Introduction to MATLAB

Introductory Problems

1. Create a vector that contains the numbers 1, 2, 3, 4, 5:
   \[
   \text{vector} = [1 \ 2 \ 3 \ 4 \ 5];
   \]
   \[
   \text{vector} = [1:5];
   \]

2. Clear the Workspace and Command windows:
   \[
   \text{clear}
   \]
   \[
   \text{clc}
   \]

3. Create a matrix with the same contents as the following table:

   \[
   \begin{array}{ccc}
   4 & 5 & 6 \\
   10 & 50 & 3 \\
   15 & 56 & 20 \\
   \end{array}
   \]

   \[
   \text{Matrix} = [4 \ 5 \ 6; 10 \ 50 \ 3; 15 \ 56 \ 20];
   \]

   Alternatively:

   \[
   \begin{array}{l}
   1 \ V1 = [4 \ 5 \ 6]; \\
   2 \ V2 = [10 \ 50 \ 3]; \\
   3 \ V3 = [15 \ 56 \ 20]; \\
   4 \ \text{Matrix1} = [V1; V2; V3]; \\
   5 \ \text{\% The semicolon between V1, V2 and V3 specify that we want to insert V2 ...} \\
   \text{\quad and V3 into new rows of the matrix. Moreover, Matrix} = [V1, V2, V3] ... \\
   \text{\quad would insert all variables into the same row.}
   \end{array}
   \]

4. Transpose the matrix from (3) using an inverted comma ('):

   \[
   \begin{array}{ccc}
   4 & 10 & 15 \\
   5 & 50 & 56 \\
   6 & 3 & 20 \\
   \end{array}
   \]

   \[
   \text{Matrix2} = [4 \ 5 \ 6; 10 \ 50 \ 3; 15 \ 56 \ 20]' ;
   \]
   Or
   \[
   \text{TransposedMatrix} = \text{Matrix}';
   \]

   **Note:** You cannot use spaces in variable names, instead remove the spaces or use an underscore.
5. Vertically concatenate matrices 1 and 2:
Matrix3 = cat(1, Matrix1, Matrix2);

6. Horizontally concatenate matrices 1 and 2:
Matrix4 = cat(2, Matrix1, Matrix2);

7. Create a cell array that contains matrix 1 in slice 1 and matrix 2 in slice 2:

```
1
CellArray = zeros(3, 3, 2);
2
CellArray(:, :, 1) = Matrix1;
3
CellArray(:, :, 2) = Matrix2;
4
5
% zeros creates an empty matrix, as given:
6
zeros(Number Columns, Number Rows, Number Slices)
7
8
% We use a colon in the second and third lines to select all elements of ...
9
each column and row.
```

8. Determine the third power of each element within vector (a):
a = [4 6 7].^3

9. Multiply each element of vector (a) by the respective element in vector (b):

```
1
a = [4 6 7];
2
b = [3 4 5];
3
c = a.*b;
```

10. Multiply each element of matrix 1 by the respective element in matrix 2

```
1
Matrix5 = Matrix1. * Matrix2;
2
% Removing the dot (.) will perform matrix multiplication, which ...
3
  calculates a very different result. Check for yourself. In these ...
4
  problems, we essentially use matrices as arrays to store data.
```

11. Create an empty 3 x 3 x 2 array and fill the centre of slices 1 and 2 with 6 and 7 respectively.

```
1
Array = zeros(3,3,2);
2
Array(2,2,1) = 6;
3
Array(2,2,2) = 7;
```
12. Insert numbers 5:13 into a 3 by 3 matrix using a for loop:

```matlab
TempMatrix = zeros(3,3);
for i = 1:9;
a = 4 + i;
TempMatrix(i) = a;
end
```

13. Insert numbers 5:13 into a 3 by 3 matrix using a while loop:

```matlab
TempMatrix = zeros(3,3);
i = 1;
a = 5;
while i < 10
    TempMatrix(i) = a + i;
i = i + 1;
end
```

14. Multiply each row of matrix 1 by the following factors using a nested loop.

<table>
<thead>
<tr>
<th>4</th>
<th>5</th>
<th>6</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>50</td>
<td>3</td>
</tr>
<tr>
<td>15</td>
<td>56</td>
<td>20</td>
</tr>
</tbody>
</table>

