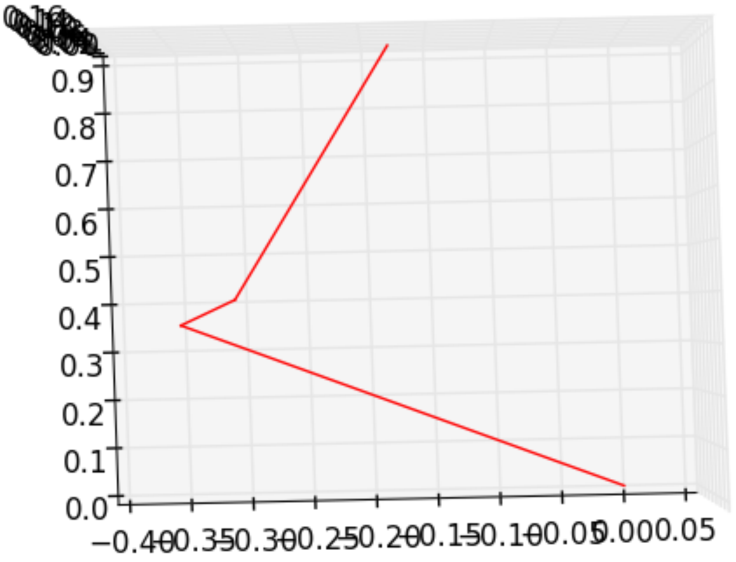
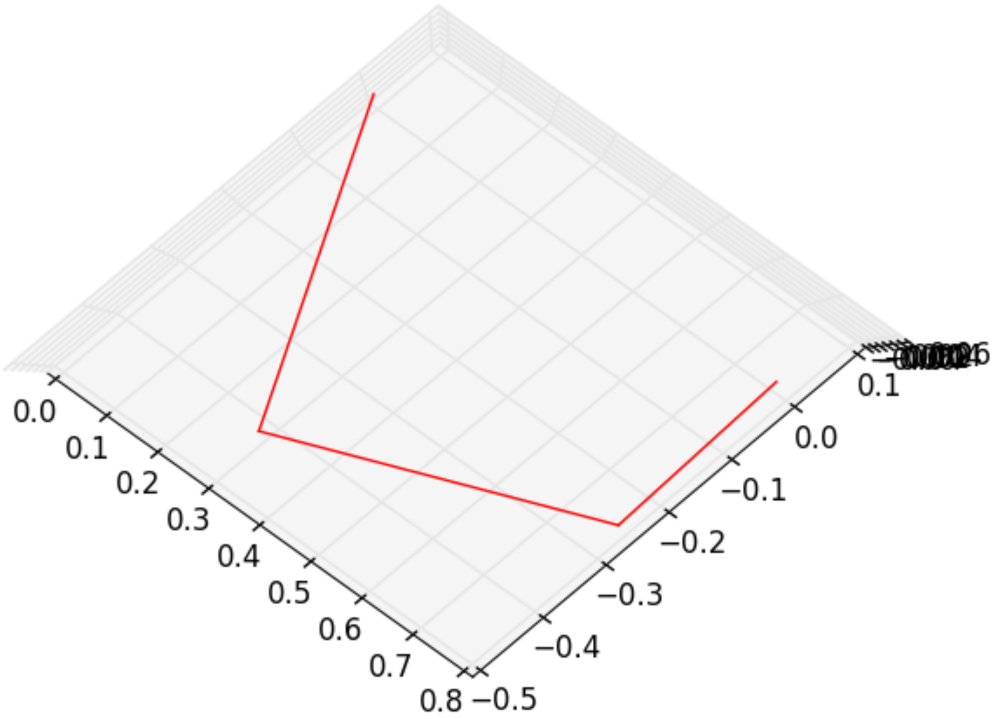
```
#!/usr/bin/env python
import warnings
warnings.filterwarnings('ignore', 'The iteration is not making good progress')
import sys
sys.path.append(r"/Users/robertbrothers/Desktop/Fall 2014/Fundamentals_of_Robotics/r
obo_git/python/")
import robotics_functions as rf
import matplotlib.pyplot as plt
import numpy as np
import scipy as sc
from scipy import optimize
import sympy as sy
from mpl_toolkits.mplot3d import Axes3D

#########################   Problem 1 #######################################
# set up the D-H table
dh_table = np.array( [
  [0,-np.pi/2, 0, np.pi/4],
  [0, np.pi/2,.5, np.pi/4],
  [0, 0, .1, 0],
  [0,-np.pi/2, 0, np.pi/4],
  [0, np.pi/2, 0, np.pi/4],
  [0, 0, .5, np.pi/4]
  ])

A0n = []
A06 = np.identity(4)
for link in dh_table:
  print link
  A06 = A06*rf.denavit_hartenberg(link)
  print A06
  A0n.append(A06)
P = []
for i in A0n:
  P.append(i[:,3][:3])

print A06
X,Y,Z = [np.array([ p[i] for p in P]).T[0][0] for i in range(3)]

# Plot the Graph
fig = plt.figure()
ax = fig.add_subplot(111, projection='3d')
line1 = ax.plot([X[0],X[1]], [Y[0], Y[1]], [Z[0], Z[1]], 'r')
line2 = ax.plot([X[1],X[2]], [Y[1], Y[2]], [Z[1], Z[2]], 'r')
line3 = ax.plot([X[2],X[3]], [Y[2], Y[3]], [Z[2], Z[3]], 'r')
line4 = ax.plot([X[3],X[4]], [Y[3], Y[4]], [Z[3], Z[4]], 'r')
line5 = ax.plot([X[4],X[5]], [Y[4], Y[5]], [Z[4], Z[5]], 'r')
plt.show()

