Rank finding algorithm

Name: 谭演锋 Student ID:202130100456

April 22, 2023

1 Question

Implement the rank finding algorithm based upon the divide-and- conquer approach. Compare it with the straightforward approach.

2 Pseudocode

2.1 Divide and conquer

Algorithm 1 Divide and conquer

```
Input A set S of planar points $P_1, P_2, ..., P_n$.
Output: The rank of every point in S.
Step1:
if $S$ contains only one point then
    Return its rank as 0.
else
    Use the quickSelect algorithm find the median $x$ value of the set. Make a line that perpendicular to the x-axis with $L : x =$median x of the points. Such that n/2 points of Shave X- values less than L (call this set of points A) and the remainder points have X-values greater than L (call this set B).
end if
Step2:
Recursively, do the Step1 to find the ranks of points in A and ranks of points in B.
Step3:
Merge the set A and set B:
Use quickSort to sort points in A and B according to their y-values. Scan these points sequentially and determine, for each point in B, the number of points in A whose y-values are less than its y-value. The rank of this point is equal to the rank of this point among points in B (found in Step 2), plus the number of points in A whose y-values are less than its y-value.
```

2.2 Straightforward Approach

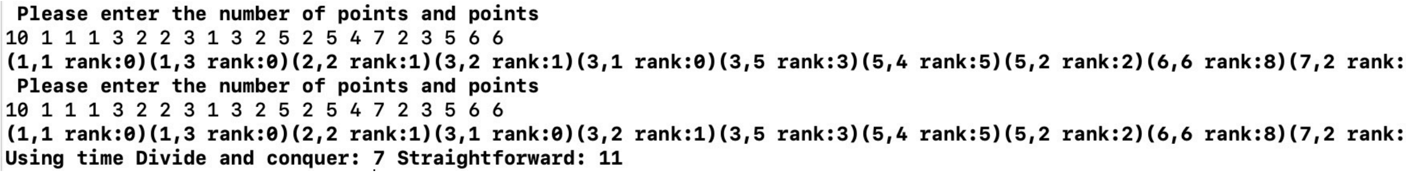
Algorithm 2 Straightforward

```
Input A set S of planar points $P_1, P_2, ..., P_n$.
Output: The rank of every point in S.
for $n = 1$ to number of the points do
    for $m = 1$ to number of the points do
        if $P_n.x > P_m.x$ and $P_n.y > P_m.y$ then
            $P_n.rank+ = 1$
        end if
    end for
end for
```

3 Result Compare and Time Complexity Compare

3.1 Result

We can see the result figure of the code. The first number 10 is the number of points. The others are the points' x and y values. Be careful the position of point(3,1) and (3,2) and reverse in the two result. And we can see the Divide and conquer is faster than the Straightforward.



```
 Please enter the number of points and points
10 1 1 1 3 2 2 3 1 3 2 5 2 5 4 7 2 3 5 6 6
(1,1 rank:0)(1,3 rank:0)(2,2 rank:1)(3,2 rank:1)(3,1 rank:0)(3,5 rank:3)(5,4 rank:5)(5,2 rank:2)(6,6 rank:8)(7,2 rank:
 Please enter the number of points and points
10 1 1 1 3 2 2 3 1 3 2 5 2 5 4 7 2 3 5 6 6
(1,1 rank:0)(1,3 rank:0)(2,2 rank:1)(3,1 rank:0)(3,2 rank:1)(3,5 rank:3)(5,4 rank:5)(5,2 rank:2)(6,6 rank:8)(7,2 rank:
Using time Divide and conquer: 7 Straightforward: 11
```

Figure 1: Result example

3.2 Time Complexity Compare

Divide and conquer：

(l) In Step 1, there is an operation which finds the median of a set of numbers. The lowest time complexity of the median finding algorithm: quickSelect is $O(n)$.

2) In Step 3, there is a sorting operation. The lowest time complexity the sorting algorithm quickSort is $O(nlogn)$.

3)The scanning takes $O(n)$ steps.

4)After calculation the final time complexity is $O(n\ log\ ^2n)$

Straightforward:

It has 2 loops of n so its time complexity is $O(n^2)$