# ECE 111 - Homework #1

Week #1: Matlab Introduction

## Bison Academy: Homework Sets & Solutions

1) What are the solutions to

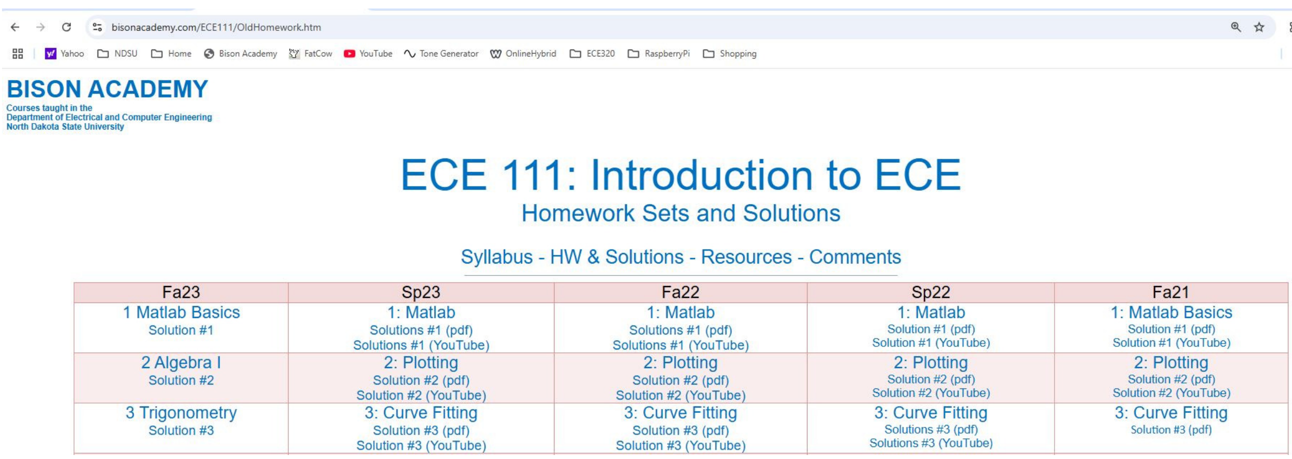
$$y = \sin(2x)$$

$$y = (x+1)(x-1)$$

hint: See homework #2, problem #2 solutions for Fall 2023

If you ever get stuck and need help on a homework set, Bison Academy has solutions to previous homework sets posted under

Bison Academy - Homework Sets and Solutions - Fall 2023



bisonacademy.com/ECE111/OldHomework.htm

**BISON ACADEMY**
Courses taught in the
Department of Electrical and Computer Engineering
North Dakota State University

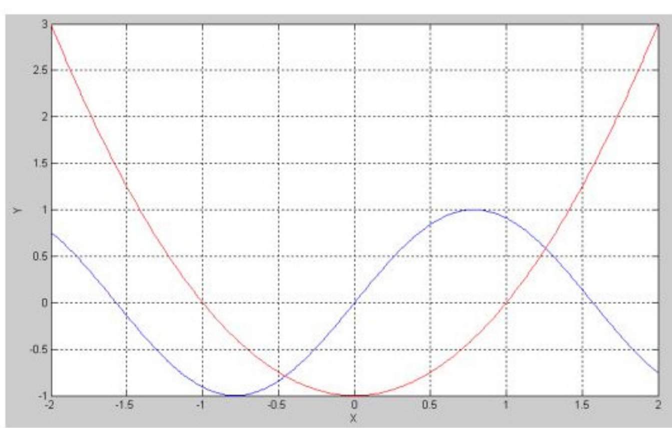
ECE 111: Introduction to ECE
Homework Sets and Solutions