######################    End of Problem1 ####################################

#######################   Problem 2 #########################################
ee_position = np.array([.75,0,0])
def inv_kin(th):
  print th
  dh_table = np.array([
    [0.5, 0, 0, th[0]],
    [0.5, 0, 0, th[1]],
    [0.25,0, 0, th[2]]
    ])
  A03 = np.identity(4)
  for link in dh_table:
    A03 = A03*rf.denavit_hartenberg(link)
  x = A03[:,3][:3][0] - ee_position[0]
  y = A03[:,3][:3][1] - ee_position[1]
  z = A03[:,3][:3][2] - ee_position[2]
  return np.array(np.hstack((x[0],y[0],z[0])))[0]

th = sc.optimize.fsolve(inv_kin, [-.9, 1.2, 1.3], xtol=1e-3 )
dh_table = np.array([
  [0.5, 0, 0, th[0]],
  [0.5, 0, 0, th[1]],
  [0.25,0, 0, th[2]]
  ])
A0n = []
A03 = np.identity(4)
for link in dh_table:
  A03 = A03*rf.denavit_hartenberg(link)
  A0n.append(A03)
# plot the graph

P = []
for i in A0n:
  P.append(i[:,3][:3])

print A03
X,Y,Z = [np.array([ p[i] for p in P]).T[0][0] for i in range(3)]
fig = plt.figure()
ax = fig.add_subplot(111, projection='3d')
linei = ax.plot([   0,X[0]], [   0, Y[0]], [   0, Z[0]], 'r')
line1 = ax.plot([X[0],X[1]], [Y[0], Y[1]], [Z[0], Z[1]], 'r')
line2 = ax.plot([X[1],X[2]], [Y[1], Y[2]], [Z[1], Z[2]], 'r')
plt.show()
```


0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0.0
−0.40
−0.35
−0.30
−0.25
−0.20
−0.15
−0.10
−0.05
0.00
0.05


0.0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
−0.5
−0.4
−0.3
−0.2
−0.1
0.0
0.1