file you upload for this problem HW3P4 and submit as either a pdf or jpg