4)

Code for graphs

Points calculation for AIMD, MIAD, MIMD and AIAD

```
import matplotlib.pyplot as plt
import numpy as np

def AIMD(x, y):
    xy_points = [[x, y]]
    for i in range(0, 9):
        pt = xy_points[-1]
        if x + y < 50:
            x+=6
            y+=6
        else:
            x/=2
            y/=2
        xy_points.append([x, y])
    return xy_points


def MIAD(x, y):
    xy_xy_points = [[x, y]]
    for i in range(0, 9):
        pt = xy_points[-1]
        if x + y < 50:
            x*=1.5
            y*=1.5
        else:
            x-=6
            y-=6
        xy_points.append([x, y])
    return xy_points

```

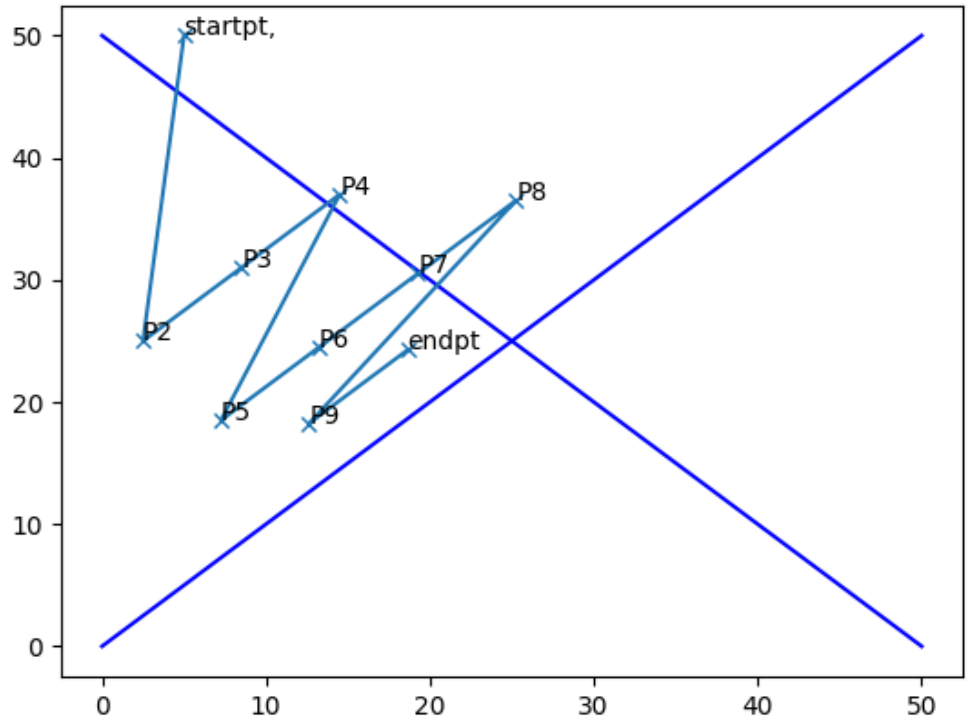
```
def MIMD(x, y):
    xy_points = [[x, y]]
    for i in range(0, 9):
        pt = xy_points[-1]
        if x + y < 50:
            x*=1.2
            y*=1.2
        else:
            x/=1.9
            y/=1.9
        xy_points.append([x, y])
    return xy_points

def AIAD(x, y):
    xy_points = [[x, y]]
    for i in range(0, 9):
        pt = xy_points[-1]
        if x + y < 50:
            x+=10
            y+=10
        else:
            x-=6
            y-=6
        xy_points.append([x, y])
    return xy_points

