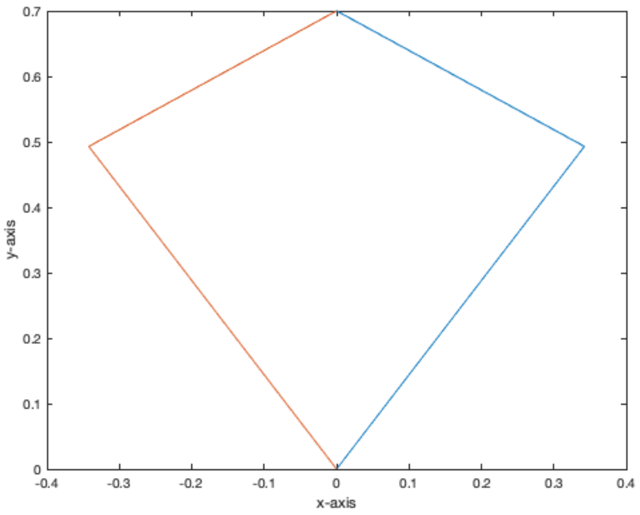
```
close all
clear
clc

% Test for function one
% l = [2;4];
% conf = [pi/4;pi/3];
% pos_end = RR_direct_2D(l,conf);
l = [0.6;0.4];
endPoint = [0;0.7];
result = RR_inverse_2D(l,endPoint);
if (isstring(result) == 1) %check if the returned result is string. if it is string, the error has occured.
    disp(result);  %display the error message
else
    startPoint = [0;0];

    midPointXSoln1 = l(1)*cos(result(1,1));
    midPointYSoln1 = l(1)*sin(result(1,1));
    midPointSoln1 = [midPointXSoln1;midPointYSoln1];

    midPointXSoln2 = l(1)*cos(result(1,2));
    midPointYSoln2 = l(1)*sin(result(1,2));
    midPointSoln2 = [midPointXSoln2;midPointYSoln2];
    plotLinks(startPoint,midPointSoln1,endPoint);
    hold on;
    plotLinks(startPoint,midPointSoln2,endPoint);
end


function pos_end = RR_direct_2D(Link_Lengths,config)   %output position of end-effector (column vector) by inputting angles and link lengths
pos_end(1) = Link_Lengths(1)*cos(config(1))+Link_Lengths(2)*cos(config(1)+config(2));  % x-component
pos_end(2) = Link_Lengths(1)*sin(config(1))+Link_Lengths(2)*sin(config(1)+config(2));  % y-component
end

function config = RR_inverse_2D(Link_Lengths,pos_end)  %output angles (2 by 2 matrix, each column vector for each of the two solns) by inputting p
cosTheta2 = (pos_end(1)^2+pos_end(2)^2-Link_Lengths(1)^2-Link_Lengths(2)^2)/(2*Link_Lengths(1)*Link_Lengths(2)); %first equation for theta2
if (cosTheta2 >= -1)&&(cosTheta2 <= 1)  %check if the point is accessible.
    sinTheta2 = sqrt(1-cosTheta2^2); %second equation for theta2

    config(2,1) = atan2(sinTheta2,cosTheta2);  %calculate theta2 for the first soln.
    config(1,1) = atan2(pos_end(2),pos_end(1)) - atan2(Link_Lengths(2)*sinTheta2,Link_Lengths(1)+Link_Lengths(2)*cosTheta2); %calculate theta1 for
    config(2,2) = atan2(-1*sinTheta2,cosTheta2); %theta2 for second soln.
    config(1,2) = atan2(pos_end(2),pos_end(1)) - atan2(Link_Lengths(2)*-1*sinTheta2,Link_Lengths(1)+Link_Lengths(2)*cosTheta2); %calculate theta1
else
    config = "The point is not accessible"; %retunr the error message.
end
end

function graph = plotLinks(startP,midP,endP)
sumP = [startP,midP,endP];  %merge the three point into one 2 by 3 matrix
rowX = sumP(1,:);
rowY = sumP(2,:);
plot(rowX,rowY);   %plot the graph
xlabel("x-axis");
ylabel("y-axis");
end
```



Published with MATLAB® R2018b