Syllabus - HW & Solutions - Resources - Comments

| Fa23 | Sp23 | Fa22 | Sp22 | Fa21 |
|---|---|---|---|---|
| 1 Matlab Basics<br>Solution #1 | 1: Matlab<br>Solutions #1 (pdf)<br>Solutions #1 (YouTube) | 1: Matlab<br>Solutions #1 (pdf)<br>Solutions #1 (YouTube) | 1: Matlab<br>Solution #1 (pdf)<br>Solution #1 (YouTube) | 1: Matlab Basics<br>Solution #1 (pdf)<br>Solution #1 (YouTube) |
| 2 Algebra I<br>Solution #2 | 2: Plotting<br>Solution #2 (pdf)<br>Solution #2 (YouTube) | 2: Plotting<br>Solution #2 (pdf)<br>Solution #2 (YouTube) | 2: Plotting<br>Solution #2 (pdf)<br>Solution #2 (YouTube) | 2: Plotting<br>Solution #2 (pdf)<br>Solution #2 (YouTube) |
| 3 Trigonometry<br>Solution #3 | 3: Curve Fitting<br>Solution #3 (pdf)<br>Solution #3 (YouTube) | 3: Curve Fitting<br>Solution #3 (pdf)<br>Solution #3 (YouTube) | 3: Curve Fitting<br>Solutions #3 (pdf)<br>Solutions #3 (YouTube) | 3: Curve Fitting<br>Solution #3 (pdf) |

From the graph, there are two solutions:

- x = -0.4, y = .03
- x = +1.25, y = +0.6



## Roots to a Polynomial

2) Use the *roots()* command to find the roots to

a) $y = x^3 - 3x^2 - 64x + 192$

```
>> P = [1,-3,-64,192]

P =     1    -3   -64   192

>> roots(P)

   -8.0000
    8.0000
    3.0000
```

b) $y = x^4 - 15x^3 + 23x^2 + 315x - 324$

```
>> P = [1,-15,23,315,-324]

P =     1   -15    23   315  -324

>> roots(P)

    9.0000
    9.0000
   -4.0000
    1.0000
```

c) $y = x^5 - 17x^4 + 21x^3 + 837x^2 - 3402x$

```
>> P = [1,-17,21,837,-3402, 0]

P =             1          -17            21           837        -3402       9

>> roots(P)

  -7.0000
   9.0000 + 0.0000i
   9.0000 - 0.0000i
   6.0000
   0.0000
```

## Matlab as a Graphing Calculator: (CdS Light Sensor equations)

Assume a CdS light sensor and voltage divider have the following relationship:

$$R = 5000 \cdot (lux)^{-0.6} \Omega$$

$$V = \left( \frac{R}{R+500} \right) \cdot 5V$$

3) Determine the resistance and voltage if

Light = 30 Lux (dim room)

```
>> Lux = 30;
>> R = 5000 * ( Lux ^ (-0.6) )

R =  649.6766

>> V = R / (R + 500) * 5

V =    2.8255
```

Light = 100 Lux (typical room)

```
>> Lux = 100;
>> R = 5000 * ( Lux ^ (-0.6) )

R =  315.4787

>> V = R / (R + 500) * 5

V =    1.9343

>>
```

4) Plot the resistance vs. light level for 10 < Lux < 100. From the graph, determine

