

# CIRCLE

Sub 60%

Khi thêm vào 1 đường tròn thì số vùng chỉ tăng thêm đúng 1. Vì vậy, đáp án luôn là  $N + 1$ .

Sub 40% còn lại

Khi thêm 1 đường tròn vào mặt phẳng hiện tại thì số vùng tăng lên chỉ 1 hoặc 2. Trường hợp tăng lên 2 vùng chỉ xảy ra khi tồn tại 1 tập các đường tròn con kề nhau chia đường tròn cha thành 2 vùng trên dưới. Vì vậy, mình sẽ sắp xếp các đường tròn tăng dần theo bán kính để đảm bảo khi thêm 1 đường tròn vào thì các đường tròn con bên trong nó đã thêm vào hết.

Gọi  $trace[right]$  là giao điểm bên trái xa nhất của đường tròn có giao điểm phải với trục  $Ox$  là  $right$ . Như vậy, khi thêm 1 đường tròn vào mình sẽ truy vết từ giao điểm phải của đường tròn đó với trục  $Ox$ . Nếu truy vết được đến giao điểm bên trái tức là tồn tại 1 tập các đường tròn con được nói ở trên và đường tròn đang được thêm vào sẽ tăng lượng vùng lên 2, ngược lại tăng 1. Sau đó cập nhật  $trace$  cho đường tròn hiện tại. Vì tọa độ  $right$  khá lớn nên mình sẽ sử dụng `ctdl map` trong C++ để lưu trữ.

Độ phức tạp:  $O(N * \log N)$ . Vì mỗi đường tròn chỉ được gọi tối đa 2 lần trong *map*.

Ngoài ra, có cách sử dụng *stack* và sweep line xử lí nhanh hơn nhưng đpt chung vẫn là  $N * \log N$ .