

150. Write a program that finds the closest pair of points in a set of 2D points using the brute force approach.

Program:-

```
import math
```

```
def euclidean_distance(point1, point2):
```

```
    x1, y1 = point1
```

```
    x2, y2 = point2
```

```
    return math.sqrt((x2 - x1) ** 2 + (y2 - y1) ** 2)
```

```
def closest_pair_brute_force(points):
```

```
    n = len(points)
```

```
    if n < 2:
```

```
        return None, float('inf')
```

```
    min_distance = float('inf')
```

```
    closest_pair = None
```

```
    for i in range(n):
```

```
        for j in range(i + 1, n):
```

```
            dist = euclidean_distance(points[i], points[j])
```

```
            if dist < min_distance:
```

```
                min_distance = dist
```

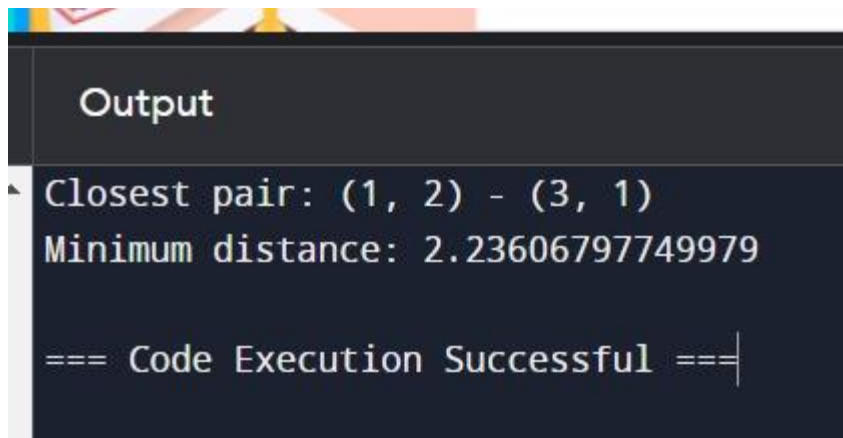
```
                closest_pair = (points[i], points[j])
```

```
    return closest_pair, min_distance
```

input:-

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

ouput:-

A screenshot of a code execution environment's output window. The window has a dark background with a light-colored header bar that says "Output". Below the header, the text "Closest pair: (1, 2) - (3, 1)" is displayed on the first line, and "Minimum distance: 2.23606797749979" is on the second line. At the bottom of the window, the text "=== Code Execution Successful ===" is shown, followed by a vertical cursor line.

```
Output
Closest pair: (1, 2) - (3, 1)
Minimum distance: 2.23606797749979

=== Code Execution Successful ===
```

Time complexity:- $O(n^2)$