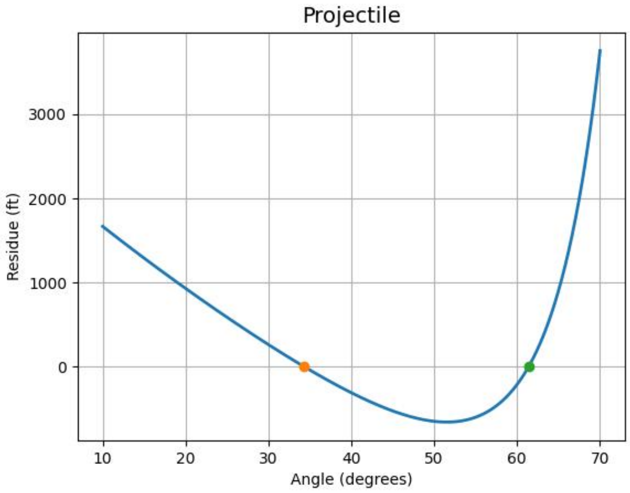
PYTHON CODE

```
"""
Charles Wszalek Assignment 2

Objective: Practice the Bisection OR Secant methods to compute the firing angle for an 81mm
L16 Mortar to hit a target at a specified coordinate.

Instructions:
X Q1) Compute the maximum range and report it.
X Q2) Plot the residue versus the firing angle, theta, for angles ranging from 10 to 70 degrees. Include
    the plot in your report. (Note: Residues are y-axis and the angles are x-axis for the plot)
X Q3) Find a firing angle between 10 to 40 degrees that hits the target. (Report the angle.)
X Q4) Find a firing angle between 40 to 70 degrees that hits the target. (Report the angle.)
"""

from lib.Header import *

v0 = 450 # ft/s : Muzzle Speed
g= 32.2 # ft/s^2 : Gravity Constant
y0 = 0 # starting height
[x, y] = [5000, 500] # ft : Target Coordinate


def Projectile(theta):
    return y + .5 * g * ( x / (v0 * np.cos(theta * np.pi/180)) )**2 - x * np.tan(theta * np.pi/180)


def fn_bisection(xmin, xmax, mypoly, counter): # BISECTION METHOD
    tol = 1e-10
    counter += 1
    xa = (xmin + xmax)/2
    ya = mypoly(xa)

    if (abs(ya) < tol):
        xf = xa
        yf = ya
        return [xf, yf, counter]

    if (ya * mypoly(xmin) < 0):
        xmax = xa
    else:
        xmin = xa

    [xf, yf, counter] = fn_bisection(xmin, xmax, mypoly, counter)

    return [xf, yf, counter]

def fn_secant(x0, x1, mypoly, counter):  # UNUSED ONLY FOR TEST PURPOSES
    tol = 1e-10
    counter += 1
    y0 = mypoly(x0)
    y1 = mypoly(x1)

    x2 = x1 - (y1 * (x1 - x0)) / (y1 - y0)
    y2 = mypoly(x2)

    if abs(y2) < tol:
        xf = x2
        yf = y2
        return [xf, yf, counter]

    [xf, yf, counter] = fn_secant(x1, x2, mypoly, counter)

    return [xf, yf, counter]

maxdist = Projectile(45 * np.pi /180)
print2("Q1) The Mortar's maximum range is ", round(maxdist, 3), "ft. Which is ", round(maxdist * 0.3048, 3),
        "This is reflected by the predicted range.")

anglemin = 10
anglemid = 40
anglemax = 70

N = 1000
counter = 0
angles1 = np.linspace(anglemin, anglemax, N)

[xf0, yf0, counter0] = fn_bisection(anglemin, anglemid, Projectile, counter)
print2("Q3) That angle btween 10 and 40 degrees that hits the target is:", round(xf0,3), "degrees (orange po
[xf1, yf1, counter1] = fn_bisection(anglemid, anglemax, Projectile, counter)
print2("Q4) That angle btween 40 and 70 degrees that hits the target is:", round(xf1,3),"degrees (green poi

plt.figure()
plt.grid()
plt.xlabel("Angle (degrees)")
plt.ylabel("Residue (ft)")
plt.title('Projectile', fontsize=14)
plt.plot(angles1, Projectile(angles1), linewidth=2)
plt.plot(xf0,yf0,'o', linewidth=20)
plt.plot(xf1,yf1,'o', linewidth=20)
SAVE(1)
plt.show()

PDF("Assignment2.py", "Assignment2.pdf")
```

OUTPUT

Plots



Prints

```
Q1) The Mortar's maximum range is  2419.485 ft. Which is  737.459 meters.
This is reflected by the predicted range.
Q3) That angle btween 10 and 40 degrees that hits the target is: 34.331 degrees (orange point).
Q4) That angle btween 40 and 70 degrees that hits the target is: 61.38 degrees (green point).
```