АНАЛИЗ НА РЕШЕНИЕТО НА ЗАДАЧА ИГРА

Най-важната част от решението е да се забележи, че за да бъде кръг с център O_1 и радиус r във вътрешността на окръжност с център O и радиус R, трябва да е вярно, че: $O_1O + r \leq R$. От условието следва, че неравенството ще е строго, тъй като не се допуска допиране на окръжност и граница на кръг. Ако O_1 е с координати (x_1,y_1) и O е с координати (x,y), то трябва `sqrt((x1-x)*(x1-x) + (y1-x)*(y1-x)) + r < R` да има стойност `true`. Тъй като няма проверка и за равенство, използването на тип `double` е напълно безопасно.

Като се знае това, може за всяка от Q-те заявки лесно да се провери колко от N-те кръга са във всяка от M-те окръжности. Това решение има сложност `O(Q*M*N)` и тъй като толкова пъти изчислява разстоянието между точки, може да върви само за **10% от тестовете**, ако няма други оптимизации.

Ако за `O(M*N)` операции за всяка двойка център-кръг се преизчисли какъв е минималният радиус на окръжност с такъв център, която да съдържа кръга:

`R >= int(ceil(sqrt((x1-x)*(x1-x) + (y1-x)*(y1-x)) + r))` //защото R винаги ще е цяло число

Тогава сложността на програмата отново е `O(Q*M*N)`, но вече е пъти по-бърза от първото решение, затова минава при поне **50% от тестовете**.

Решението за **100 точки** използва факта, че ако за всеки от M-те центъра, всичките N кръга се сортират по нужния радиус на октъжност със съответния център, който ги съдържа (тоест има M сортирани масива, всеки с размер N), може на дадена заявка (и съответно даден радиус на окръжността – R), за който и да е от M-те центъра, да се използва двоично търсене, за да се намери колко кръга биха се съдържали в окръжността. Това решение, като се вземе предвид и нужното време за сортирането, е със сложност $O(M*N*\log_2 N + Q*M*\log_2 N)$.

Автор: Андрей Лалев