```
% Matlab Assignment #1
% MCET-220: Principle of Statics
% Matheus Laurentys
% Due Date: 09/15/2021
clc;clear all;close all;
```

Problem #1

```
disp("Problem #1")
```

```
Problem #1
```

```
% Given values
x = 3;
y = 4;
% Solves expected expressions
w = 4*(y-5)/3*x-6; % w = 4(y-5)/(3x-6)
z = 3*y/(4*x-8); % z = 3y/(4x-8)
% Display resuts
fprintf('w = %4.2f', w)
```

```
w = -10.00
```

```
fprintf('z = %4.2f', z)
```

```
z = 3.00
```

Problem #2

```
disp("Problem #2")
```

```
Problem #2
```

```
% Creates initial variables
x = 0.1:0.1:2; y = 6:25;
% Computes desired expressions
z = 3*y./(4*x - 8);
w = 4*(y-5)./(3*x-6);
% Computes array lengths
Lx = length(x)
```

```
Lx = 20
```

```
Lw = length(w)
```

```
Lw = 20
```

```
% Prints desired information
fprintf("w = [");
```

```
w = [
```

```
fprintf("%g, ", w);
```

```
-0.701754, -1.48148, -2.35294, -3.33333, -4.44444, -5.71429, -7.17949, -8.88889, -10.9091, -13.3333, -16.2
```

```
fprintf("]");
```

```
]
```

```
fprintf("z = [");
```

```
z = [
```

```
fprintf("%g, ", z);
```

```
-2.36842, -2.91667, -3.52941, -4.21875, -5, -5.89286, -6.92308, -8.125, -9.54545, -11.25, -13.3333, -15.93
```

```
fprintf("]");
```

```
]
```

```
fprintf("Lw = %d", Lw);
```

```
Lw = 20
```

```
fprintf("Lx = %d", Lx);
```

```
Lx = 20
```

Problem #3

```
disp("Problem #3")
```

```
Problem #3
```

```
% Creates initial variables
u = [6,-8,3];
v = [5,3,-4];
w = [-2,-5,7];
% Computes desired expressions
u1 = u(1);
u23 = u(2:3); % Slices elements 2 and 3
u3 = u(end); % Last element of u
a = u + v;
b = -3*(u - 8*v);
c = norm(u) + norm(v); % |u| + |v|
d = norm(-3*(u-8*v)); % -3(u-8v)
eu = u/norm(u); ev = v/norm(v);
f = dot(u,v); % Dot product u . v
g = atan2(norm(cross(u,v)),dot(u,v)); % Angle in radians
h = v * (dot(u,v)/dot(u,u)); % Projection of u onto v
k = cross(u,v); % Cross-product
l = cross(v,u);
m = cross(u,cross(v,w));
% Prints desired information
fprintf("u1 = %d", u1);
```

```
u1 = 6
```

```
fprintf("u23 = %d %d", u23);
```

```
u23 = -8 3
```

```
fprintf("u3 = %d", u3);
```

```
u3 = 3
```

```
fprintf("a = %d %d %d", a);
```

```
a = 11 -5 -1
```

```
fprintf("b = %d %d %d", b);
```

```
b = 102 96 -105
```

```
fprintf("eu = %f %f %f", eu);
```

```
eu = 0.574696 -0.766261 0.287348
```

```
fprintf("ev = %f %f %f", ev);
```

```
ev = 0.707107 0.424264 -0.565685
```

```
fprintf("f = %g", f);
```

```
f = -6
```

```
fprintf("g = %g radians", g);
```

```
g = 1.65216 radians
```

```
fprintf("h = %g %g %g", h);
```

```
h = -0.275229 -0.165138 0.220183
```

```
fprintf("h = %d %d %d", h);
```

```
h = -2.752294e-01 -1.651376e-01 2.201835e-01
```

```
fprintf("k = %d %d %d", k);
```

```
k = 23 39 58
```

```
fprintf("l = %d %d %d", l);
```

```
l = -23 -39 -58
```

```
fprintf("m = %d %d %d", m);
```

```
m = 233 117 -154
```

```

