

**CODE :**

```
import numpy as np
import pandas as pd
import math

x=np.array([[2,4],[4,2],[4,4],[4,6],[6,2],[6,4]])
y=np.array([0,0,1,0,1,0])
# 0=negative 1=positive class
def euclidian_distance(x1,y1,x2,y2):
    return math.sqrt((x1-x2)**2+(y1-y2)**2)
def chooseK(arr):
    print("Size of array :",arr.shape[0])
    k=round(math.sqrt(arr.shape[0]))
    if(k%2==0):
        k=k+1;

    #k should be odd so that classification can be done
    properly(No chance of 50%-50% classification)
    print("Choosen value of K : ",k)
    return k;
chooseK(x)

def classifyPoint(x,y,point,k):
    inputSize=x.shape[0];
    distance=[]; #for string euclidian distance
    for i in range(inputSize):
        distance.append(euclidian_distance(point[0],point[1],x[i][0],x[i][1]));
    mergedList=list(zip(distance,y));
    mergedList.sort(); #sort according to increasing distance
    freq0=0; #Freq of group 0 (negative)
    freq1=0; #Freq of group 1 (positive)
    for i in range(int(k)): #Iterate for k neighbours
        if(mergedList[i][1]==0):
            freq0=freq0+1;
        elif (mergedList[i][1]==1):
            freq1=freq1+1;
    if(freq0>freq1):
        return 0;
    else:
```

```

return 1;

def main():
    print("Input X coordinate");
    x_co=int(input())
    print("Enter Y coordinate")
    y_co=int(input())
    pointt=(x_co,y_co)
    print(pointt)
    k=chooseK(x);
    label="--"
    if(classifyPoint(x=x,y=y,point=pointt,k=k)==0):
        label="Negative";
    else:
        label="Positive";
    print("Point {} belongs to {} class".format(pointt,label))
    print (classifyPoint(x=x,y=y,point=pointt,k=k))
main()

```

Output :

```

('Size of array :', 6)
('Chosen value of K : ', 3.0)

```

Input X coordinate

6

Enter Y coordinate

6

(6, 6)

('Size of array :', 6)

('Chosen value of K : ', 3.0)

Point (6, 6) belongs to Negative class

0