



Université d'Ottawa • University of Ottawa

**Faculty of Engineering**

## **GNG 1106: Fundamentals of Engineering Computation**

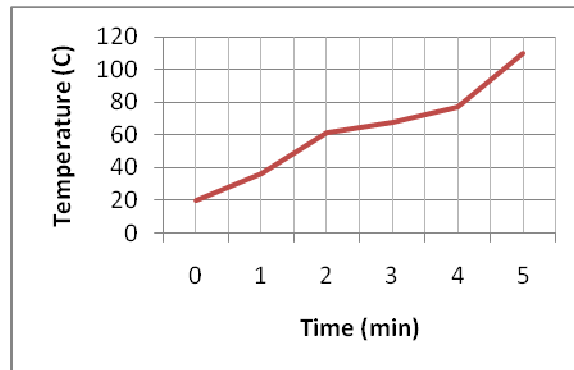
### **Assignment 3: Linear Regression**

**Report Due date: Monday 11 March by noon**

#### **Problem**

The temperature of a car engine has been measured, from real-world experiments, as shown in the table and graph below:

<u>Time (min)</u>	<u>Temperature (°C)</u>
0	20
1	36
2	61
3	68
4	77
5	110



Use linear regression to find the engine's temperature at 1.5 minutes, 4.3 minutes, and any other time specified by the user.

#### **Background**

In engineering, many times we measure several data points in an experiment, but then we need to predict a value that we have not measured which lies between two measured values, such as the problem statement above. If the relation between the measured parameters seems to be roughly linear, then we can use [linear regression](#) to find the relationship between those parameters. In the graph of the problem statement above, the relation seems to be roughly linear. Hence, we can apply linear regression to the above problem.

Assuming  $y \{y_0, y_1, \dots, y_{n-1}\}$  has a linear relation with  $x \{x_0, x_1, \dots, x_{n-1}\}$ , we can say that:

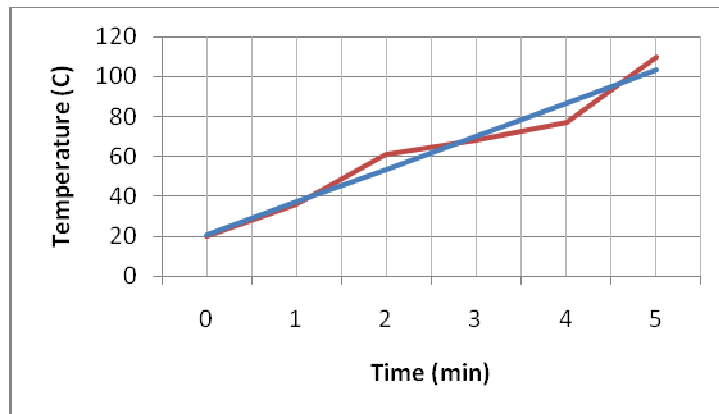
$$y = mx + b$$

where  $m$  and  $b$  can be found with linear regression as follows:

$$m = \frac{\sum_{k=1}^n x_k \sum_{k=1}^n y_k - n \sum_{k=1}^n x_k y_k}{\left(\sum_{k=1}^n x_k\right)^2 - n \sum_{k=1}^n x_k^2}$$

$$b = \frac{\sum_{k=1}^n x_k \sum_{k=1}^n x_k y_k - \sum_{k=1}^n x_k^2 \sum_{k=1}^n y_k}{\left(\sum_{k=1}^n x_k\right)^2 - n \sum_{k=1}^n x_k^2}$$

For the problem in this lab, using linear regression gives us the following line (in blue) compared to the measured curve (in red). As you can see, there is usually a difference between the measured values and the estimated (predicted) values. What linear regression does is to minimize those differences and still give us a straight line (blue).



Other methods, such as non-linear regression, are also possible to achieve higher accuracy and better curve fitting.

## Requirements

Your program should first print the table of the temperatures similar to the way it's printed in the problem statement. It should then calculate the temperature at minute 1.5 and 4.3 and show the answers to the user. Next, it should prompt the user to enter a time in minutes (or -1 to quit), and after reading the user's specified time it should give the value of the engine's temperature at that time. It should then go back to the prompt.

## Hints

- Use a one dimensional array to store the temperature values given in the problem statement.
- Use functions to separate tasks such as calculating  $m$ , calculating  $b$ , calculating the temperature at a given time, printing the prompt, etc. You can then give your algorithm as well as you pseudo code per function, as opposed to one large algorithm diagram or one large sequence of pseudo code.