# **ECOR 1010 – Introduction to Engineering**

**Dorian Wang**

**Assignment 8**

**Assignment Title: Programming with Matlab**

***TO Marking TA:***

Name: ...

Email: [...@....ca](mailto:...@....ca)

**Lab Section: L14**

**Room: 4301CB – Carleton University**

**FROM:**

Email: [dorianwang@cmail.carleton.ca](mailto:dorianwang@cmail.carleton.ca)

Student Number: 101009020

Number of Figures and Tables (Including handwritten ones): ...

Last Date and Time of Revision: DD/MM/YYYY @ HH:MM (AM/PM)

# Introduction

Matlab may be used as a simple programming language. It is faster to code than many other languages, and allows simple code for manipulating matrices and vectors.

# Materials and Methods

Matlab

# Results

Results go here as text...

Results go here as text...

Results go here as text...

Results go here as text...

Results go here as text...

Results go here as text...

# Discussion

Add you discussion as text here ...

Add you discussion as text here ...

Add you discussion as text here ...

Add you discussion as text here ...

Add you discussion as text here ...

Add you discussion as text here ...

# Conclusions

Add you conclusion as text here ...

Add you conclusion as text here ...

Add you conclusion as text here ...

Add you conclusion as text here ...

Add you conclusion as text here ...

# APPENDIces- Figures and Tables

|  |
| --- |
| function Q2 = s(x, y)  xDim = size(x);  yDim = size(y);    if (xDim(1, 2) ~= 1)  if (xDim(1, 1) == 1)  x = transpose(x);  else  'The input is not correct.'  return  end  end    if (yDim(1, 2) ~= 1)  if (yDim(1, 1) == 1)  y = transpose(y);  else  'The input is not correct.'  return  end  end    if (size(x) ~= size(y))  'Input is incompatable.'  return  end    xy2 = [];  disp(size(x))  disp(size(y))  for n = 1:size(x)  disp(n)  xy2 = [xy2; x(n, 1)\*(y(n, 1)^2)];  end    disp(' x y xy^2')  answ = [x, y, xy2];  disp(answ)    end |

**Figure 1.**

|  |
| --- |
| for i = -40:0.1:20  if (i >= 9.0)  plot(i, 1.5\*sqrt(4\*i)+10, '\*'); hold on;  elseif((i < 9.0) && (i >= 0.0))  plot(i, 38/(11 - i), '\*'); hold on;  else  plot(i, i\*sin(i) + 38/11, '\*'); hold on;  end  end      xlabel('x')  ylabel('y')  title('Dorian Wang Matlab plot of piecewise function') |

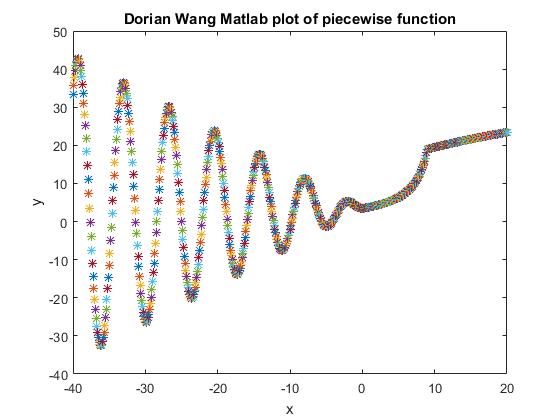
**Figure 2.**

|  |
| --- |
| function Q4 = y(distance, fuel)    sizeD = size(distance);    KMpLToMpG = 2.35214583;    if ((size(distance) ~= size(fuel)) | (sizeD(1, 2) ~= 1))  disp('Input is not the correct size.')  return %error  end    KmPer100L = zeros(sizeD(1,1), sizeD(1,2));    for n = 1:sizeD(1,1)  KmPer100L(n, 1) = distance(n, 1)/fuel(n, 1) \* 100.0;  end    MlPerG = zeros(sizeD(1,1), sizeD(1,2));    for n = 1:sizeD(1,1)  MlPerG(n, 1) = KmPer100L(n, 1) \* KMpLToMpG / 100.0;  end    months = zeros(sizeD(1, 1), 1);  for n = 1:sizeD(1, 1)  months(n, 1) = n;  end    month = months;  MilesPerGallon = MlPerG;  fuelUsed = fuel;  distanceTraveled = distance;    T = table(month, distanceTraveled, fuelUsed, MilesPerGallon, KmPer100L);    disp(T)  fprintf('The average fuel efficiency is %f MPG.\n', sum(MilesPerGallon(:))/ sizeD(1, 1));  fprintf('The average fuel consumption is %f L/100 km.\n', sum(KmPer100L(:))/ sizeD(1, 1));    end |

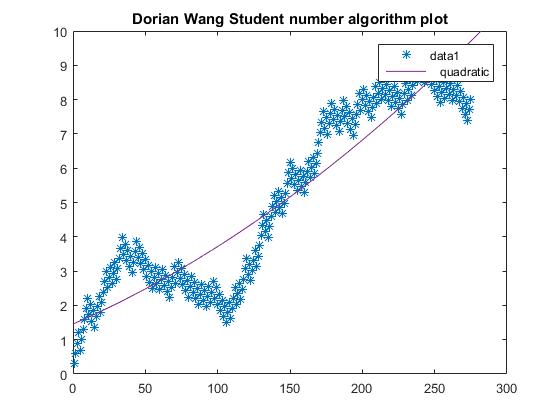
**Figure 3.**

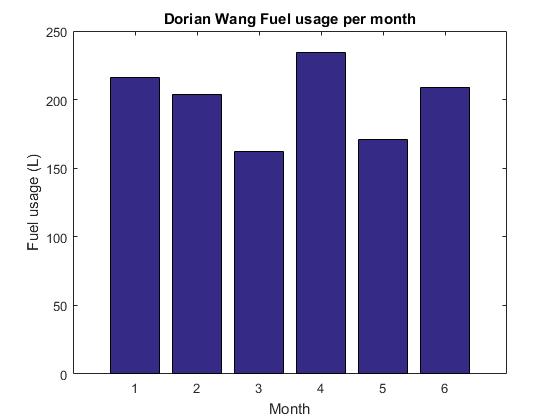
|  |
| --- |
| function Q5 = y(input)    if size(input) ~= size(1)  return % Error  end    inputLog = zeros(275, 1);  count = 0;    while input ~= 1  count = count + 1;  inputLog(count, 1) = input;  if mod(input, 2) == 0  input = input / 2;  else  input = input \* 3 + 1;  end  end    revSeries = flipud(inputLog);  revSeries = log10(revSeries);    plot((1:275), revSeries, '\*');    end |

**Figure 4.**



**Figure 5.**

**Figure 6.**

****

**Figure 7.**