```
clear all;
close all;

% NOTE: for parts 1f and 1g, it is easier to see the output of the
%   plotLinks() function by commenting it out in 2 of the 3 loops at a time

theta1_init = 0.0;
theta2_init = 0.0;

xDes = -10;
yDes = -10;

% 1a) Inverse Kinematics Approach
[invKinTheta1, invKinTheta2] = findAnglesInverseKin(xDes, yDes)

% 1b) Optimization Approach (the global variables are for part 1f, where
% several desired positions are passed to the optimization function)
global xDesired;
global yDesired;
xDesired = xDes;
yDesired = yDes;
opt_answer = fminsearch(@findAnglesOptimization,[theta1_init,theta2_init])

% 1c) Brute Force Approch
[bForceTheta1, bForceTheta2] = findAnglesBruteForce(xDes,yDes)


% 1d) Circle of Points
xVals = [];
yVals = [];
radius = 15;
for i = 0:pi/8:2*pi
    xVals = [xVals radius*cos(i)];
    yVals = [yVals radius*sin(i)];
end

% 1e) Plotting Links Function
plotLinks(-pi/2, -pi/2);
title("Proof that link plotting function works");

% 1f) Plotting links for each point on the circle using inverse kinematics
tic
for i = 1:length(xVals)
    [angle1, angle2] = findAnglesInverseKin(xVals(i), yVals(i));
    plotLinks(angle1, angle2);
end
toc

% 1g) Plotting links for each point on the circle using optimization
tic
for i = 1:length(xVals)
    xDesired = xVals(i);
    yDesired = yVals(i);
    opt_answer = fminsearch(@findAnglesOptimization,[theta1_init,theta2_init]);
    plotLinks(opt_answer(1), opt_answer(2));
```

```
end
toc

% 1g) Plotting links for each point on the circle using brute force (guess
% and check) approach
%   NOTE: Joint angles limited to [0,90 deg] and [-90,90 deg] for theta1
%   and theta2, respectively.
tic
for i = 1:length(xVals)
    [bForceTheta1, bForceTheta2] = findAnglesBruteForce(xVals(i), yVals(i));
    plotLinks(bForceTheta1, bForceTheta2);
end
toc

% 2) Using fmincon function to solve inverse kinematic equation
fun2 = @inverseEqn2;
theta1_init = 0.6;
z = fminunc(fun2,theta1_init)
z = fmincon(fun2,theta1_init,[1;-1],[pi;pi])
```

```
invKinTheta1 =

   -3.1416


invKinTheta2 =

    1.5708


opt_answer =

   -3.1416    1.5708


bForceTheta1 =

     0


bForceTheta2 =

   -1.5708

Elapsed time is 1.589311 seconds.
Elapsed time is 1.593445 seconds.
Elapsed time is 1.477697 seconds.

Local minimum found.

Optimization completed because the size of the gradient is less than
the value of the optimality tolerance.


z =
```

```
    0.7854

Local minimum found that satisfies the constraints.

Optimization completed because the objective function is non-decreasing in
feasible directions, to within the value of the optimality tolerance,
and constraints are satisfied to within the value of the constraint tolerance.


z =

    0.7854
```

---

*Published with MATLAB® R2019b*