**Step test application – It is far easier to breakdown any problem into smaller, more manageable tasks so I have created this document to give you a possible methodology to consider.**

What data do you need to collect for a testing session?

Operator / Tester

Step Height

Other?

The step height will determine the X values on a graph so how will you get the selection? The options available are 15cm, 20cm, 25cm or 30cm. You could use a dropdown box / radio buttons / individual buttons.

**Step 1 – get the operator initials and step height**

For example, I could use a textbox for the operator and radio buttons within a groupBox with one for each step height



Each radioButton has a default event called radioButtonName\_CheckedChanged() so you could use that to initialise an array or other object. You could have an Array at class level

public partial class Form1 : Form

{

decimal[] X\_Values = null;

…….

…….

And then when a radio button is selected, populate that array

rbt15\_CheckedChanged(object sender, EventArgs e)

{

X\_Values = new decimal[] { 11, 14, 18, 21, 25 };

}

Now that you have initialised the array you will be able to use the step values whenever you want in your application. For example, if I want the first value I could use:

X\_Values[0];

Or initialise a variable int x = X\_Values[0];

Step 1 complete -----------------------------------------------------------------------------------------------------

Step 2: Store the values for each test

User Name, User Age, User Gender

Once I have the user’s age I can initialise the class level variables for

Max Heartrate, HeartRate85 and HeartRate50

Step 2 complete -------------------------------------------------------------------------------------------------------

Step 3: get the values for each test heartrate – the operator may enter up to 5 values depending on how many tests were performed

You could store the values in another array or other object. These values would be the Y values if you were plotting the points on a graph.

Remember: Each Y value has a corresponding X value so if the y values were stored in an array then

X\_value[0] and Y\_value[0] would be a pair. You could have 5 valid pairs but you may have less if the number of tests competed by the user was less.

Step 3 complete -----------------------------------------------------------------------------------------------------------

Step 4: You need to ensure that you are working with VALID pairs only. You can now calculate the line of best fit using the equation of a straight line.

Equations of straight lines are in the form y = mx + b (m and b are numbers). m is the gradient of the line and b is the y-intercept (where the graph crosses the y-axis)

You need to calculate ‘m’ and ‘b’ so the worksheet on Moodle will help you do this.

Once you have m and b then y = max heartrate

So you will have values for Y (Max heartrate), M (Gradient) and B (y-intercept)

The final value you need to calculate is X which can be done using the rules of algebra.

Future steps include storing the data in your database and using a list of names to be tested but I suggest you focus on getting the application working and completing a test before thinking about the later tasks.