WIT COMP1000 Computer Science I

Prof. Thai

Student:

**Lab7: Greatest Common Divisor**

1. **GcdEuclidean** (GcdEuclidean.java)

First write a method called gcd() to calculate the greatest common divisor (GCD) of two positive integers using Euclidean algorithm. Then write a main() method that requests two positive integers from the user, validates the input, calls the gcd() method to compute the GCD, and outputs the return value of the gcd method() (all user input and output should be done in main()).

Check Wikipedia to find more information about GCDs and Euclidean algorithm. Below is a pseudocode for calculating the GCD, which you might find it useful:

method gcd(a, b) {

while (b ≠ 0) {

t = b

b = a mod b

a = t

}

return a

}

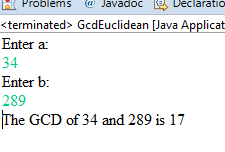
Test your program with the following data:

Enter a: 34

Enter b: 289

The GCD of 34 and 289 is 17

**TAKE A SCREENSHOT** of the console window showing the above sample run.



1. **BodyMassIndex** (BodyMassIndex.java)

The body mass index (BMI) is a measure of a person’s health status based on the person’s weight and height. It is calculated by taking a person’s weight in kilograms and dividing it by the square of person’s height in meters:

The interpretation of the BMI for people 16 years and older is as following:

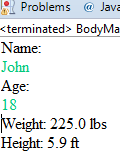
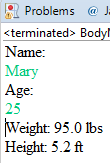
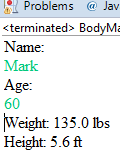
|  |  |
| --- | --- |
| **BMI** | **INTERPRETATION** |
| Below 18.5 | Underweight |
| 18.5 – 24.9 | Normal |
| 25.0 – 29.9 | Overweight |
| Above 30.0 | Obese |

1. Write a method called bmi() to calculate the body mass index (BMI). The interface of the bmi() is:

*double bmi(double weight, double height)*

1. You need to convert the units from English to Metric measurements. Write two methods called lbsToKg() and ftToM() to do the conversions. You are not allowed to use library functions to do the conversions.
2. Write a method called healthStatus() to take in appropriate parameters (name, age, weight, height) and display the health status of the person, i.e., Normal, Obese, etc.
   * Note: you need to check for correct user inputs
3. Write a main() method to integrate the above methods and allow the user to run the test multiple times.
4. Run the following test cases to verify your program.
5. Test case 1:
   * Name: John
   * Age: 18
   * Weight: 255 lbs
   * Height: 5.9 ft
6. Test case 2:
   * Name: Mary
   * Age: 25
   * Weight: 95 lbs
   * Height: 5.2 ft
7. Test case 3:
   * Name: Mark
   * Age: 60
   * Weight: 135 lbs
   * Height: 5.6 ft
8. Before coding, think how you are going to tackle this problem, and write a brief description of the logic of your program. **INSERT YOUR DESCRIPTION HERE.**

I am going to write the main method at the beginning. And, I will store all the general information in Main(). For instance: Age and Name. Then, I will proceed to the creation of the methods mentioned above and their parameters; ibm(), lbsToKg(),ftToM (), and healthStatus(). After that, I will call those methods in main and pass the right parameters to call the methods.

1. **TAKE A SCREENSHOT** of your input and output and paste them here. Do not paste your source code in this document.
2. Submit your source code and this document to Blackboard for grading.