```
import numpy as np
from mpl_toolkits.mplot3d import Axes3D
import matplotlib.pyplot as plt
import serial
import time
from lidar_classdef import *

ser=serial.Serial('/dev/ttyACM0', 9600) #Defines the serial port to use

data=[] #List that stores the data from the arduino

horzBegin=115 #Sets the home horizontal angle
horzEnd=60 #Sets the end horizontal angle

vertBegin=64 #sets the home vertical angle
vertEnd=103 #Sets the end vertical angle

print "Waiting for arduino to be ready....." #Indicates that the programme is
waiting for the arduino to initialize

time.sleep(1) #Waits for the arduino to initialize

print "Programme beginning now" #Indicates that the python programme is running

#Moves the servo to the starting position

ser.write('5') #Sets the servo over to its home position on the left
time.sleep(0.5) #Waits half a second to ensure the arduino is ready for next command
ser.write('6') #Sets the servo to it's starting vertical angle

horzAngle=range(horzBegin, horzEnd, -1) #creates the horizontal range of angles that
the servo should sweep through
vertAngle=range(vertBegin,vertEnd,1) #Creates the vertical range of angles that the
servo should sweep through

for eachVerticalAngle in vertAngle:


        for eachHorizontalAngle in horzAngle:

                ser.write('7') #Tells the arduino to send back a distance reading

                distance_response=ser.readline() #Receives the distance reading from
the arduino
                cleanReading=distance_response[0:-2] #Removes the last 2 characters
("\r\n") from the arduino output
                distance=float(cleanReading) #Type-casts it as an float so that it
can be plotted

                ser.write('8') #Tells the arduino to send back the current vertical
and horizontal angles

                angle_response=ser.readline() #Receives the vertical and horizontal
angles from the arduino as a string

                cleanReading=angle_response[0:-2] #Removes the last 2 characters
("\r\n") from the arduino output
                seperator_index=cleanReading.index(',') #Figures out at what index
the comma appears (which is how we delineate vertical and horizontal angle)
                vertical_angle=cleanReading[0:seperator_index] #Retreives the
vertical angle from the string that is received from the arduino
                vertical_angle=int(vertical_angle) #Converts the angle from string
to integer
```

```
                horizontal_angle=cleanReading[(seperator_index+1): ] #Retreives the
horizontal angle from the string received from the arduino
                horizontal_angle=int(horizontal_angle) #Converts the angle from a
string to a integer

                processed_full_data_holder=lidar(horizontal_angle, vertical_angle,
distance) #Passes the distance reading and angles into a lidar object (defined
seperately)

                data.append(processed_full_data_holder) #Appends the new lidar
object to the data list

                ser.write('1') #Moves the servo one degree to the right after taking
a reading

        ser.write('3') #At the end of the horizontal sweep, increase the vertical
angle by 1 degree
        time.sleep(0.5) #Waits to allow the arduino to execute the command and be
ready for another

        ser.write('5') #Resets the arduino back to the left to perform a full
horizontal sweep
        time.sleep(0.5) #Waits for the arduino to execute the command and be ready
for another

#def randrange(n, vmin, vmax):
#    return (vmax-vmin)*np.random.rand(n) + vmin

fig = plt.figure() #Creates a new pyplot figure
ax = fig.add_subplot(111, projection='3d') #adds a 3d plot axis to it

xs = [] #Creates a new list for x axis values
ys = [] #Creates a new list for y axis values
zs = [] #Creates a new list for z axis values

for eachPoint in data:
        xs.append(eachPoint.x_pos) #Adds each x axis value to the x axis data list
        ys.append(eachPoint.y_pos) #Adds each y axis value to the y axis data list
        zs.append(eachPoint.z_pos) #Adds each z axis value to the z axis data list

ax.scatter(xs,ys,zs, c ='r', marker = 'o') #Creates a scatter plot of all data

ax.set_xlabel('X Distance in centimeters') #X axis label
ax.set_ylabel('Y Distance in centimeters') #Y axis label
ax.set_zlabel('Z Distance in centimeters') #Z axis label

plt.show() #show the pyplot figure
```