```

Graph Plotting code

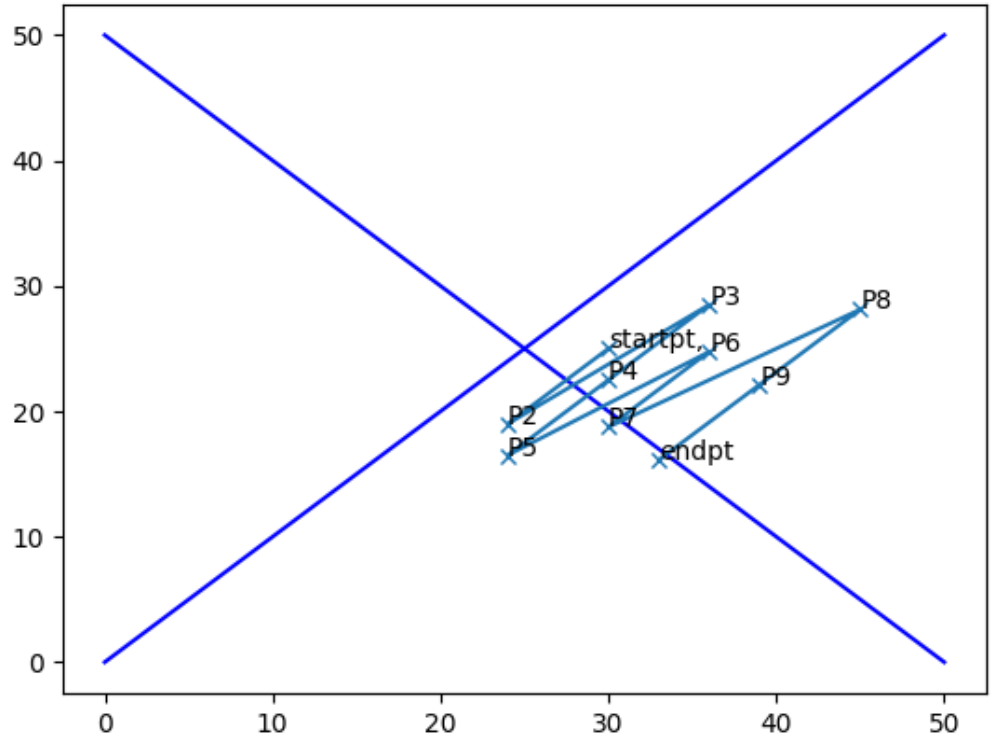
```
#xy_points = MIAD(30, 25)
#xy_points = AIMD(5, 50)
#xy_points = MIMD(20, 45)
xy_points = AIAD(10, 25)

plotinput = np.transpose(xy_points)
annotations = ["startpt,","P2","P3","P4","P5", "P6", "P7", "P8", "P9", "endpt"]

x = [0, 50]
y = [0, 50]
x1 = [0, 50]
y1 = [50, 0]
plt.plot(x, y,  color = 'b')
plt.plot(x1, y1, color = 'b')

plt.plot(plotinput[0], plotinput[1], marker = 'x')
for i, label in enumerate(annotations):
    pt = xy_points[i]
    plt.annotate(label, (pt[0], pt[1]))
plt.waitforbuttonpress
```

Plots in next page

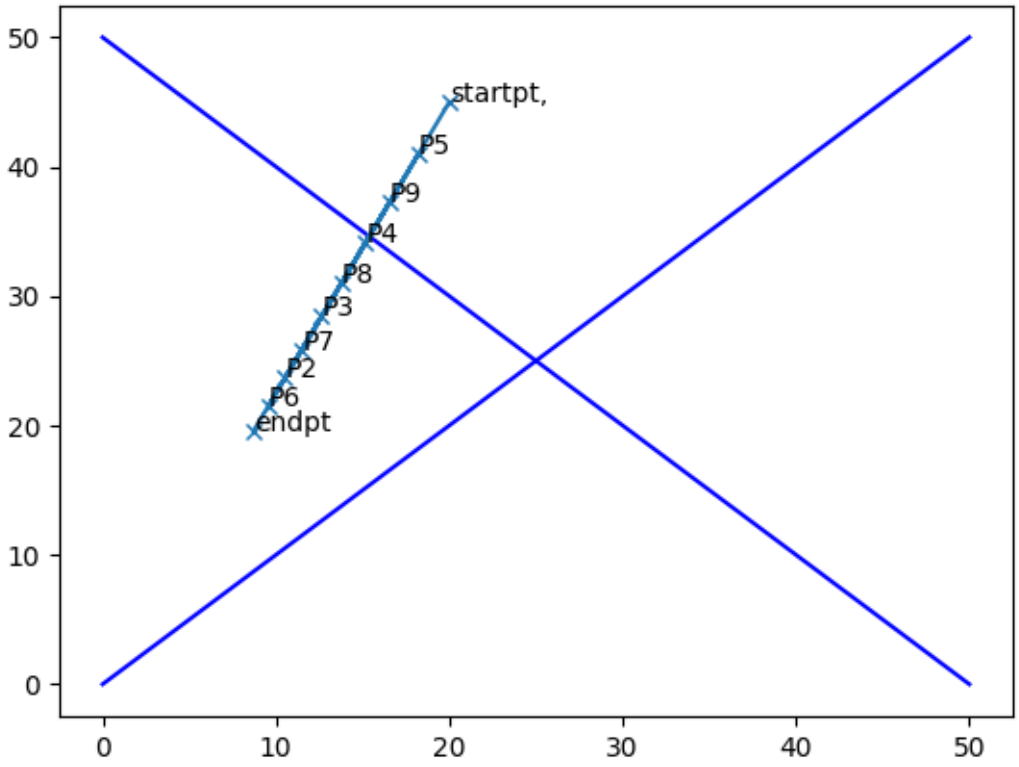
AIMD – Coverging towards optimum solution - FAIR



MIAD – Not Converging towards optimum solution, in-fact it is diverging – NOT FAIR



MIMD – Not Converging towards optimum solution – NOT FAIR



AIAD – Not converging towards optimum solution – NOT FAIR