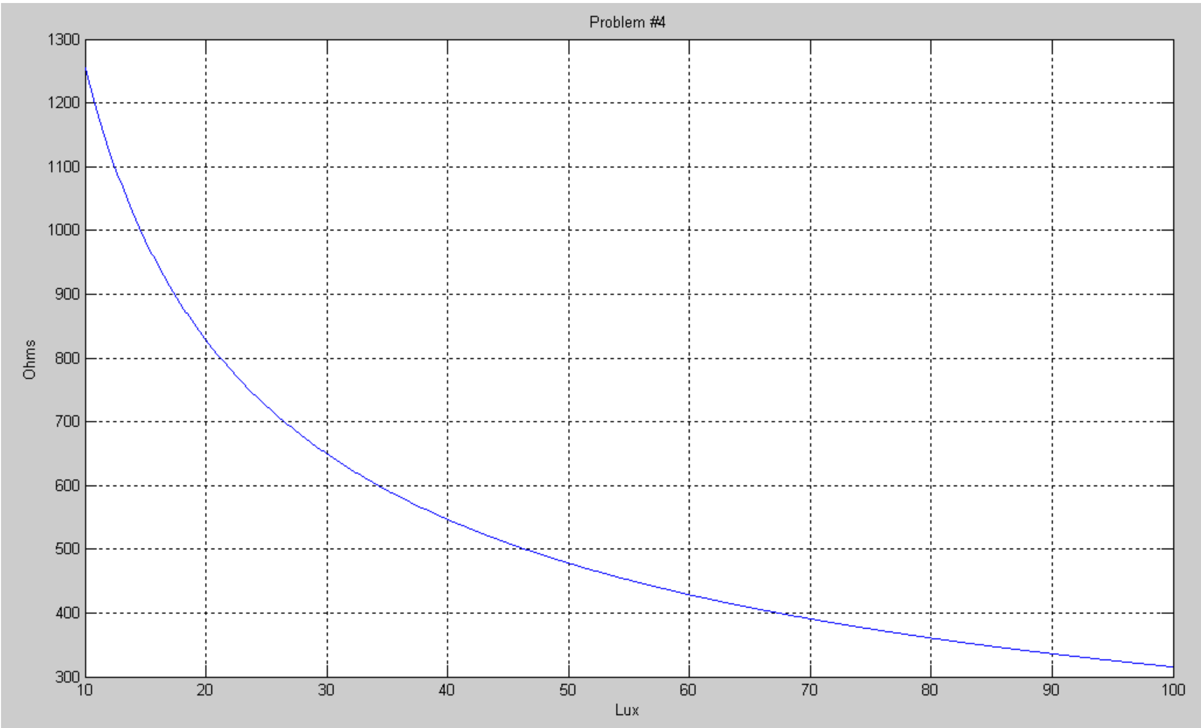
- The light level when R = 900 Ohms
- The light level when R = 600 Ohms

In Matlab

```
>> Lux = [10:0.1:100]';
>> R = 5000 * ( Lux .^ (-0.6) );
>> plot(Lux, R)
>> xlabel('Lux');
>> ylabel('Ohms');
>> title('Problem #4')
>> grid on
```

From the graph, zoom in to find the answer:

- When R = 900 Ohms, Light = 17.42 Lux
- When R = 600 Ohms, Lighe = 34.25 Lux



5) Plot the votlage vs. light level for 10 < Lux < 100.  From the graph, determine

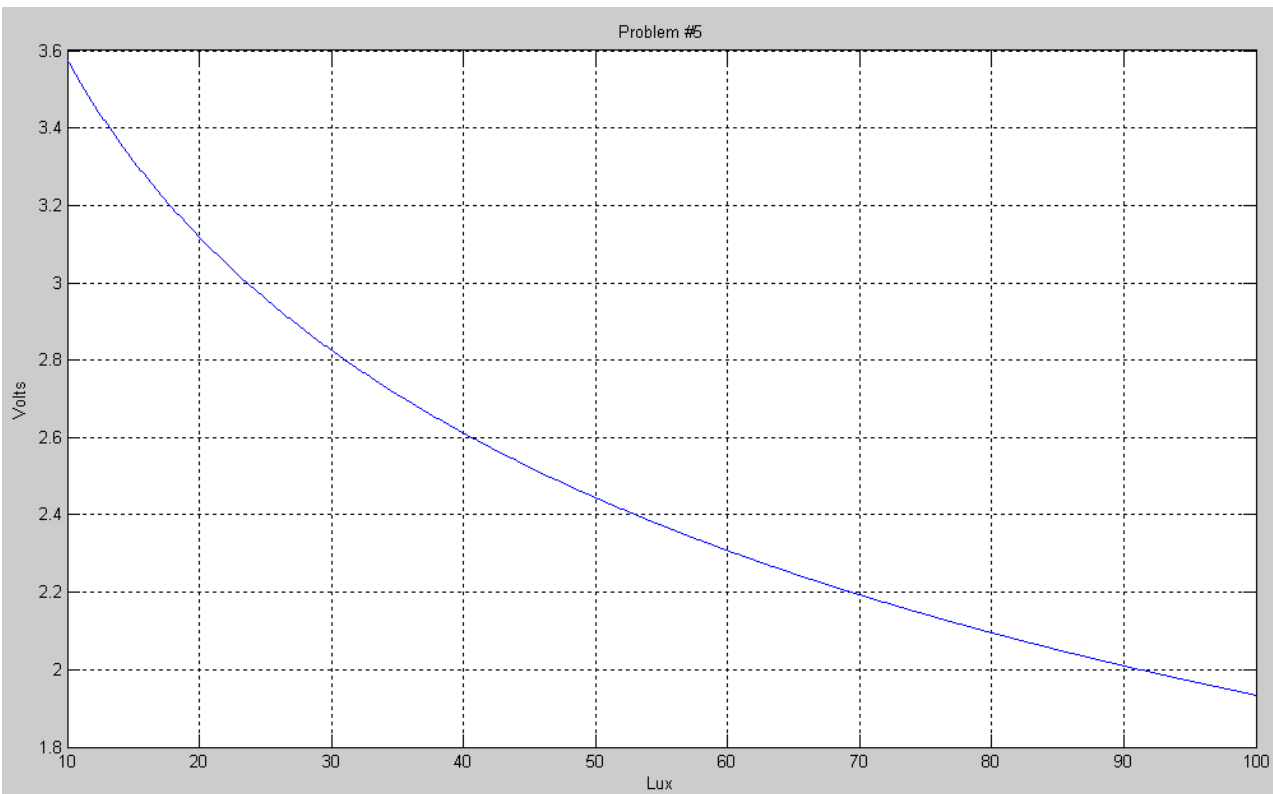
- The light level when V = 3.20 Volts
- The light level when V = 2.20 Volts

