Homework #6

Table of Contents

Jason Chiarulli

ENGR 108

Matlab for Engineers, Third Edition

```
Note: Some program ouput is not displayed since the publish feature

      does not work well when input is requested from the user.

      This output is located in a separate pdf.
```

Problem 8.3

```
% Defines a matrix of casting curing data
Casting_Curing_Data = [1 116 45 110
                       2 114 42 115
                       3 118 41 120
                       4 124 38 95
                       5 126 61 118];

% Parts a-c determine which batches did and did not meet the
% criterion for temperature, humidity, and pressure using
% the find command and logical operators

% a)
Temperature = Casting_Curing_Data(:,2);
Meet_Temperature_Criterion = find(Temperature>115 & Temperature<125)
Do_Not_Meet_Temperature_Criterion = find(Temperature<115 |...
                                         Temperature>125)

% b)
Humidity = Casting_Curing_Data(:,3);
Meet_Humidity_Criterion = find(Humidity>40 & Humidity<60)
Do_Not_Meet_Humidity_Criterion = find(Humidity<40 | Humidity>60)

% c)
Pressure = Casting_Curing_Data(:,4);
Meet_Pressure_Criterion = find(Pressure>100 & Pressure<200)
Do_Not_Meet_Pressure_Criterion = find(Pressure<100 | Pressure>200)
```

```
% d)
% Determines which batches did and did not meet all the criterion
% using the find command and logical operators
Passed_Batches = find(Temperature>115 & Temperature<125 &...
    Humidity>40 & Humidity<60 & Pressure>100 & Pressure<200)
Failed_Batches = find(Temperature<115 | Temperature>125 |...
    Humidity<40 | Humidity>60 | Pressure<100 | Pressure>200)

% e)
% Determines the number the total number of batches, the number
% of batches that passed, and the number of batches that failed
% The numbers are then used to calculate the failure rate and
% pass rate and is presented as a percentage
length_of_Batches = length(Casting_Curing_Data(:,1));
length_of_Passed_Batches = length(Passed_Batches);
length_of_Failed_Batches = length(Failed_Batches);
Failure_Rate = length_of_Failed_Batches/length_of_Batches*100
Passing_Rate = length_of_Passed_Batches/length_of_Batches*100


Meet_Temperature_Criterion =

     1
     3
     4


Do_Not_Meet_Temperature_Criterion =

     2
     5


Meet_Humidity_Criterion =

     1
     2
     3


Do_Not_Meet_Humidity_Criterion =

     4
     5


Meet_Pressure_Criterion =

     1
     2
     3
     5
```

```
Do_Not_Meet_Pressure_Criterion =

     4


Passed_Batches =

     1
     3


Failed_Batches =

     2
     4
     5


Failure_Rate =

    60


Passing_Rate =

    40
```

Problem 8.12

```
% Requests  input from the user
Temp = input('Enter the outside air temperature in Fahrenheit: ');
% Determines whether the user input is greater than or equal to 80,
% between 60 and 80, or equal to or below 60 and displays the
% corresponding message
if Temp >= 80
    disp('It''s hot outside you should wear shorts today.')
elseif Temp > 60 & Temp < 80
    disp('It''s a beautiful day')
else
    disp('It''s cold outside you should wear a jacket or coat.')
end
```

```
Error using input
Cannot call INPUT from EVALC.

Error in Jason_Chiarulli_Ch_8_and_9_HW (line 66)
Temp = input('Enter the outside air temperature in Fahrenheit: ');
```

Problem 8.16

```
% Creates a menu box on screen with a series of buttons
```

```
engineering_program = menu('Select an engineering program:
 ','Civil Engineering','Chemical Engineering','Computer
 Engineering','Electrical Engineering','Mechanical Engineering')
% Outputs text to the command line based off of which button
% the user selects in the menu
switch engineering_program
    case 1
        disp('Minimumn Number of Required Credits: 130')
    case 2
        disp('Minimumn Number of Required Credits: 130')
    case 3
        disp('Minimumn Number of Required Credits: 122')
    case 4
        disp('Minimumn Number of Required Credits: 126.5')
    case 5
        disp('Minimumn Number of Required Credits: 129')
end
```

Problem 9.1

```
% Defines a vector
x = [1 23 43 72 87 56 98 33];
% Defines a variable
total = 0;
% Executes the loop for each entry in the vector x
for k = 1:length(x)
    % Sets total equal to the previous total then adds the value
    % of each successive entry in the vector x to the total
    total = total + x(k);
end
% Outputs the sum of each entry in the vector
disp('The total is: ')
disp(total)
% Checks the answer from the loop with the sum function
sum_of_x = sum(x)
```

Problem 9.5

```
% Defines a list of all the prime numbers below 100
p = primes(100);
% Creates an array of zeros where the dimensions are set to 1 and
% one less than the length of the list of prime numbers < 100
a = zeros(1, length(p)-1);
% Executes the loop from the initial value of one to
% one less than the length of the list of prime numbers < 100
for k = 1:length(p)-1
    % Multiplies each adjacent value in the list of prime numbers
    % and stores the value in array 'a'
    a(k) = p(k)*p(k+1);
end
% Outputs the array
disp(a)
```

Problem 9.14

```
% Defines the starting parameters
y(1) = 1;
total(1) = y(1);
criterion = 0.001;
max_iterations = 10;

% Executes the loop
for k=2:max_iterations
    % Approximates the value of cos(2)
    y(k) = ((-1)^(k-1))*((k^((k-1)*2))/factorial((k-1)*2));
    % Calculates the value of the series
    total(k) = total(k-1) + y(k);
    % Determines when to break the loop based off the given error
    if(abs(total(k)-total(k-1))<criterion)
        break
    end
end

% Displays the number of iterations of the sequence and whether
% or not the sequence converged and the value of the series at that
% point
if k==max_iterations
    fprintf('The sequence did not converge in %5.0f iterations
 \n',max_iterations)
    fprintf('At which point the value of the series is %8.3f \n',
 total(k))
else
    fprintf('The sequence converged in %5.0f iterations \n',k)
    fprintf('The final element is equal to %8.3f \n',y(k))
    fprintf('At which point the value of the series is %8.3f \n',
 total(k))
end
```

Published with MATLAB® R2017a