**SSN College of Engineering**

**Department of Information Technology**

**UIT2201 — Programming and Data Structures**

**2022 – 2023**

**Exercise — 02**

**Part A**

**1)Define a class Point, a simple class to represent 2-dimensional points (Non-mutable). Each object has two fields: '\_x' and '\_y'. Methods include 'distance' that returns Euclidean distance between 'this' object and another object.**

**Aim:**

**To create a simple class to represent 2-dimensional points (Non-mutable) and return the Euclidean**

**Distance between the two points.**

**Program:**

import random

class Point:

    '''

    This class represents 2-Dimensional point and gives the Euclidean distance

    between the two points

    Orginal Author: Paranthagan S

    Created On: 20/04/23

    '''

    def \_\_init\_\_(self,a,b):     #constructor of point class

        self.x=a

        self.y=b

    def \_\_str\_\_(self):          #display the value for object

        return "(" + str(self.x) + "," + str(self.y) + ")"

    def distance(self,other):   #find Euclidean distance between two points by using formula

        xdis=(self.x-other.x)\*\*2

        ydis=(self.y-other.y)\*\*2

        dis=(xdis+ydis)\*\*0.5

        return dis

#Driver Code

if \_\_name\_\_ == "\_\_main\_\_":

    x1=float(input("Enter Coordinate of X1: "))

    y1=float(input("Enter Coordinate of Y1: "))

    x2=float(input("Enter Coordinate of X2: "))

    y2=float(input("Enter Coordinate of Y2: "))

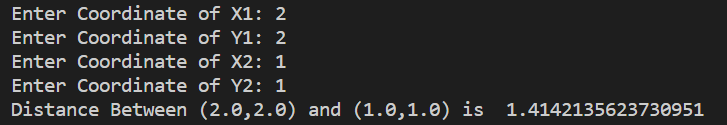
    p1=Point(x1,y1)

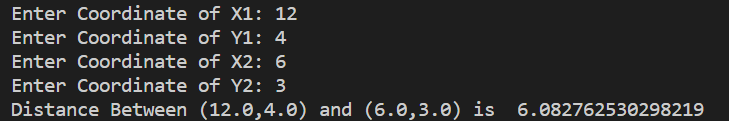
    p2=Point(x2,y2)

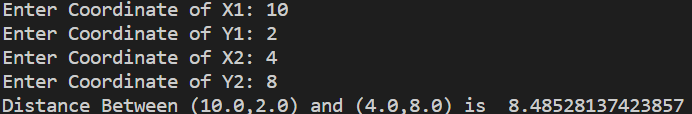
    print(f"Distance Between {p1} and {p2} is ",p1.distance(p2))

**Output:**

**Test Cases:**







**2)Write a Python code to generate a random sequence of n Points. Define a function that, given an integer k and a new Point Pnew, returns k-nearest neighbours of Pnew in the given sequence of n Points.**

**Aim:**

**To generate a random sequence of n Points. And create a function that given an integer k and a new** **Point Pnew, returns k-nearest neighbours of Pnew in the given sequence of n Points.**

**Program:**

'''

    This module generates the random sequence of N points, takes integer "k",

    takes New point and returns K nearest neighbours of that point

    Orginal Author: Paranthagan S

    Created On: 20/04/23

'''

from point1 import Point                #importing required module

import random

x1=float(input("Enter point x1: "))

y1=float(input("Enter point x2: "))

Pnew=Point(x1,y1)

print("New Point: ",Pnew)

def generate\_point(l):                  #this function is to generate Number of random points

    n=int(input("Enter Number Points: "))

    for i in range(n):

        x2=random.randint(1,100)

        y2=random.randint(1,100)

        point=Point(x2,y2)

        l.append((str(point),Pnew.distance(point)))

k=int(input("Enter Number of Nearest Neighbours: "))    #getting the integer 'k'

l=[]

generate\_point(l)                       #calling the function for generating points

l.sort(key = lambda x : x[1])

print ("Number of Point: ",l)           #returns list of points in ascending order

print (f"Number of {k} Nearest Neighbours: ",l[0:k])    #return 'k' nearest neighbours from new point

**Output:**

**Test Cases:**

