

CSE3040: Java Programming (Fall 2020)

Midterm Project 2

Due: December 8, 11:59PM (KST)

※ Write a Java code for each problem. Make sure your program satisfies all specified requirements. When an example result is given, output of your program should match the result.

Problem 16 (20 points)

- In this problem, you will write a program that reads data from an input file and prints out the data in a sorted order.
- The input file contains the list of items and their retail prices. An example input file looks like the following.

```
apple 3.99
watermelon 11.99
strawberries 4.99
pear 6.99
mango 8.99
bananas 4.99
pineapples 7.99
kiwi 5.99
raspberries 2.99
peaches 5.99
```

- Your program should print the items in the ascending order of price (cheap items first). If multiple items have the same price, then they should be sorted in the alphabetical order of their names.
- You are given the code for class Problem16, which must NOT be modified. Write other parts of the code to complete the program.

```
public class Problem16 {
    public static void main(String args[]) {
        ArrayList<Element> list = ElementReader.readElements("input.txt");
        if(list == null) {
            System.out.println("Input file not found.");
            return;
        }
        Collections.sort(list);
        Iterator<Element> it = list.iterator();
        while(it.hasNext()) System.out.println(it.next());
    }
}
```

- Assume that if the input file exists, the format will be always correct: item name and price on each line. You do not need to worry about handling wrong input format.

- For the example input file shown above, the expected output is as follows.

```
raspberries 2.99
apple 3.99
bananas 4.99
strawberries 4.99
kiwi 5.99
peaches 5.99
pear 6.99
pineapples 7.99
mango 8.99
watermelon 11.99
```

- In the code you write, a method must never throw exceptions. For checked exceptions, you should handle all of them using try-catch statements.
- If the input file is not found, your program should print "Input file not found." and terminate. You should NOT make your program crash due to run-time errors.
- In the evaluation process, we may use a different input file to check the correctness of your program.
- If you define a new class, all instance variables must be declared as private, if there is any.

Problem 17 (20 points)

- This problem is similar to Problem 16, but we will use a **map** instead of a list here.
- In this problem, the goal is to print the elements in the alphabetical order of their names. The example file and the corresponding output of the program is shown below.

input.txt

```
apple 3.99
watermelon 11.99
strawberries 4.99
pear 6.99
mango 8.99
bananas 4.99
pineapples 7.99
kiwi 5.99
raspberries 2.99
peaches 5.99
```

output.txt

```
apple 3.99
bananas 4.99
kiwi 5.99
```

```
mango 8.99
peaches 5.99
pear 6.99
pineapples 7.99
raspberries 2.99
strawberries 4.99
watermelon 11.99
```

- You are given the code for class Problem17, which must NOT be modified. Write other parts of the code to complete the program.

```
public class Problem17 {
    public static void main(String args[]) {
        Map<String, Double> map = MapManager.readData("input.txt");
        if(map == null) {
            System.out.println("Input file not found.");
            return;
        }
        System.out.println(map);
    }
}
```

- Assume that if the input file exists, the format will be always correct: item name and price on each line. You do not need to worry about handling wrong input format.
- Also, you can assume that **there are no two items with the same name**.
- In the code you write, a method must never throw exceptions. For checked exceptions, you should handle all of them using try-catch statements.
- If the input file is not found, your program should print "Input file not found." and terminate. You should NOT make your program crash due to run-time errors.
- In the evaluation process, we may use a different input file to check the correctness of your program.
- If you define a new class, all instance variables must be declared as private, if there is any.

Problem 18 (20 points)

- This problem is similar to Problem 17, except that this time your program should print the items in the ascending order of price (cheap items first). If multiple items have the same price, then they should be sorted in the alphabetical order of their names. In other words, your output should be the same as that of Problem 16.
- An example input, and its corresponding output are as follows.

input.txt

```
apple 3.99
watermelon 11.99
strawberries 4.99
pear 6.99
mango 8.99
bananas 4.99
pineapples 7.99
kiwi 5.99
raspberries 2.99
peaches 5.99
```

output.txt

```
raspberries 2.99
apple 3.99
bananas 4.99
strawberries 4.99
kiwi 5.99
peaches 5.99
pear 6.99
pineapples 7.99
mango 8.99
watermelon 11.99
```

- You are given the code for class Problem18, which must NOT be modified. Write other parts of the code to complete the program.

```
public class Problem18 {
    public static void main(String args[]) {
        Map<String, Double> map = MapManager.readData("input.txt");
        if(map == null) {
            System.out.println("Input file not found.");
            return;
        }
        System.out.println(map);
    }
}
```

- Assume that if the input file exists, the format will be always correct: item name and price on each line. You do not need to worry about handling wrong input format.
- Also, you can assume that **there are no two items with the same name.**
- In the code you write, a method must never throw exceptions. For checked exceptions, you should handle all of them using try-catch statements.
- If the input file is not found, your program should print "Input file not found." and terminate. You should NOT make your program crash due to run-time errors.

- In the evaluation process, we may use a different input file to check the correctness of your program.
- If you define a new class, all instance variables must be declared as private, if there is any.

