

CS526  
Homework Assignment 2

Due: 2/4

This assignment has two parts.

### Part 1

Part 1 is a practice of:

- Reading from a text file
- Defining a class
- Creating objects
- Using *ArrayList*

In homework 1, you wrote a program that determines the course grade of a student. In that assignment, grades of grading components were hardcoded in the main method.

For this assignment, you are required to write a program named *Hw2\_part1.java*, which reads grades of grading components from a text file. Detailed instructions are given below.

An input file, named *student\_grades.txt*, contains grades of grading components of a number of students. The format of the input file is:

```
John, 100, 100, 85, 96
Susan, 85, 90, 95, 72
Molly, 93, 90, 95, 92
Lindsey, 80, 100, 92, 75
Jake, 75, 80, 82, 80
```

Each line has grades of one student. Note that tokens are separated by “a comma and a space.” You need this information when you parse the line (or separate the tokens). The first token is the name of a student and the four numbers that follow are homework grade, project grade, midterm grade, and the final exam grade, in that order.

You must define a class named *CourseGrade* in a separate *CourseGrade.java* file. The class definition must include at least the following:

- A constructor (you may define multiple constructors if you want)
- The following instance variables

```
private String name;
private int homework;
private int project;
private int midterm;
private int finalExam;
```

```
private double average;  
private String grade;
```

- All necessary *getter* and *setter* methods.

Then, your program must (within the *main* method of *Hw2\_part1.java*):

- Read the input file, one line at a time
- Parse a line (or separate tokens)
- Calculate the weighted average and determine the letter grade of the student
- Create an object of *CourseGrade* class
- Add the object to an *ArrayList*
- Repeat the above until all lines in the input file are read
- Print all students' grades in the following format:

```
Number of students is: 5  
John: Average = 94.85, Grade = A  
Susan: Average = 83.45, Grade = B  
Molly: Average = 92.85, Grade = A-  
Lindsey: Average = 83.25, Grade = B  
Jake: Average = 79.0, Grade = C+
```

Technically, you don't need to store students' information in an *ArrayList*. When you read each line from the input file, you can determine the grade and print the necessary information without storing that information. However, since one of the goals of this assignment is a practice of using an *ArrayList*, you must store students' information in an *ArrayList*.

## Part 2

This part is a practice of defining and using a subclass.

**Note:** For this part, you first need to study the *Progression* class and its subclasses that are described in Section 2.2.3 of the textbook.

Create a file named *Hw2Progression.java* that defines the *Hw2Progression* class as a subclass of the *Progression* class. This progression generates a sequence of natural numbers and the next value is determined as follows:

- If the current value  $x$  is an even number, the next value is  $2x - 1$ .
- If the current value  $x$  is an odd number, the next value is  $x + 1$ .
- For example,  
If the first value of the sequence is 1 (default value), the following sequence is created:  
1 2 3 4 7 8 15 16 31 32  
If the first value of the sequence is 4, the following sequence is created:  
4 7 8 15 16 31 32 63 64 127

The class definition must include:

- A default constructor that starts with 1.
- A parameterized constructor that starts with a specified natural number.
- An *advance()* method that determines the next value.

Then, modify the *main* method of the *TestProgression.java* file by adding a code segment that performs the following test:

- Create an *Hw2Progression* object with a default constructor and print the first 10 values on the screen.
- Create an *Hw2Progression* object with 5 as the start value and print the first 10 values on the screen.

### **Documentation**

No separate documentation is needed. However, you must include sufficient inline comments within your program.

### **Deliverables**

Submit *Hw2\_part1.java*, *CourseGrade.java*, *Hw2Progression.java*, and the modified *TestProgression.java* files on Blackboard. Make sure that you include your name at the top of each file as a comment.

### **Grading**

The assignment is worth 20 points.

Part 1 will be tested with a different input file and up to 6 points will be deducted if your output is wrong.

Part 2 will be tested with two different start values and up to 3 points will be deducted for each wrong output.

Up to 4 points will be deducted if your program does not have sufficient inline comments.