In Matlab

```
>> Lux = [10:0.1:100]';
>> R = 5000 * ( Lux .^ (-0.6) );
>> V = R ./ (R + 500) * 5;
>> plot(Lux, V);
>> xlabel('Lux');
>> ylabel('Volts');
>> title('Problem #5')
>> grid on
```

Zooming in on the graph

- 3.20V means 17.79 Lux
- 2.20V means 69.38 Lux



## For-Loops

6) A and B are playing a match consisting of 5 games. For each game

- A rolls eight 6-sided dice and takes the sum (A = 8d6)
- B rolls two 20-sided dice and takes the sum (B = 2d20).

Whoever has the higher total wins the game (A wins on ties). Determine the odds that A wins the match using a Monte-Carlo simulation with 100,000 games.

Start with playing a single game:

```
>> d6 = ceil(6*rand(8,1))'

d6 =     2     4     5     3     5     3     1     5

>> d20 = ceil(20*rand(2,1))'

d20 =    15    19

>> d6 = sum(d6)

d6 =    28

>> d20 = sum(d20)

d20 =    34
```

Once that works, play a match of 5 games to see who wins

```
W = 0;
T = 0;
L = 0;

A = 0;
B = 0;
    for i=1:5
       d6 = sum( ceil(6*rand(8,1)) );
       d20 = sum( ceil(20*rand(2,1)) );
       if(d6 > d20)
           A = A + 1;
       elseif(d6==d20)
           A = A + 0.5;
           B = B + 0.5;
       else
           B = B + 1;
       end
    end
if(A > B)
    W = W + 1;
elseif(A==B)
    T = T + 1;
else
    L = L + 1;
end
disp([A, B, W, T, L])
```

Check that this code works (run this as a script several times)

```
        A      B      W      T      L
        4      1      1      0      0
        3      2      1      0      0
        2      3      0      0      1
```

Looks good.  Now play 100,000 games

```
% 100,000 matches
W = 0;
T = 0;
L = 0;
for m=1:1e5
    % Single Match
    A = 0;
    B = 0;
    for i=1:5
       % Single Game
       d6 = sum( ceil(6*rand(8,1)) );
       d20 = sum( ceil(20*rand(2,1)) );
       if(d6 > d20)
           A = A + 1;
       elseif(d6==d20)
           A = A + 0.5;
           B = B + 0.5;
       else
           B = B + 1;
       end
    end
    if(A > B)
        W = W + 1;
    elseif(A==B)
        T = T + 1;
    else
        L = L + 1;
    end
end
disp([W, T, L])
```

| W | T | L |
|---|---|---|
| **89462** | **2644** | **7894** |

The results is

- 89.46% of the time A wins
- 2,64% of the time there is a tie
- 7.89% of the time B wins

7) A and B are playing a match consisting of 5 games. For each game,

- A has a 65% chance of winning (+1 point for A), and
- A has a 35% chance of losing (+1 point for B).

