АНАЛИЗ НА РЕШЕНИЕТО НА ЗАДАЧА
СПЕКТРИ

Както и в самото условие е казано, задачата се свежда до това да покрием изцяло максимален брой кръгове с еднакъв радиус в равнината, с един по-голям кръг. Както други геометрични задачи, които включват кръгове с еднакви радиуси, и в тази можем да приложим един известен трик, облекчавайки я допълнително. Можем да смалим кръговете до точки и да намалим големия радиус с малкия радиус, като така вместо кръгове разглеждаме точки. Оттам нататък центровете на кръговете стават точките, които разглеждаме, и искаме да покрием максимален брой от тях с кръг с радиус R – r.
Нещо, което се очаква повечето състезатели да се досетят е, че при оптимално покриване на множество точки с кръг винаги бихме могли да имаме поне 2 точки на границата на кръга. Така лесното решение на задачата е да фиксираме две от точките, които ще са на ръба (възможно е те да са повече от две, но ние искаме да фиксираме само 2 от тях). Макар и да са нужни 3 точки за да определим еднозначно една окръжност, в случая, където знаем неговия радиус, две точки почти я определят – има само два варианта, в които би могла да бъде тя. След като фиксираме точките, можем просто да разгледаме двата случая и да преброим колко от останалите точки биха били покрити от кръга във всеки от тези случаи – което би решило задачата ни. Тъй като не бихме могли да фиксираме две от N точки за под квадратично време, а проверката колко от останалите точки лежат вътре е линейна (освен ако не използваме много сложни структури данни, които са далеч извън възможностите за написване на състезание за ученици), общата сложност на това решение е O(N^3). Това е добро решение, тъй като е доста по-просто за измисляне от решението за 100 точки, а от гаранцията за поне 50% от тестовете да са с ограничение N <= 100, знаем, че ще хванем поне половината точки (всъщност 65 или 70, при добра имплементация). Все пак за това решение се изисква известна досетливост - смаляването на кръгове до точки, а също и някакви основни познания по геометрия за намиране на центъра на големия кръг по 2 точки (тоест намиране на симетрала на 2 точки).
Решението за 100 точки е доста различно – за него трябва да сведем задачата още веднъж до по-проста такава. За целта избираме всяка от точките и я обявяваме за център на координатната система. Тя ще бъде една от точките по края на кръга. След това сортираме останалите точки по полярен ъгъл спрямо нея. Можем да забележим, че ако почнем да въртим кръга около центъра на координатната система (тоест точката, която сме избрали) точките ще влизат една по една и след това ще излизат (това можете лесно да си го представите като забит пирон в земята и въртите пръстен около него). Нещо повече – тъй като сме ги сортирали по полярен ъгъл, те ще влизат и излизат точно в този ред, в който са след сортирането. Това дава възможност да използваме метода на метящата линия (sweep line) и да намерим оптималния отговор, при дадената фиксирана точка. Тъй като прилагаме въпросния алгоритъм за всяка от точките, то със сигурност ще намерим и глобалния оптимален отговор. Сложността на този алгоритъм е O(N) за фиксиране на център на координатната система, и още O(N * logN) за сортирането на точките по полярен ъгъл, последвано от O(N * logN) за sweep line алгоритъма. Тъй като соритрането и метящата линия се изпълняват един СЛЕД друг, а не един в друг, общата сложност остава O(N * N * logN). Това решение би хванало 100 точки.

Макар и това да беше началната идея на задачата, докато писах решението си, успях да реализирам по-лесен алгоритъм, който върши същото нещо (като ползва друга известна задача в основата си). Нека отново разгледаме въртенето на кръга около точката – в някакъв момент една точка влиза в него, и след известно време излиза. Можем да запишем ъгъла, в който влиза, и ъгъла, в който излиза. Той ни дава някакъв интервал от ъгли (освен когато втората точка не е твърде далеч за да влезе изобщо, в който случай не я разглеждаме), в които тя би била вътре в кръга, заедно с първата. Ако направим това за всяка точка (която е достатъчно близо до фиксираната), ще получим множество от интервали. Реално, максималното покриване на тези интервали ще бъде и максималният отговор на подзадачата с този център на координатната система. Тази задача също е известна, и отново се решава чрез сортиране. Всъщност като идея тя е почти същата като sweep line, но се реализира малко по-лесно. Сложността отново е O(N * N * logN).
Има няколко частни случая, които не трябва да забравяме. Първият от тях е, когато 2 точки са точно на разстояние 2(R-r) една от друга – тогава интервалът на ъглите всъщност е точка. Това като обаче за щастие не ни пречи изобщо. Другият по-голям проблем е, ако някъде между началото и края на интервала се преминава през 360 градуса (тоест крайният ъгъл е по-малък от началния). За целта просто вкарваме всеки от интервалите два пъти: веднъж като (start, end), и още веднъж като (start + 360, end + 360). Това решава този проблем и не усложнява по никакъв начин алгоритъма (само го прави около два пъти по-бавен). Също съществува частния случай, в който R = r, (тогава трябва да върнем 1), но при добра реализация на горните алгоритми това не е проблем и не трябва да се пише допълнителен код за да се handle-ва този случай explicitly.

Автор: Александър Георгиев