Problem 19 (20 points)

- In this problem, you will write a program that connects to a web server and finds the information we need.
- Specifically, we want to find the bestseller books from the Barnes & Noble website, which is a US online bookstore.
- You can find the webpage at:
https://www.barnesandnoble.com/b/books/_/N-1fZ29Z8q8
- From this site, you should fetch the 20 bestseller books, and print them on the screen like the example shown below.
 - Each line should include the title, the author(s), and the price in US dollars.
 - The books must be sorted in the reserve order of their ranks, as shown below.

```
#20 No Time Like the Future: An Optimist Considers Mortality, Michael J. Fox, $24.99
#19 Piece of My Heart, Mary Higgins Clark, $18.89
#18 The Ickabog, J. K. Rowling, $21.49
#17 Is This Anything?, Jerry Seinfeld, $21.00
#16 Patients at Risk: The Rise of the Nurse Practitioner and Physician Assistant in
Healthcare, Niran Al-Agba, $27.95
#15 The Return, Nicholas Sparks, $19.60
#14 Clanlands: Whisky, Warfare, and a Scottish Adventure Like No Other, Sam Heughan, $21.59
#13 Fortune and Glory (B&N Exclusive Edition) (Stephanie Plum Series #27), Janet Evanovich,
$20.29
#12 The Deep End (Diary of a Wimpy Kid Series #15), Jeff Kinney, $10.49
#11 Greenlights, Matthew McConaughey, $23.99
#10 Our Time Is Now: Power, Purpose, and the Fight for a Fair America (Signed Book), Stacey
Abrams, $27.99
#9 Broken Horses (Signed Book), Brandi Carlile, $28.00
#8 The Law of Innocence (Signed Book) (Mickey Haller Series #6), Michael Connelly, $29.00
#7 A Wealth of Pigeons: A Cartoon Collection, Steve Martin, $20.99
#6 Share Some Kindness, Bring Some Light, Apryl Stott, $15.99
#5 Accidentally Wes Anderson, Wally Koval, $29.99
#4 Daylight (Signed Book) (Atlee Pine Series #3), David Baldacci, $29.00
#3 A Time for Mercy, John Grisham, $20.96
#2 Deadly Cross (Alex Cross Series #26), James Patterson, $20.30
#1 A Promised Land, Barack Obama, $34.99
```

- Note that at the time of submission, the list of best-selling books may change, and the output may not exactly look like this. Still, you will get the idea.
- You are given the code for class Problem 19, which must NOT be modified. Write other parts of the code to complete the program.

```

public class Problem19 {
    public static void main(String[] args) {
        ArrayList<BookInfo> books;
        books = BookReader.readBooks("https://www.barnesandnoble.com/b/books/_/N-1fZ29Z8q8");
        Collections.sort(books);
        for(int i=0; i<books.size(); i++) {
            BookInfo book = books.get(i);
            System.out.println(book);
        }
    }
}

```

- For this problem, you must not use any external library, such as jsoup.
- If you define a new class, all instance variables must be declared as private, if there is any.
- You must handle all checked exception using the try-catch block, but it is enough that you call `printStackTrace()` in your catch block.

Problem 20 (20 points)

- In this problem, you will write a program that prints the same output as Problem 19. However, this time you must use jsoup to parse the HTML file. (You must not use raw string parsing as in Problem 19.)
- You are given the code for class Problem 20, which must NOT be modified. Write other parts of the code to complete the program.

```

public class Problem20 {
    public static void main(String[] args) {
        ArrayList<BookInfo> books;
        books = BookReader.readBooksJsoup("https://www.barnesandnoble.com/b/books/_/N-1fZ29Z8q8");
        Collections.sort(books);
        for(int i=0; i<books.size(); i++) {
            BookInfo book = books.get(i);
            System.out.println(book);
        }
    }
}

```

- The output of the program should be the same as Problem19.
- If you define a new class, all instance variables must be declared as private, if there is any.
- You must handle all checked exception using the try-catch block, but it is enough that you call `printStackTrace()` in your catch block.

※ Submission Instructions

※ Carefully read this part and follow the instructions. Not properly following the instructions here may result in point deduction.

(1) For this homework, you are going to submit only the .java files. You are going to submit one .java file for each problem.

(2) For problem 16, your file name should be **Problem16.java**. This means that your public class name should also be Problem16. For other problems, you should name your .java files this way.

(3) Once you are ready to submit, you should make the .java files into a single zip file named **cse3040_mp2_20180001.zip**. The numbers in the file name should be your **student ID**.

When you make the zip file, **make sure that you are not using subfolders inside the zip file**. When the zip file is extracted, the .java files should appear without having any subdirectory structures.

(4) Submit your zip file on the cyber campus.

※ Evaluation Criteria

Your solution to each problem will be tested with various test cases. You will get full points if your program passes all tests. If not, points will be given based on percentage of test cases passed.

For this homework, each problem is worth 20 points. The perfect score is 100.

※ Academic Integrity

- You should write your own code. You can discuss ideas with other students but must not copy their work. You can also get help from the Internet, but you must not copy the source code from the Internet either. We have a duplicate check program which tests whether your source code is similar to other students' code as well as codes that are on the Internet.

- Duplicate work will receive zero grade.

※ Late Policy

- 10% of the score is deducted for each day, up to three days. Submissions are accepted up to three days after the deadline.