2.11 FINDS THE CONVEX HULL OF A SET OF 2D POINTS

Question:

Write a program that finds the convex hull of a set of 2D points using the brute force approach. Input: A list or array of points represented by coordinates (x, y). Points: [(1, 1), (4, 6), (8, 1), (0, 0), (3, 3)] Output: The list of points that form the convex hull in counter-clockwise order. Convex Hull: [(0, 0), (1, 1), (8, 1), (4, 6)].

AIM

To determine the set of points that form the convex hull of a given set of 2D points using the brute force method and return them in counter-clockwise order

ALGORITHM

1. Read the list of 2D points.
2. For each pair of points (p1, p2) in the set:
3. Assume they form an edge of the convex hull.
4. Check all other points and Compute the cross product to determine if all points lie on one side of the line segment (p1, p2).
5. If all points lie on the same side or are collinear, mark (p1, p2) as part of the hull.
6. Collect all points that are part of at least one hull edge.
7. Sort the hull points in counter-clockwise order using the centroid angle method.
8. Output the hull points

PROGRAM

```
def orientation(p, q, r):
    val = (q[1] - p[1]) * (r[0] - q[0]) - \
          (q[0] - p[0]) * (r[1] - q[1])
    return 0 if val == 0 else (1 if val > 0 else 2)

def convex_hull(points):
    n = len(points)
    if n < 3:
        return []

    hull = []
    for i in range(n):
        for j in range(i+1, n):
            left = right = False
            for k in range(n):
                if k == i or k == j:
                    continue
                o = orientation(points[i], points[j], points[k])
                if o == 1:
                    left = True
                elif o == 2:
                    right = True
            if not (left and right):
                if points[i] not in hull:
                    hull.append(points[i])
                if points[j] not in hull:
                    hull.append(points[j])
    return sorted(hull)

def run_convex_hull():
    raw = input("Enter points as x,y separated by space: ").split()
    points = [tuple(map(int, p.split(','))) for p in raw]
    hull = convex_hull(points)
    print("Convex Hull:", hull)

run_convex_hull()
```

Input:

[(1, 1), (4, 6), (8, 1), (0, 0), (3, 3)]

Output:

```
Enter points as x,y separated by space: 1,1 4,6 8,1 0,0 3,3
Convex Hull: [(0, 0), (4, 6), (8, 1)]
>>>
```

RESULT:

Thus finding convex hull of a set of 2d points using the brute force approach is successfully executed and the output is verified.

PERFORMANCE ANALYSIS:

· Time Complexity: $O(n^3)$

· Space Complexity: O(n)