```

Problem #4

```
disp("Problem #4")
```

```
Problem #4
```

```
% Creates initial variables
A = [3,-2,1;6,8,-5;7,9,10];
B = [6,9,-4;7,5,3;-8,2,1];
C = [-7,-8;6,2;3,-4];
I3 = [1,0,0;0,1,0;0,0,1];
% Computes  and Prints desired expressions
a1 = A(:,1)
```

```
a1 = 3×1
     3
     6
     7
```

```
a23 = A(2,3)
```

```
a23 = -5
```

```
a3 = A(:,end)
```

```
a3 = 3×1
     1
    -5
    10
```

```
D = A(2:3, 1:2)
```

```
D = 2×2
     6     8
     7     9
```

```
E = A + B
```

```
E = 3×3
     9     7    -3
    13    13    -2
    -1    11    11
```

```
F = A * B
```

```
F = 3×3
    -4    19   -17
   132    84    -5
    25   128     9
```

```
G = A .* B % Term by term
```

```
G = 3×3
    18   -18    -4
    42    40   -15
   -56    18    10
```

```
G = B*A
```

```
G = 3×3
    44    24   -79
    72    53    12
    -5    41    -8
```

```
K = A * C
```

```
K = 3×2
   -30   -32
    -9   -12
    35   -78
```

```
L = (A + B) * C
```

```
L = 3×2
   -30   -46
   -19   -70
   106   -14
```

```
M = det(A) % Determinant
```

```
M = 563
```

```
N = inv(A) % Inverse Matrix
```

```
N = 3×3
    0.2220    0.0515    0.0036
   -0.1687    0.0409    0.0373
   -0.0036   -0.0728    0.0639
```

```
P = A * inv(A)
```

```
P = 3×3
    1.0000   -0.0000         0
   -0.0000    1.0000    0.0000
   -0.0000         0    1.0000
```

```
Q = I3 * A
```

```
Q = 3×3
     3    -2     1
     6     8    -5
     7     9    10
```

Problem #5

```
disp("Problem #5");
```

```
Problem #5
```

```
% Creates initial variables
A = [4,-2,3,-1,7;-3,-5,1,-6,-4;5,-2,6,-3,-1;-6,4,-5,9,2;9,6,-3,0,5];
b = [-28.5;-29.5;67;-75;-49];
% Computes desired expressions
x = inv(A)*b;
M = [A,b];
M_rref = rref(M); % Computes reduced echelonf form
x_rref = M_rref(:,end);
% Prints desired expressions
fprintf("A = [");
```

```
A = [
```

```
fprintf("[%g %g %g %g %g]\n", A);
```

```
[4 -3 5 -6 9]
[-2 -5 -2 4 6]
[3 1 6 -5 -3]
[-1 -6 -3 9 0]
[7 -4 -1 2 5]
```

```
fprintf("]");
```

```
]
```

```
fprintf("b = [");
```

```
b = [
```

```
fprintf("%g\n", b);
```

```
-28.5
-29.5
67
-75
-49
```

```
fprintf("]");
```

```
]
```

```
fprintf("x = [");
```

```
x = [
```

```
fprintf("%g\n", x);
```

```
-281.886
451.561
284.018
-267.193
126.132
```

```
fprintf("]");
```

```
]
```

```
fprintf("M = [");
```

```
M = [
```

```
fprintf("[%g %g %g %g %g %g]\n", A);
```

```
[4 -3 5 -6 9 -2]
[-5 -2 4 6 3 1]
[6 -5 -3 -1 -6 -3]
[9 0 7 -4 -1 2]
[5
```

```
fprintf("]");
```

```
]
```

```
fprintf("M_rref = [");
```

```
M_rref = [
```

```
fprintf("[%g %g %g %g %g]\n", M_rref);
```

```
[1 0 0 0 0]
[0 1 0 0 0]
[0 0 1 0 0]
[0 0 0 1 0]
[0 0 0 0 1]
[-281.886 451.562 284.018 -267.194 126.132]
```

```
fprintf("]");
```

```
]
```

```
fprintf("x_rref = [");
```

```
x_rref = [
```

```
fprintf("%g\n", x_rref);
```

```
-281.886
451.562
284.018
-267.194
126.132
```

```
fprintf("]");
```