Row 1 Multiplication Factor = 7
Row 2 Multiplication Factor = 3
Row 3 Multiplication Factor = 4

```matlab
Matrix1 = [ 4 5 6; 10 50 3; 15 56 20];
NewMatrix = zeros(3,3);
RowNumber = 1;
ColumnNumber = 1;
RowFactor = 7;
for i = 1: 3
    for j = 1:3
        NewMatrix(RowNumber, ColumnNumber) = ...
        Matrix1(RowNumber, ColumnNumber) * RowFactor;
    end
    if RowNumber == 3
        RowFactor = 3;
    elseif RowNumber == 3 && ColumnNumber == 2;
```
Alternative Method:

```matlab
Matrix1 = [ 4 5 6; 10 50 3; 15 56 20];
NewMatrix = zeros(3,3);
RowNumber = 1;
ColumnNumber = 1;
while ColumnNumber < 4 && RowNumber < 4
    NewMatrix(RowNumber, ColumnNumber) = ...
    Matrix1(RowNumber, ColumnNumber) * RowFactor;
    RowNumber = RowNumber + 1;
    if RowNumber == 4;
        ColumnNumber = ColumnNumber + 1;
        RowNumber = 1;
        RowFactor = 3;
    elseif RowNumber == 1 && ColumnNumber == 3;
        RowFactor = 4;
    end
end
```

Problem Section 2

Problem 1
Determine the first 100 numbers in the fibonacci sequence and plot. The first two numbers in the sequence are:

\[
\begin{align*}
F_1 &= 0 \\
F_2 &= 1
\end{align*}
\]

Where \( F_n = F_{n-1} + F_{n-2} \)

```matlab
f0 = 0;
f1 = 1;
fibonacci_Sequence = zeros(100,1);
fibonacci_Sequence(1) = 0;
fibonacci_Sequence(2) = 1;
for i = 1:98
    fibonacci_Sequence(i+2,1) = fibonacci_Sequence(i+1,1) + ...
    fibonacci_Sequence(i,1);
```
Problem 2

Create a matrix, which contains the following in each column using for/while loop (where \( x = 1:100 \)). Plot the functions on the same graph and try the subplot function.

Note: You must use the command: hold on to plot several functions on the same graph. Use help subplot in the command window

Column 1: \( y = x \)
Column 2: \( y = x^2 \)
Column 3: \( y = x^3 \)
Column 4: \( y = x^4 \)
Problem 3

Create a matrix, which contains the following in each column using a nested loop (where \( x = 1:100 \))

**Column 1:** \( y = x \)
**Column 2:** \( y = x^2 \)
**Column 3:** \( y = x^3 \)
**Column 4:** \( y = x^4 \)

```matlab
Matrix = zeros(20,4);
for j = 1:4
    for i = 1:20
        h = i;
        if j == 2
            h = i^2;
        elseif j == 3
            h = i^3;
        elseif j == 4
            h = i^4;
        end
        Matrix(i,j) = h;
    end
end
```

Problem 4

We have collected absorbance data for Cytochrome C. Import sample and control absorbance files into MATLAB. Write a function to perform a baseline correction, plot the baseline corrected absorbance, and calculate the concentration using the Beer Lambert Law

\[
A = e * l * c
\]

Pathlength = 1cm
Molar extinction Coefficient 280nm = 11585 M\(^{-1}\) cm\(^{-1}\)
Wavelength Range: 400 to 250 nm, 0.2 nm steps.

Note: Baseline Absorbance data has not been stored the same as Cytochrome C Absorbance data, you must either rearrange both datasets into columns or rows.
% In the Home Tab, select Import Data; import Sample.txt and ...
Control.txt files.

% Wavelength:
Baseline_Corrected_Data(:,1) = Wavelength;

% Absorbance:
Baseline_Corrected_Data(:,2) = Sample_Data - Control_Data;

assignin('base','Baseline_Corrected_Absorbance', ...
Baseline_Corrected_Data)

% Plot:
plot(Baseline_Corrected_Data(:,1), Baseline_Corrected_Data(:,2))
xlabel('wavelength')
ylabel('Absorbance')
title('Protein Concentration')

% Determine Concentration:
Number_Data_Points_below_280nm = (280-250)/0.2;
Abs_280nm = Baseline_Corrected_Data(Number_Data_Points_below_280nm, 2);
Concentration = (Abs_280nm / (1 * 11585)) * 1000000;

Output = sprintf('Absorbance at 280 nm =%d; Cytochrome C ... 
Concentration = %d micromolar', Abs_280nm, Concentration);

% Type the following in the Command Window:
ProteinConc(Wavelengthnm,CytochromeCAbs, ControlAbs)