C9. Interview

21600244 박건희 21500800 홍순규

문제 분석

- 최대한 많은 group을 만들어야 하니 Rule 1($x_i > x_j \ AND \ y_i < y_j$, vice versa)을 지키는 applicant 끼리 같은 그룹에 두고, 나머지는 전부 각 각의 그룹에 분리시킨다.
- Rule 1을 만족하는 지원자들을 찾기 위해 다른 지원자들과의 모든 pair를 비교해야 한다.
- 각 지원자마다 다른 사람을 비교할 때, 아래 두 가지 Case로 처리한다.

  Case 1) 자신과 Rule 1 이 성립되는 다른 지원자가 모두 그룹이 없는 경우:
  자신과 다른 지원자들을 새로운 그룹으로 묶는다.

  Case 2) 자신과 Rule 1 이 성립되는 다른 지원자가 한명이라도 이미 다른 그룹에 속한 경우:
  자신과 다른 지원자들을 하나의 다른 그룹에 추가한다.

X	Y	match	is_group
3	6		X
8	13	X	X
11	7	X	X
20	10	X	X

그룹 수: 1

Required Data Structure:

Applicant의 성적 x, y를 기록할 **pair<int, int> array 'applicant'**
Applicant가 현재 그룹이 있는지 확인할 **bool array 'is_group'**
현재 그룹을 추가할지 말지 결정할 **bool variable 'add_group'**

X	Y	match	is_group
8	13		X
11	7	O	X
20	10	X	X

그룹 수: 2

솔루션:

1. x의 오름차순으로 applicant 들을 정렬한다. ← $Average: O(n \log n), Worst: O(n^2)$
2. 1번째 사람부터 자신보다 x값이 큰 모든 사람들과 y값을 비교한다. (x값은 이미 오름차순이기 때문에 y 값이 자기보다 작으면 같은 그룹에 있어야 한다.) ← $O(n^2)$
   - 현재 지원자와 rule 1이 성립하는 지원자의 is_group를 true로 바꾼다. (지원자와 같은 그룹이어야 한다)
     > 이 때 해당 지원자의 is_group가 이미 true면 add_group을 false로 바꾼다.
     > Case 2에 해당하는 것으로 그룹의 개수는 기존의 그룹의 개수와 동일하다.
   - 만약 모두와 비교하는 동안 add_group이 true로 유지된다면 Case 1에 해당하는 것으로 그룹의 개수를 하나 증가시킨다.

X	Y	match	is_group
11	7		O
20	10	O	X

그룹 수: 2 (Case 2)

Time Complexity: $T(n) = O(n^2)$