CS 2334: Programming Structures and Abstraction

Project 2

Due Date: Oct 09, 2019

Abstraction with ASCII

Fall 2019

100 pts

1. Objective:

The objective of this programming project is to implement a java program where Abstract class/method can be inherited. After completing the project, students will have an intermediate understanding of Abstraction.

2. Project Specification:

2.1. Overall Program Behavior:

Form Project 1, you are already familiar with Mesonet where a station ID is a 4-letter code. For project 2, a 4-letter Station ID will be used as the input. Using abstraction, you will write classes (Empty class will be provided) to compute some averages of the 4-letter code. Calculation method of the averages have been discussed later.

In this project, you are required:

1. to write MesoAbstract.java, MesoInherit.java, PosAvg.java, and LetterAvg.java; we will provide the Main.java and MesoStation.java, and a text file (Mesonet.txt);
2. to submit on Github at least 20 times for version control; and
3. to write a README.md and upload on Github (will be reviewed and graded by an instructor during code review).

2.2 Input Format: (from Driver.java)

One station ID will be entered. For example, NRMN:

```
String stID = "NRMN";
MesoInherit mesoInherit = new MesoInherit(new MesoStation(stID));

PosAvg posAvg = new PosAvg(stID);
System.out.println("The Index of the city is: " + posAvg.indexOfStation());
System.out.println(posAvg);
System.out.print("\n");

System.out.println("Ascii Ceiling is " + mesoInherit.calAverage()[0]);
System.out.println("Ascii Floor is " + mesoInherit.calAverage()[1]);
System.out.println("Ascii Average is " + mesoInherit.calAverage()[2]);
System.out.print("\n");

System.out.println("Letter Avg: " + mesoInherit.letterAverage());
System.out.print("\n");

LetterAvg letterAvg = new LetterAvg(mesoInherit.letterAverage());
```

```
System.out.println("Total number of stations starting with letter '" +
mesoInherit.letterAverage() + "' is " + letterAvg.numberOfStationWithLetterAvg() +
".");

System.out.print(letterAvg);
```

2.3 Output Format:

```
The Index of the city is: 77
This index is average of NOWA and OILT, NEWP and OKCE, and so on.

Ascii Ceiling is 79
Ascii Floor is 78
Ascii Average is 79

Letter Avg: O

Total number of stations starting with letter 'O' is 4.

They are:
OILT
OKCE
OKEM
OKMU
```

** This is a sample output, for ‘NRMN’ as input. Your code will be tested with many different inputs.

2.4 Description of output:

You have 4 letters: ‘N’, ‘R’, ‘M’, and ‘N’

```
The Index of the city is: 77
```

/*If you calculate the Index of NRMN in the Mesonet.txt file, starting from 1 is 77. Remember, array index 0 is the first element, i.e., file index is 1.*/

```
This index is average of NOWA and OILT, NEWP and OKCE, and so on.
```

/*If the index is 77, then, 77 is the average of 76(NOWA) and 78(OILT). It’s also the average of 75(NEWP) and 79(OKCE). Showing up to two stations is enough.
Note: since you are calculating up to N+2 or N-2, if you give the first two stations or last two stations as input, it will generate error. Therefore, you can skip giving the first two stations or last two stations as input.*/

```
Ascii Ceiling is 79
Ascii Floor is 78
Ascii Average is 79

Letter Avg: O
```

/*You have to get the ASCII value for N R M and N as 78, 82, 77, and 78. Sum of these ASCII value is 315. Dividing 315 by 4, we get 78.75

Taking the ceiling of 78.75, you will get the first part: `Ascii Ceiling is 79`

Taking the floor, you will get the second part: `Ascii Floor is 78`

For the third part, if the fraction part is less than 0.5, the Average would be floor. Otherwise, if the fraction part is greater than or equal to 0.5, the Average would be ceiling.

For the fourth part, the letter value of the average which is 79 (ceiling of 78.75), which is equivalent to letter ‘O’.*/

```
Total number of stations starting with letter 'O' is 4.
```

/*In the station list, number of stations starting with ‘O’ is 4.*/

```
They are:
OILT
OKCE
OKEM
OKMU
```

/*Finally, here is the list of those four stations.*/

Pls, print in the specific format as provided above, since, Zylab will grade you automatically.

3. File names:

File names for this project are as follows:

We will provide: Main.java, MesoStation.java, and Mesonet.txt

You will write: *MesoAbstract.java, MesoInherit.java, PosAvg.java, LetterAvg.java, and* README.md

MesoAbstract.java: You will create two abstract method calAverage() and letterAerage(). Return type of calAverage() should be an integer array and for letterAverage() is char.

```
 int[] calAverage() and char letterAverage();
```

PosAvg.java: You will write necessary code to generate the first two lines of the output, related with the position/index.

MesoInherit.java: You need to write this class to generate intended output from line 3 to line 6. Here, you must inherit the abstract class/method as necessary.

LetterAvg.java: This class will contain the code to generate the output from line 7 to the end.

The documentation (README.md) should consist of discussion of your problem-solving approach and details analysis of your implemented code including description of each class/method/variable.

4. Points Distribution:

Bits and pieces	Points
Java files (on Zylab for automatic grading)	80
Github submission (at least 20 submissions, each submission contains 0.5 point, will be graded during reivew)	10
README.md (on Github, will be graded during review)	10

5. Submission Instructions: (**Due October 09, 2019, 11:59 pm**)

This project requires the submission of *soft copy only, as follows:*

- *Main.java,*
- *MesoAbstract.java,*
- *PosAvg.java,*
- *MesoInherit.java,*
- *MesoStation.java,*
- *LetterAvg.java, and*
- *Documentation (README.md)*

Plagiarism will not be tolerated under any circumstances. Participating students will be penalized depending on the degree of plagiarism.

6. Late Penalty:

Submit your project before the due date/time to avoid any late penalty. **A late penalty of *20% per hour* will be imposed after the *due date/time*.** After five hours from the due date/time, you are not allowed to submit the project.

Good Luck!!