Student Name: Andrew Downs

Class and Section: COP 3330 0R02

Total Points (100) \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Due: Saturday January 25, 2014 at 11:59 pm**

**Programming Assignment 1: Hearth Rate Calculator**

COP 3330 Object Oriented Programming – Spring 2014

**Problem Description:**

(Target- Heart- Rate Calculator) While exercising, you can use a heart- rate monitor to see that your heart rate stays within a safe range suggested by your trainers and doctors. According to the American Heart Association (AHA) (www. americanheart. org/ presenter. jhtml? identifier= www.americanheart.org/presenter.jhtml?identifier= 4736), the formula for calculating your maximum heart rate in beats per minute is 220 minus your age in years. Your target heart rate is a range that’s 50– 85% of your maximum heart rate.

[Note: These formulas are estimates provided by the AHA. Maximum and target heart rates may vary based on the health, fitness and gender of the individual. Always consult a physician or qualified health care professional before beginning or modifying an exercise program.]

Create a class called HeartRates. The class attributes should include the person’s first name, last name, date of birth (consisting of separate attributes for the month, day and year of birth) and current year. Your class should have a constructor that receives this data as parameters. For each attribute provide set and get methods. The class also should include a method that calculates and returns the person’s age (in years), a method that calculates and returns the person’s maximum heart rate and two methods that calculates and returns the person’s maximum and minimum of target heart rates.

Write a Java application that prompts for the person’s information, current year, instantiates an object of class HeartRates and prints the information from that object— including the person’s first name, last name and date of birth— then calculates and prints the person’s age in (years), maximum heart rate and target-heart-rate range.

**Analysis: - 10 points**

(Describe the problem including input and output in your own words.)

This program will prompt for and take the input of a person’s name (first and last), date of birth (year, month, and day), and the current year. This info will be put into an object of type HeartRates. The class HeartRates will have several methods that will get and set the object’s attributes and calculate the following values: age, maximum heart rate, maximum target heart rate, and minimum target heart rate. Once the user has entered the data, the program will print out the person’s name, age, date of birth, maximum heart rate, and their target range heart rate.

**Design: - 10 points**

(Describe the major steps for solving the problem. Create a UML diagram to accompany your major steps).

|  |
| --- |
| HeartRates |
| - firstName: String  - lastName: String  - birthYear: int  - birthMonth: int  - birthday: int  - currentYear: int  - minTargetCoefficient: float = 0.5  - maxTargetCoefficient: float = 0.85 |
| <<constructor>> HeartRates( name : String, surname : String, year : int, month : int, day : int, presentYear : int )  + GetAge() : int  + GetMaxHeartRate() : int  + GetMaxTargetHeartRate() : float  + GetMinTargetHeartRate() : float  + SetFirstName( name : String )  + SetLastName( name : String )  + SetBirthYear( year : int )  + SetBirthMonth ( month : int )  + SetBirthDay ( day : int )  + SetCurrentYear ( year : int )  + GetFirstName() : String  + GetLastName() : String  + GetBirthYear() : int  + GetBirthMonth() : int  + GetBirthDay() : int  + GetCurrentYear() : int |

Design cont. :

The first step of the assignment was to create the class HeartRates by first creating the instance variables and their get and set methods. I chose to create the two coefficient variables that would determine the person’s target heart range instead of directly using their values because it would make modifying a future version of the program much easier. For example, if those coefficients had been used in multiple methods and were proven by a study to be incorrect, they could be easily changed at the top of the code instead of having to search through it.

Next, I created the contructor, age and heart rate get methods. For the target heart rate methods, because of how integer multiplication with a float works, the return type had to be a float instead of an integer. I could have used a typecast declaration here to keep the result as an int but decided against it as I felt it decreased readability.

After this, I created the class HeartMain which would house the main method for the program. In this method the program would prompt for a user’s input, store that input in temporary variables, create an object of HeartRates using those temporary variables, and then print out the content of that object.

**Testing**: **- 10 points** (Describe how you tested, or will test this program)

To test this program I used various names, ages, and current years to make certain the methods functioned properly. Because the get and set methods for the individual attributes of the class were not needed for the scope of the program, I made a second main class that tested those methods by changing the attributes and printing out the results.

**Submit the following items:**

1. Save this Word file; submit it via Webcourses Assignments on or before the due date and time.

2. Compile, Run, and Submit your .java file(s) to Webcourses. You must submit the program and the report regardless whether it is complete or incomplete, correct or incorrect.