Table of Contents

Homework 4

ENGR 133-003 Created by Sean DeBarr 2/15/2019

```
clear
clc
```

Problem 3

```
%{
function angle = findAngle(x,y)
% findAngle finds theta in radians if given x and y
    angle = atan2(y,x);
end

function degree = convDegree(angle)
% convDegree takes a radian angle and converts it to degrees
    degree = (angle*180) / pi;
end
%}
clear

disp("**********************" + newline + "Problem 3" + newline);

%
 ************************************************************************
% Part a
disp("Part a" + newline);

% Declare x and y
x = 5;
y = 8;

% Calculate angle in radians then convert to degrees
degreeR = findAngle(x, y);
degreeD = convDegree(degreeR);

% Display results
disp("The angle in radians is: " + degreeR + newline);
disp("The angle in degrees is: " + degreeD + newline);
```

```
%
 ************************************************************************
% Part b
disp("Part b" + newline);

% Declare x and y
x = -5;
y = 8;

% Calculate angle in radians then convert to degrees
degreeR = findAngle(x, y);
degreeD = convDegree(degreeR);

% Display results
disp("The angle in radians is: " + degreeR + newline);
disp("The angle in degrees is: " + degreeD + newline);

%
 ************************************************************************
% Part c
disp("Part c" + newline);

% Declare x and y
x = 5;
y = -8;

% Calculate angle in radians then convert to degrees
degreeR = findAngle(x, y);
degreeD = convDegree(degreeR);

% Display results
disp("The angle in radians is: " + degreeR + newline);
disp("The angle in degrees is: " + degreeD + newline);

%
 ************************************************************************
% Part d
disp("Part d" + newline);

% Declare x and y
x = -5;
y = -8;

% Calculate angle in radians then convert to degrees
degreeR = findAngle(x, y);
degreeD = convDegree(degreeR);

% Display results
disp("The angle in radians is: " + degreeR + newline);
disp("The angle in degrees is: " + degreeD + newline);

**********
Problem 3
```

```
Part a

The angle in radians is: 1.0122

The angle in degrees is: 57.9946

Part b

The angle in radians is: 2.1294

The angle in degrees is: 122.0054

Part c

The angle in radians is: -1.0122

The angle in degrees is: -57.9946

Part d

The angle in radians is: -2.1294

The angle in degrees is: -122.0054
```

Problem 9

```
clear

disp("**********************" + newline + "Problem 9" + newline);

%{
function t = time(h, v0, g)
% time calculates the time required to reach a certain height
    t = roots([0.5*g, -v0, h]);
end
%}

% Declare know values
h = 100;
v0 = 50;
g = 9.81;

% Calculate time to reach height both ascending and decending
timeToH = time(h, v0, g);

% Display results
disp("The time it takes to reach " + h + "m while ascending is: " +
 timeToH(2, 1) + "sec" + newline);
disp("The time it takes to reach " + h + "m while decending is: " +
 timeToH(1, 1) + "sec" + newline);

**********
```

Problem 9

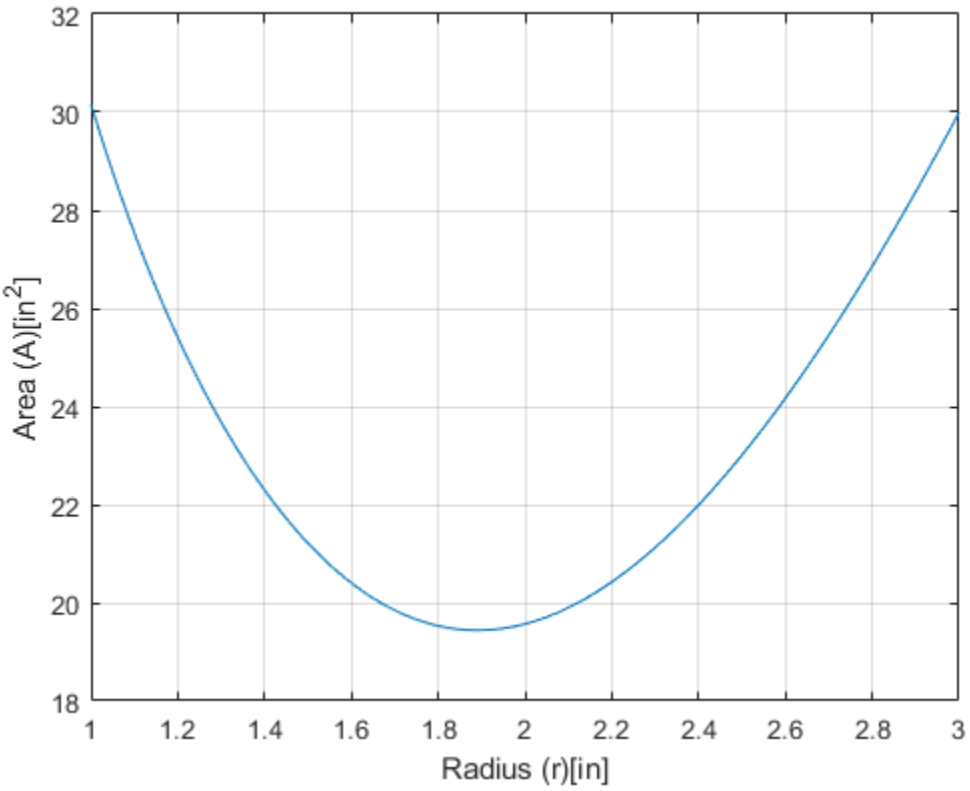
The time it takes to reach 100m while ascending is: 2.7324sec

