COIT20245

Assignment 1

# Assessment Task

You are to develop a Java console application using the NetBeans 8.2 IDE. The application is to calculate and classify MAP (Mean Arterial Pressure) for a randomised subset of individuals in a clinical trial. The number of individuals in the subset is not fixed, but will be in the range 5-10.

MAP is determined from an individual’s blood pressure using the following formula:

MAP = 1.0/3.0 \* SBP + 2.0/3.0 \* DBP

where

* MAP is the mean arterial pressure,
* SBP is the systolic blood pressure, and
* DBP is the diastolic blood pressure.

Normal blood pressure for a healthy adult is typically between 90 mmHg and 120 mmHg for systolic blood pressure and between 60 mmHg and 80 for diastolic blood pressure. The normal mean arterial pressure in healthy patients should be in the range of between 70 and 100 mmHg.

For each person in the trial, you are to enter their identifier, together with their SBP (in mmHg) and DBP (in mmHg). SBP is always greater than DBP; if this is not the case, an error message is to be displayed and no processing is to be performed for that particular record. The identifier is an integer in the range 1-100. Blood pressures are to be stored in variables of type double. The application is to then calculate and display the trial identifier and the MAP value, together with a message indicating whether the MAP value is high, normal or low. When all data has been entered, a summary report consisting of

1. The lowest MAP value
2. The highest MAP value
3. The average MAP value

is to be displayed. Values are to be displayed to one decimal place, e.g. 10.1, NOT 10.123456789.

Only language constructs covered in weeks 1-6 are to be used. In particular, the use of arrays will be penalised.

The application is to use the following classes and methods:

public class MAPCalculator {

public MAPCalculator() {

// no code required

}

public double value(double sbp, double dbp) {

// return MAP value

}

public String category(double map) {

// return one of “high”, “normal” or “low”

}

}

public class Assignment1 {

public static void main(String[] args) {

// create scanner object

// create MAP calculator

// read and process records

// display summary results

}

}

The classes are to be included in your NetBeans project in two separate source files – Assignment1.java and MAPCalculator.java.

# Submission

You are to submit a zipped folder containing

* Your zipped NetBeans project folder. Details of how to do this are available in the NetBeans FAQ on the unit website.
* Report.docx. This file is to contain the following Sections:

1. Limitations
2. Test plan
3. Test results

Student name, student ID number, unit name and unit code are to be included on the title page., a test plan and test results.

The limitations section is to specify any limitations that your program has in terms of calculations and data validation. With respect to data validation, note that the only validation that you are **required** to do is to check whether the SBP is greater than the DBP.

The test plan is to contain a comprehensive list of program functionality to be tested, the input values to be used to test each item of functionality, the expected output from the test and the actual output from the test. Note that because MAP is calculated as a floating point value, we can’t (in general) test it for equality. What this means in terms of your test plan is that you do **not** need to test the category end points

The test results section is to contain screenshots to demonstrate that the program generates the actual outputs shown in the test plan.

# Marking Criteria

|  |  |  |
| --- | --- | --- |
|  | **Criteria** | **Marks Allocated** |
| **1.** | **Calculation of results** | /3 |
| **1** | **Language use** | /7 |
|  | Variable declarations |  |
|  | Class definition/object creation |  |
|  | Loop control |  |
|  | Selection statement use |  |
|  | Method definition and use |  |
|  | Reading input |  |
|  | Display of results |  |
| **2** | **Layout and documentation** |  |
|  | Spacing and indentation conventions | /1 |
|  | Naming conventions | /1 |
|  | Comments | /3 |
| 3 | **Report** |  |
|  | Limitations | /1 |
|  | Test Plan and Test Results | /4 |
|  | **Sub-Total** | /20 |
|  | **Penalties** |  |
|  | Does not compile: up to 20 marks |  |
|  | Partial implementation: up to 20 marks |  |
|  | Late submission : 5% (1 mark) / day or part of a day |  |
|  | **Total** | /20 |

For Language Use, a checklist is given, but not a detailed marks breakdown. For checklist item non-compliance, you will be penalised 0.5 – 1 mark depending on the degree of non-compliance.

Note that it is your responsibility to ensure that source code files are included in your submission as well as report.docx. If either are missing, the assignment will not be marked.