EF234302 Object-Oriented Programming (IUP)

Quiz 1

Starting date:	21 September 2023
Deadline:	28 September 2023, 23:59 WIB. **Penalty: 0.15% of grade/minute of tardiness.**
Exam type:	Open, Individual Project
Send to:	
	MM Irfan Subakti <yifana@gmail.com>
	CC to Mashita Dewi <mashitaad@gmail.com>, Adam Haidar Azizi <adamhadaizi2002@gmail.com> & Michael Ariel Manihuruk <michaelariel8@gmail.com> with the subject: **EF234302_OOP(D)_Q1_StudentID_Name**
File type and format:	A zip file containing all of the .java source files & the declaration
Filename format:	EF234302(D)_Q1_StudentID_Name.ZIP

Instruction

Please do these steps as in the following.

1. [**10 points**] In the following `e1` and `d1` represent the fraction $\frac{e_1}{d_1}$ and correspondingly `e2` and `d2` the fraction $\frac{e_2}{d_2}$, where `e1` and `e2` are integers and `d1` and `d2` are positive integers. Write a `Java class`, namely `Q11Fraction.java`, which computes numbers `es`, `ds` and `ep`, `dp`, which stand for the sum and the product of the two fractions, respectively. E.g., $\frac{1}{2}+\frac{1}{3}=\frac{5}{6}$, and $\frac{1}{2}*\frac{1}{3}=\frac{1}{6}$. Note that addition and multiplication on fractions are defined by:

$$\frac{e_1}{d_1}+\frac{e_2}{d_2}=\frac{e_1*d_2+e_2*d_1}{d_1*d_2} \qquad \frac{e_1}{d_1}*\frac{e_2}{d_2}=\frac{e_1*e_2}{d_1*d_2}$$

Test your class for $\frac{1}{2}+\frac{1}{3},\frac{1}{3}+\frac{3}{4},\frac{1}{2}*\frac{2}{3}$, and $\frac{1}{4}*\frac{2}{3}$.

```java
// File: Q11Fraction.java
public class Q11 {
    /**
     * Write down in this class your solution and
     * its explanation of question no. 1 above.
     * Remember how to comment properly.
     */
    public static void main(String[] args) {
        // Test your class in here
    }
}
```

2. [**10 points**] This time is 13:03. We represent the time by two variables `hours` and `minutes`, that is `hours = 13;` and `minutes = 03;`. Write a `Java class`, namely `Q12Time.java`, which computes the angle between the hour hand and the minute hand on a traditional analogue clock. Angles should be measured counterclockwise from hour to minute hand. The results should be rounded and normalised so that it is between 0 and 359 (inclusively). For instance, the angles at 3:00 and 9:00 hours should be 90° and 270°, respectively.
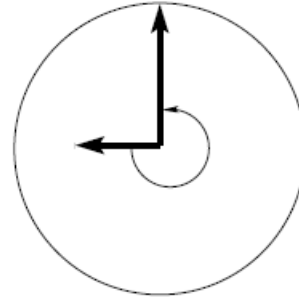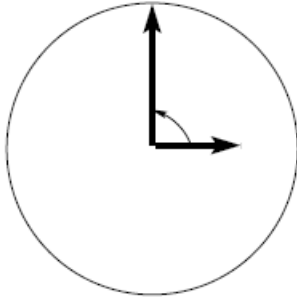
Hint: 1 minute ≈ 6°, 1 hour ≈ 30°. Start at 12 o'clock.
Note that your program must be able to compute the output for arbitrary inputs, in which hours may take values from 0 through 24 and minutes from 0 to 60. Do not forget to write comments which explain why your program is correct.
Test your program for the following times: 9:00, 3:00, 18:00, 1:00, 2:30, and 4:41 (with results 270°, 90°, 180°, 30°, 255°, and 254°, respectively).

```java
// File: Q12Time.java
public class Q12Time {
    /**
     * Write down in this class your solution and
     * its explanation of question no. 2 above.
     * Remember how to comment properly.
     */
    public static void main(String[] args) {
        // Test your class in here
    }
}
```

3. [**10 points**] Define a class `Q13ComputerAccount` and a `constructor` to create it. A computer account should be built from the three strings `realName`, `userName`, and `password`. It should implement methods `printRealName`, `printUserName`, `printPassword`, and `changePassword`. The first three methods take no arguments. The last method takes one argument of type `String`, the new password, and sets the password to the new password; its return value is `void`.
Note that you have always to comment and test your program appropriately, not just for this assignment. We will not write this to the assignments in future, but still, if you fail to do so marks will be subtracted.

```java
// File: Q13ComputerAccount.java
public class Q13ComputerAccount {
    /**
     * Write down in this class your solution and
     * its explanation of question no. 3 above.
     * Remember how to comment properly.
     */
    // Define your properties/fields here

    // Define your constructor here

    // Define your getters and setters in here

}
```

```
// File: Q13ComputerAccountTest.java
public class Q13ComputerAccountTest {
    /**
     * Write down in this class your solution and
     * its explanation of question no. 3 above.
     * Remember how to comment properly.
     */
    public static void main(String[] args) {
        // Test your class: Q13ComputerAccount in here
    }
}
```

4. [**10 points**] Define a class `Q14Student` as well as a constructor and getter methods. A `Q14Student` should be represented by her/his name, her/his Student ID, and the Degree Programme, each of which is a String. Also, write a method `toString` that returns a `String` in the following format:

   `"[Kinanti Wedariningsih, ID: 1000001, Bachelor Informatics]"`.

```
// File: Q14Student.java
public class Q14Student {
    /**
     * Write down in this class your solution and
     * its explanation of question no. 4 above.
     * Remember how to comment properly.
     */
    // Define your properties/fields here

    // Define your constructor here

    // Define your method/function toString in here

}


// File: Q14StudentTest.java
public class Q14StudentTest {
    /**
     * Write down in this class your solution and
     * its explanation of question no. 4 above.
     * Remember how to comment properly.
     */
    public static void main(String[] args) {
        // Test your class: Q14Student in here
    }
}
```

5. [**20 points**] In Lecture #3: Exercises, Ex2. Conversion, we wrote a program that converts masses given in the imperial system into the metric system. Write a program that can deal with weights given in pounds. Make use of the conversion:

   1 pound = 0.45359237 kilograms

Define a `Q15Weight` class, and write a constructor `Q15Weight(double p)` to generate a weight given in pounds. Implement methods `getPounds` and `getKilograms` which return the weight in pounds and kilograms, respectively.

```java
// File: Q15Weight.java
public class Q15Weight {
    /**
     * Write down in this class your solution and
     * its explanation of question no. 5 above.
     * Remember how to comment properly.
     */
    // Define your properties/fields here

    // Define your constructor here

    // Define your methods/functions in here

}
```

```java
// File: Q15WeightTest.java
public class Q15WeightTest {
    /**
     * Write down in this class your solution and
     * its explanation of question no. 5 above.
     * Remember how to comment properly.
     */
    public static void main(String[] args) {
        // Test your class: Q15Weight in here
    }
}
```

6. [**20 points**] In no. 1 we looked at the addition and multiplication of two fractions. In this assignment, fractions should be represented by a Java class `Q16Fraction`. You have to define this class, which in addition to the constructor and the accessor methods `getNumerator` and `getDenominator` have methods `toString` (to print a rational number) as well as `getSum` and `getProduct`, which generate the sum and the product of fractions and return the corresponding object of Class `Q16Fraction`. For instance, if we generate fractions `Q16Fraction f1 = new Q16Fraction(1, 2);` and `Q16Fraction f2 = new Q16Fraction(3, 7)` ; then
   - `f1.toString()` should return the string `"1/2"`;
   - `System.out.print(f2.getProduct(f1));` should print 3/14 (which corresponds to the product of `f2` and `f1`); and
   - `System.out.print(f2.getSum(f1));` should print 13/14 (which corresponds to the sum of `f2` and `f1`).

```java
// File: Q16Fraction.java
public class Q16Fraction {
    /**
     * Write down in this class your solution and
     * its explanation of question no. 6 above.
     * Remember how to comment properly.
     */
    // Define your properties/fields here

    // Define your constructor here

    // Define your methods/functions in here

}


// File: Q16FractionTest.java
public class Q16FractionTest {
    /**
     * Write down in this class your solution and
     * its explanation of question no. 6 above.
     * Remember how to comment properly.
     */
    public static void main(String[] args) {
        // Test your class: Q16Fraction in here
    }
}
```

7. [**20 points**] Please define the class ***Jokes***. There will be a variable of type *string* namely *joke_string* to store the joke. There are 2 methods in this class. *is_funny()* will determine if the joke is funny or not. If the length of the joke is under 20 characters (including space) the joke is funny, otherwise, the joke is not funny. Print "It's funny" if the joke is funny and "It's not funny" if the joke is not funny. The second method is *first_word*(). The method will print the first word of the joke separated by a blank space. For example, the *joke_string* is "Hello there". When we call the *is_funny()* method, it will print "It's funny", and when we call the *first_word()* method, it will print "Hello".

Test your class for: "Knock knock who's there?" and "Your life"

```java
//File Jokes.java
public class Jokes {
    /**
     * Write down in this class the solution
     * with its explanation of the question
     * Remember to comment properly
     */
    //String of the joke
    private static String joke_string;

    //Function to determine if a joke is funny or not
    //If the length of the joke is < 20, it's funny. Otherwise, it is
not funny
    public static void is_funny() {
        //Your code here
    }

    //Function to get the first word of the joke, separated by space
    //Many methods can be used here
    // as long as it splits the string by a blank space " "
    public static void first_word(){
        //Your code here
    }

    //To test the program
    //Please don't alter this block of code
    //To make it easier to test your program
    public static void main(String[] args) {
        joke_string = "Knock knock who's there?";
        is_funny();
        first_word();
        joke_string = "Your life";
        is_funny();
        first_word();
    }
}
```

8. To avoid plagiarism/cheating, every student needs to pledge and declare, then she/he must submit her/his **signed pledge and declaration** as in the following. Failing to do so will result in getting a 0 (zero) grade. Attach the **scanned/photo** of your *declaration* to your report.

"By the name of Allah (God) Almighty, herewith I pledge and truly declare that I have solved quiz 1 by myself, didn't do any cheating by any means, didn't do any plagiarism, and didn't accept anybody's help by any means. I am going to accept all of the consequences by any means if it has proven that I have done any cheating and/or plagiarism."

[Place, e.g., Surabaya], [date, e.g., 28 September 2023]

<Signed>

[Full name, e.g., Kinanti Wedariningsih]
[StudentID, e.g., 05112240000xxx]

9. Have a nice day, guys! Good luck! 😊