

139) Finding closest pair points in 2D space

CODE:

```
import math

def calculate_distance(point1, point2):
    return math.sqrt((point1[0] - point2[0])**2 + (point1[1] - point2[1])**2)

def find_closest_pair(points):
    min_distance = float('inf')
    closest_pair = None

    for i in range(len(points)):
        for j in range(i + 1, len(points)):
            distance = calculate_distance(points[i], points[j])
            if distance < min_distance:
                min_distance = distance
                closest_pair = (points[i], points[j])

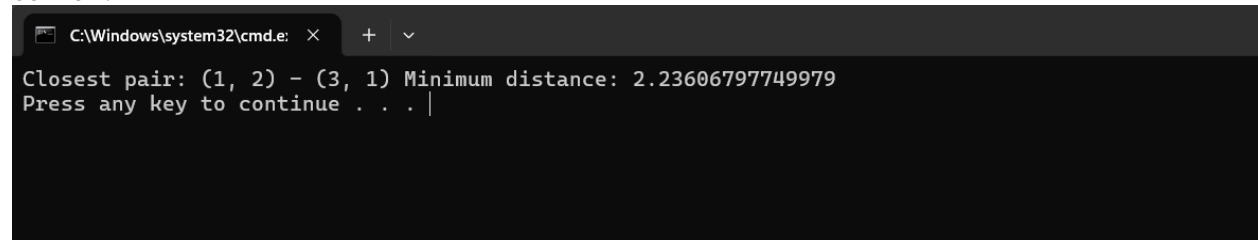
    return closest_pair, min_distance

points = [(1, 2), (4, 5), (7, 8), (3, 1)]

closest_pair, min_distance = find_closest_pair(points)

print(f"Closest pair: {closest_pair[0]} - {closest_pair[1]} Minimum distance: {min_distance}")
```

OUTPUT:

A screenshot of a Windows command prompt window. The title bar shows the path 'C:\Windows\system32\cmd.e' and standard window controls. The command prompt displays the output of the Python code: 'Closest pair: (1, 2) - (3, 1) Minimum distance: 2.23606797749979' followed by 'Press any key to continue . . . |'.

TIME COMPLEXITY : $O(n^2)$