DATA STRUCTURES - PROJECT

(Arrays, Matrices, Methods, Classes, Random Numbers)

GENERATING DISTANCE MATRIX FROM POINTS IN A 2D PLANE
AND
TRAVELING POINTS USING NEAREST NEIGHBOR ALGORITHM

1) GENERATING DISTANCE MATRIX FROM POINTS IN A 2D PLANE

In the first part of the project, you are asked to generate points in 2-dimensional (2D) Euclidean space and perform some calculations on these points. Figure 1 shows an example 2D Euclidean plane with the origin (0, 0) as the upper left corner.

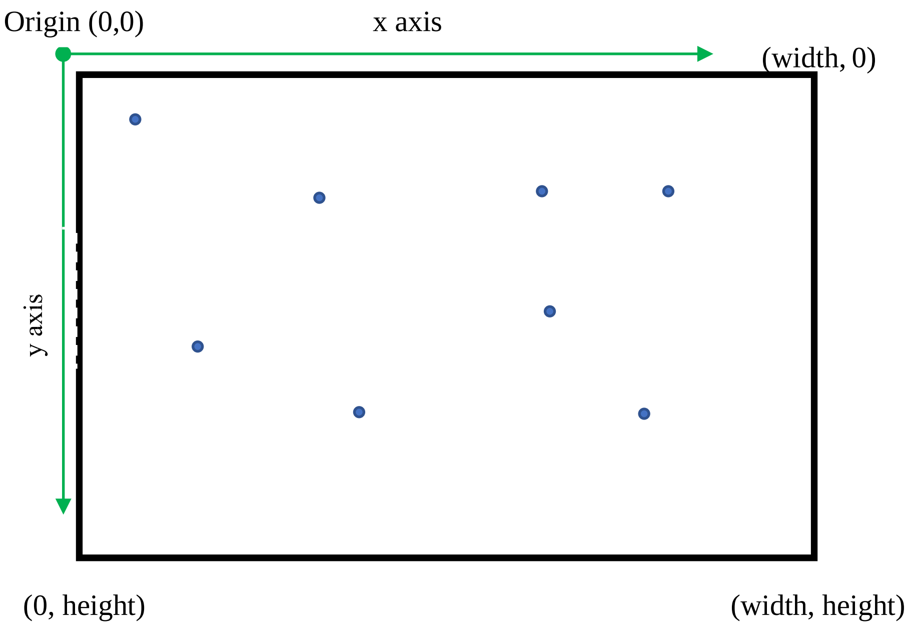

Figure 1: Representation of 2D points in the Euclidean plane

The Euclidean distance between two points in two-dimensional space is calculated according to the formula below:

$$d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

For example, the distance between points P1 (x_1 =10.2, y_1 =20.7) and P2 (x_2 =3.5, y_2 =19.9)

$$d = \sqrt{(3.5 - 10.2)^2 + (19.9 - 20.7)^2} \cong 6.75$$

a) Random Point Generation: Write a method that generates and returns n random points in a 2-dimensional space of given **width** and **height**. The generated points will be stored and returned in **a matrix of n x 2** such that each row corresponds to a point and each column corresponds to x and y coordinate values respectively. The coordinates to be generated must be of type **double**.

Test this method separately with the following parameters: Print the information of the matrix returned as a result of the test (x and y coordinate values for each point) on the console.

1. **n=20, width=100, height=100**
2. **n=50, width=100, height=100**

b) Distance Matrix (DM) Generation: Write a method that converts and returns the given nx2 matrix of points (generated using the method described in the previous item) into an nxn distance matrix. The distance matrix (DM) contains the distance information between each pair of points. For example, DM[i,j] will give the distance between points i and j. Since the distances are symmetric, the equality DM[i,j]=DM[j,i] will hold (the distance from i to j is the same as the distance from j to i). Table 1 shows an example distance matrix.
Test this method with parameters n=20, width=100, height=100. Print the generated DM to the console.

Table 1: A sample Distance Matrix (DM) for 6 points

Distance Matrix (DM)						
	0	1	2	3	4	5
0	0	1,2	0,5	4,7	5,6	4,9
1	1,2	0	3,1	2	1,4	4
2	0,5	3,1	0	6,1	2,8	1,9
3	4,7	2	6,1	0	2,1	3,5
4	5,6	1,4	2,8	2,1	0	3,3
5	4,9	4	1,9	3,5	3,3	0

c) For n = 20, write the method that starts from a random point and traverses all points according to the nearest neighbor method (The nearest neighbor method continues by finding the closest point to the starting point according to the Euclidean distance. Then, it goes to the point closest to this new point among the points that have not been traversed. This process continues until all points are visited and the tour is completed). Calculate the total path length. Repeat this for 10 different random starting points and list the following information on the console for 10 different tours:

- **DM (distance matrix),**

For each tour:

- **Tour number (from 1 to 10),**
- **Which numbered points it visited in sequence on the relevant tour (such as 5-2-0-1-3-4), The total path length of the tour (sum of distances if 5-2-0-13-4 tours were made).**

Describe the data structures and algorithms you used and explain how you made them find points that had not been explored so far.