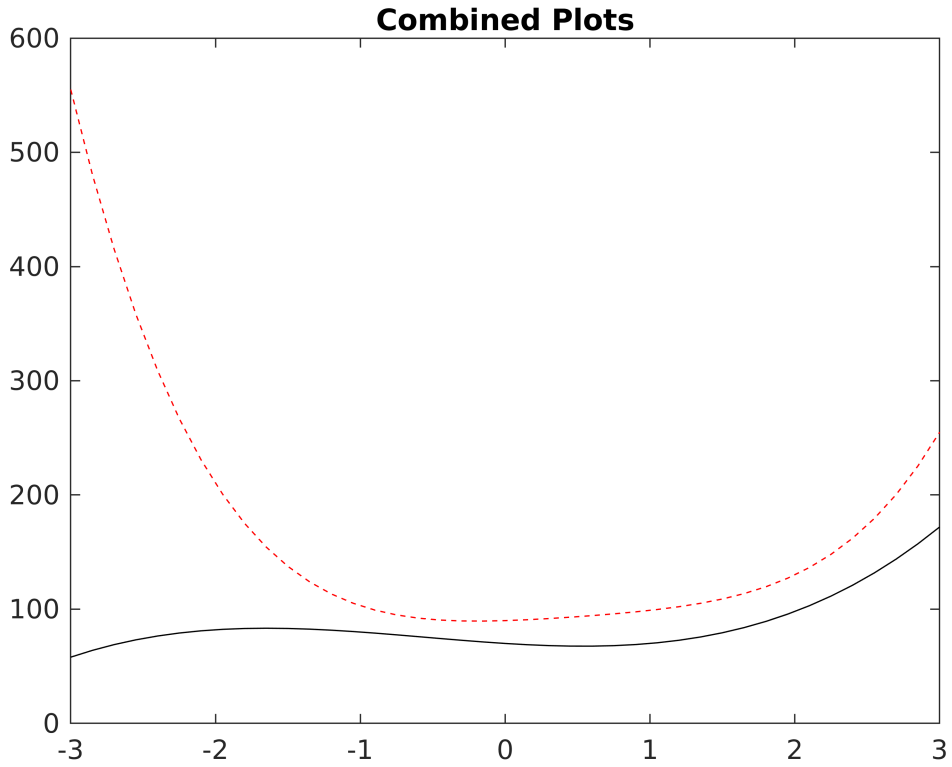
```
]
```

Problem #6

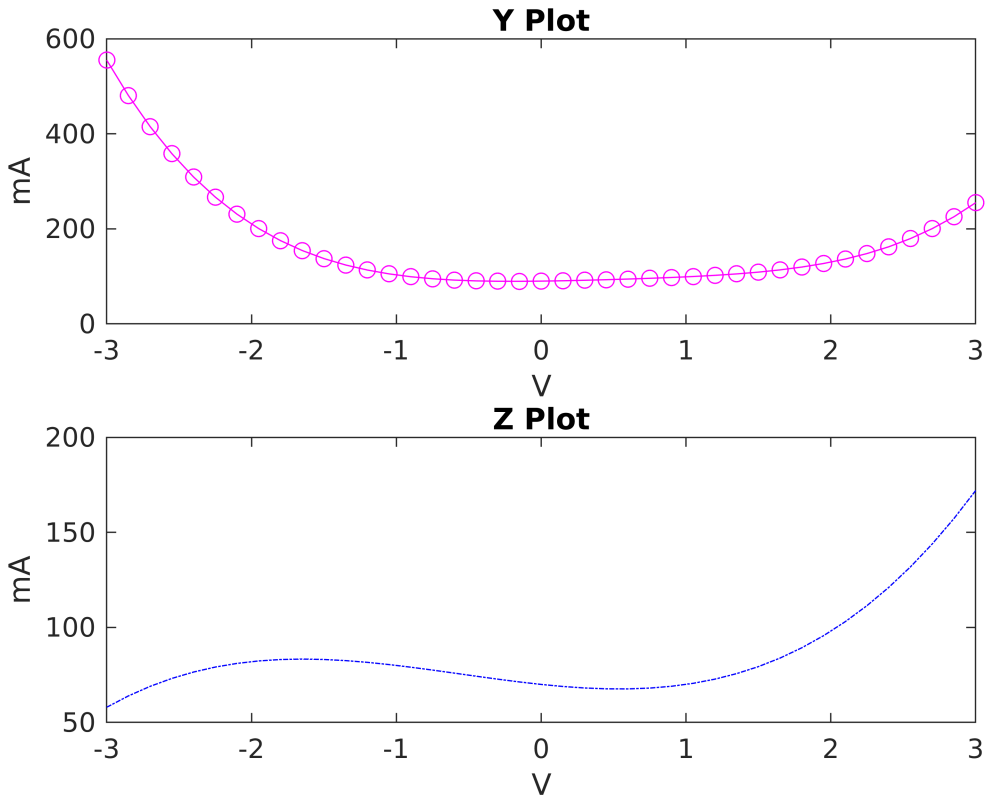
```
disp("Problem #6");
```

```
Problem #6
```

```
% Creates initial variables
x = -3:0.15:3;
y = 3*x.^4 - 6*x.^3 + 8*x.^2 + 4*x + 90;
z = 3*x.^3 + 5*x.^2 - 8*x + 70;
% Computes desired expressions
idx = find(x>=-2&x<=2); % Indices of elements of x satifying condition
ymin2 = min(y(idx(1):idx(end)));
ymax2 = max(y(idx(1):idx(end)));
zmin2 = min(z(idx(1):idx(end)));
zmax2 = max(z(idx(1):idx(end)));
y_idx = find(y <150);
x_y150 = x(y_idx(1));
z_idx = find(z<80);
x_z80 = x(z_idx(end));
% Generates plots
figure();
plot(x,y, "--r");
title('Combined Plots');
hold on;
plot(x,z, "-k");
hold off;
```




```
figure();
subplot(2,1,1);
plot(x,y,"-om");
title('Y Plot');
xlabel("V");
ylabel("mA");
subplot(2,1,2);
plot(x,z,"-.b");
title('Z Plot');
xlabel("V");
ylabel("mA");
```




```
% Prints desired expressions
fprintf("ymin2 = %g", ymin2);
```

```
ymin2 = 89.6018
```

```
fprintf("ymax2 = %g", ymax2);
```

```
ymax2 = 200.486
```

```
fprintf("zmin2 = %g", zmin2);
```

```
zmin2 = 67.648
```

```
fprintf("zmax2 = %g", zmax2);
```

```
zmax2 = 95.6571
```

```
fprintf("x_y150 = %g", x_y150);
```

```
x_y150 = -1.5
```

```
fprintf("z_z80 = %g", x_z80);
```

```
z_z80 = 1.5
```