The time it takes to reach 100m while decending is: 7.4612sec

Problem 16

```
%{
function [A, h] = coneArea(R)
% coneArea calculates the surface area of a conical paper cup given
 the
% radius and volume
    global V;
    h = (3 .* V) ./ (pi .* R.^2);
    A = (pi .* R .* sqrt((R.^2) + h.^2));
end
%}
clear

disp("**********************" + newline + "Problem 16" + newline);

% Declare variables
global V;
V = 10;
R = 1:0.01:3;

% Calculate the minimum surface area and diplay it along with the
 height
minR = fminbnd(@coneArea, 1, 3);
[minArea, minHeight] = coneArea(minR);
disp("The minimum area is: " + minArea + newline);
disp("The value of r that minimizes the surface area is: " + minR +
 newline);
disp("The corresponding height is: " + minHeight + newline);

% Plots the function to determine sensitivity
disp("Refer to external plot" + newline);
plot(R, coneArea(R));
grid on;
xlabel("Radius (r)[in]");
ylabel("Area (A)[in^2]");

% Calculate 10% above area
saVary = minArea * 1.1;

% Diplay the r value within the 10% taken from viewing the plot
disp("The radius can vary between approximately 1.5 - 2.3 to stay
 below " + saVary + "in^2.");
disp(newline);

**********************
```

```
Problem 16

The minimum area is: 19.4393

The value of r that minimizes the surface area is: 1.8901

The corresponding height is: 2.673

Refer to external plot

The radius can vary between approximately 1.5 - 2.3 to stay below
 21.3833in^2.
```



Problem 19

```
clear

disp("**********************" + newline + "Problem 19" + newline);

%{
function balance = savings_balance(a, n, r)
% savings_balance determines your account growth over a period of time
    balance = a * ((1 + r / 100).^n);
end
```

```
%}

% Declare known variables
n = 1:10;
A = 10000;
r = 3.5;

% Calculates the savings balance for every year to n
newBal = savings_balance(A , n, r);

% Display results
disp("   Year  Balance"); %disp([n' cur2str(newBal')])
disp(n' + "  " + cur2str(newBal));

**********
```

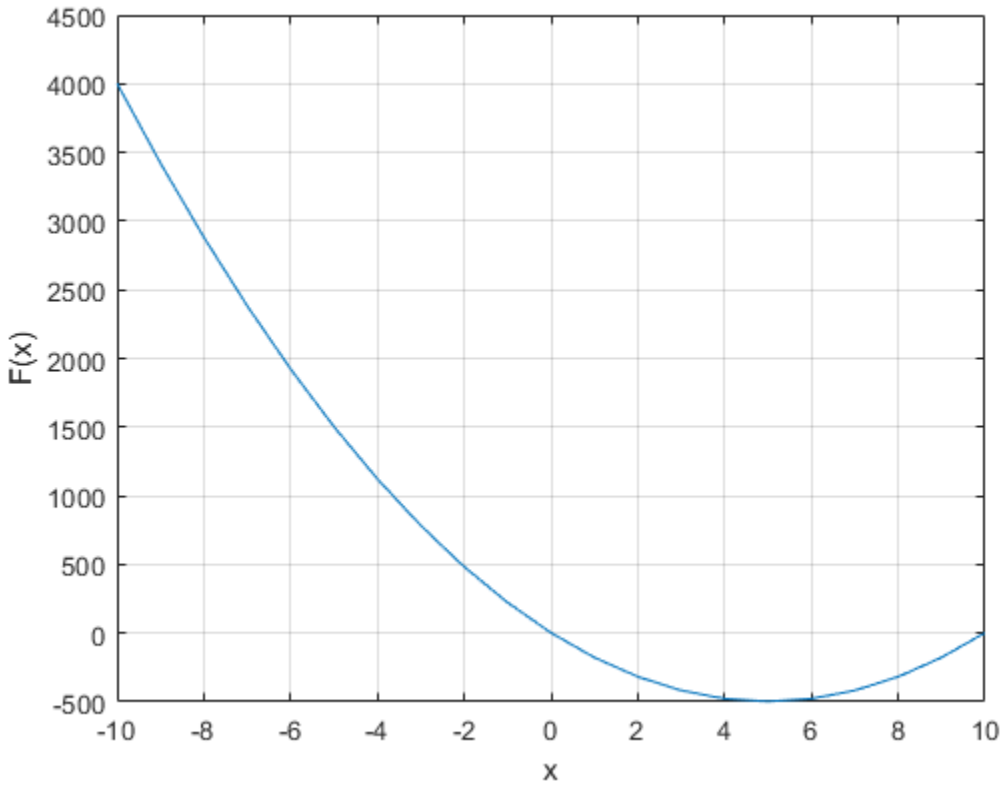
```
Problem 19

   Year  Balance
    "1  $10350.00"
    "2  $10712.25"
    "3  $11087.18"
    "4  $11475.23"
    "5  $11876.86"
    "6  $12292.55"
    "7  $12722.79"
    "8  $13168.09"
    "9  $13628.97"
    "10  $14105.99"
```

Problem 24

```
clear

disp("**********" + newline + "Problem 24" + newline);

% Anoynymous function
givenF = @(x) ((20.*(x.^2)) - (200.*x) + 3);

%
 **********
% Part a
disp("Part a" + newline);

% Plots the function
figure;
plot(-10:1:10, givenF(-10:1:10));
grid on;
xlabel("x");
ylabel("F(x)");

disp("Refer to external plot" + newline);
```

```
%
 ************************************************************************
% Part b
disp("Part b" + newline);

% Calculate the precise minimum using fminbnd
min = fminbnd(givenF, -10, 10);

% Display precise minimum
disp("The minimum value of this function is: " + min + newline);

clear
```

```
**********
Problem 24

Part a

Refer to external plot

Part b

The minimum value of this function is: 5
```



Published with MATLAB® R2017b