CS 250 - Fall 2021 - Homework 06
(OPTIONAL & EXTRA CREDITS)

Objectives

The objectives of this assignment are:

- Testing your understanding of abstract data types
- To give additional practice with List, Set, Stack, Queue, Map.
- To give additional practice with parsing inputs and exception handling.

1. Programming Questions

Q1. (30 points) Find the max occurrences in a list

Write a method maxOccurrences that accepts a list of integers as a parameter and returns the number of times the most frequently occurring integer (the “mode”) occurs in the list. Solve this problem using a map as auxiliary storage. If the list is empty, return 0. For example, given a list of integers {2, 3, 1, 3, 5, 4, 6, 3, 8}. Element number 3 has the max occurrences in the given list (3 times).

Q2. (20 pts)

What output will be produced if we call the methodA() with n=3? If we swap two lines, 5 and 6, will the output be the same or different? Explain your answers.

```
Line 1    public static void methodA(int n) {
Line 2        if (n < 1) {
Line 3            System.out.println('B');
Line 4        } else {
Line 5            methodA(n - 1);
Line 6            System.out.println('R');
Line 7        }
Line 8    }
```

Q3. (20 pts) Sort ArrayList<integer> in decreasing order

Suppose you have an ArrayList<Integer> named list that you want to be sorted from largest to smallest. Unfortunately, the sort( ) method in the Collections class will sort the list from smallest to largest. What is the easiest way to sort the list from largest to smallest?

Q4. (30 pts) Remove duplicate names in a text file

In this problem, you are given a text file containing person names which are separated by the new line characters (each name is on a different line). A sample input file is shown below:

Brooke Trout
Dinah Soars
Jed Dye
Brooke Trout
Jed Dye
Paige Turner

The input file can contain duplicate names. Therefore, you need to write a Java program that eliminates the duplicate names. First, your program reads each name from the file into a list, and then output all names in the list without duplicates.

---

2. Submission Requirements and Grading Rubrics

2.1 Submission Requirements

A. Source code: Write your **own** code

B. Program requirements: Your program can solve and meet all of the requirements described in 2.1 and 2.2. Do not write additional code that is not required.

C. Coding style: Your source code must be properly indented and contain adequate comments for class, methods, variables, etc.

- At the beginning of your Java class file, specify the following items:
    - Your name, the assignment number for this program, the creation date
    - The problem description that defines your application
    - Enumerate (list) the goals of your application
    - Enumerate (list) the inputs of your application
    - Enumerate (list) the outputs of your application
- Before each method in your class, specify the following items:
    - Specify the purpose of the corresponding method
    - Enumerate the possible inputs and outputs of the corresponding method
    - Write down the pseudocode to implement the corresponding method
- Others comments for variables, inside method implementation

2.4 Grading Rubrics

Your code in this part will be evaluated using the following grading scheme (50 points for completed functional source code + bonus 10 points for good code comments):

Points	Item
25	Correct class, variable, and method definition
10	Consistent coding style, such as indents, curly bracket positions, etc.
15	Have all required variables and methods
10	Good and helpful comments

Assignment Submission Instructions

1. Compress the whole Eclipse source code of your project questions in zip format using the Export function

2. Submit above items through D2L.