151. Write a program to find the closest pair of points in a given set using the brute force approach. Analyze the time complexity of your implementation. Define a function to calculate the Euclidean distance between two points. Implement a function to find the closest pair of points using the brute force method. Test your program with a sample set of points and verify the correctness of your results. Analyze the time complexity of your implementation. Write a brute-force algorithm to solve the convex hull problem for the following set S of points? P1 (10,0)P2 (11,5)P3 (5, 3)P4 (9, 3.5)P5 (15, 3)P6 (12.5, 7)P7 (6, 6.5)P8 (7.5, 4.5).How do you modify your brute force algorithm to handle multiple points that are lying on the sameline?

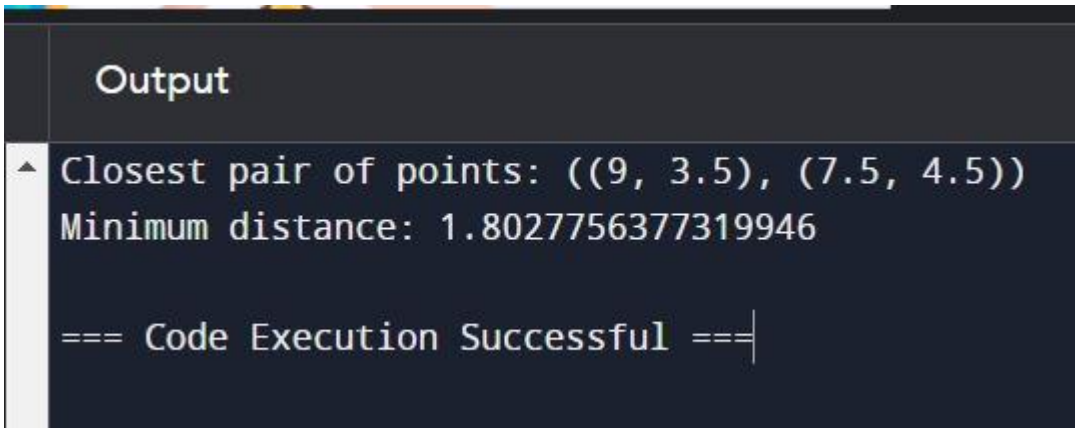
Program:-

```
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 = [(10, 0), (11, 5), (5, 3), (9, 3.5),
    (15, 3), (12.5, 7), (6, 6.5), (7.5, 4.5)]
```

Output:



Time complexity:-O(n^3)