Determine the odds that A wins the match using a Monte-Carlo simulation with 100,000 games.

Use for-loops

- Play a single match of five games (j loop)
- A has a 65% chance of winning any given game
- Play and display the result after one game (i-loop)

Once you're convinced that the program is working, increase it to 100,000 games (i-loop):

```
% 100,000 matches
W = 0;
T = 0;
L = 0;
tic
% play a single match
for i=1:1e5
    A = 0;
    B = 0;
    for j=1:5
        % play a single game
        if(rand < 0.65)
            A = A + 1;
        else
            B = B + 1;
        end
    end
    if(A > B) W = W + 1; end
    if(A == B) T = T + 1; end
    if(A < B) L = L + 1; end
end
toc
disp([W,T,L])
```

```
Elapsed time is 0.088197 seconds.
      W            T       L
    76500          0    23500
```

Result:

- A wins 76.50% of the time
- There is never a tie
- B wins 23.50% of the time

## While-Loops

8) A and B are playing a match consisting of N games. For each game

- A rolls eight 6-sided dice and takes the sum (A = 8d6)
- B rolls two 20-sided dice and takes the sum (B = 2d20).

Whoever has the higher total wins the game (A wins on ties). The match is over when one player is up three games.

Determine the odds that A wins the match using a Monte-Carlo simulation with 100,000 games.

This is similar to problem #8, but instead of playing 5 games (a for-loop), keep playing while the difference in wins is less than 3 (a while loop).

Simply change one line of code:

```
% play 100,000 matches
W = 0;
T = 0;
L = 0;
for m=1:1e5
    % play one match
    A = 0;
    B = 0;
%     for i=1:5
    while(abs(A-B) < 3)
       % play one game
       d6 = sum( ceil(6*rand(8,1)) );
       d20 = sum( ceil(20*rand(2,1)) );
       if(d6 > d20)
           A = A + 1;
       elseif(d6==d20)
           A = A + 0.5;
           B = B + 0.5;
       else
           B = B + 1;
       end
    end
    if(A > B)
        W = W + 1;
    elseif(A==B)
        T = T + 1;
    else
        L = L + 1;
    end
end
disp([W, T, L])
```

Result

```
          W              T            L
        97308            0          2692
```

This changes the odds of A winning from 894% (problem 6) to 97.3% (problem 8)

9) A and B are playing a match consisting of N games. For each game,

- A has a 65% chance of winning (+1 point for A), and
- A has a 35% chance of losing (+1 point for B).

The match is over when one player is up three games.

Determine the odds that A wins the match using a Monte-Carlo simulation with 100,000 games.

This is similar to problem #7 except that

- Instead of playing 5 games (a for-loop),
- You play until the difference in wins is less than 3 (a while-loop)

Change the for-loop to a while-loop

```
% play 100,000 matches
W = 0;
T = 0;
L = 0;
tic
for i=1:1e5
    % play one match
    A = 0;
    B = 0;
    % for i=1:5
    while( abs(A - B) < 3)
        % play one game
        if(rand < 0.65)
            A = A + 1;
        else
            B = B + 1;
        end
    end
    if(A > B) W = W + 1; end
    if(A == B) T = T + 1; end
    if(A < B) L = L + 1; end
 %   disp([A,B,W,T,L])
end
toc
disp([W,T,L])
```

```
Elapsed time is 0.062645 seconds.
    86570         0    13430
```

The result is

- A wins 86.57% of the time
- B wins 13.43% of the time