

Анализ за задача Artillery

Нека изберем някой връх за корен на дървото. Нека той е X и поддърветата му бъдат S_1, \dots, S_M .

Полезно е да мислим за задачата като динамично. Да допуснем че знаем минималните K необходими да унищожим пионката във всяко от поддърветата: $\min K(S_i)$. Тогава можем да стигнем до следната идея: Да използваме $\min K(S_1)+1$ за да унищожим пионката ако е в S_1 като всеки път удряме и корена за да не може пионката да избяга от S_1 . Същото използвайки $\min K(S_2)+1$ и т.н. до $\min K(S_M)+1$.

Така отговорът за дървото ще бъде $\max(\min K(S_i)+1)$. И имплементирайки това решение то ще бъде със сложност $O(N*N)$, понеже трябва да пробваме за всеки избор на X да пресметнем $\min K$ за всеки връх и поддървото му. По-неприятното е че това решение не винаги ще дава оптимален резултат. Например за пътечка ще дава $(N-1)/2$ закръглено нагоре (при X средния връх на пътечката), като оптималния отговор е 2.

И тук реално се появява идеята че не е наложително винаги да удряме корена. И наистина интересния въпрос е кога трябва да удряме корена.

Ако не ударим корена това позволява пионката да влезне в корена от някое от поддърветата в които пионка може да се намира и после да влезна в кое да е поддърво стига да не е възспряна от друг изстрел.

Има 3 начина да си позволим корена да не е обстрелян. Единия е да удряме някои от върховете точно под корена, но това е по-зле от просто да ударим корена. Друг вариант е все още да не сме гарантирали за никое от поддървета че пионката не може да е в него. Тогава просто не ни пречи пионката да сменя поддървета. Трети вариант е да остава само едно поддърво в което да е пионката и просто да я „натикаме“ по надолу в това поддърво.

Това дава следната идея за обстрелване: Първо да обстреляме едно поддърво „отдолу нагоре“ (т.е. да изгоним пионката от него към корена), когато не ни се налага да удряме корена. После да удряме от оставащите всички без едно удряйки корена (както при първата идея). И накрая като остава само едно поддърво в което да може да е пионката да обстреляме това поддърво „отгоре надолу“ (т.е. да не даваме на пионката да излезе през корена в процеса на обстрелване) при което отново не ни се налага да удряме корена.

Като реализация това ще изглежда като динамично със 2 състояния: корен на поддърво и дали да го обстреляме поддървото „отгоре надолу“, „отдолу нагоре“ или нормално. И реално по-съображенията по горе можем да изградим рекурсивните връзки:

- $\min K(X, \text{normal}) = \max(\min K(S_i, \text{up}), \min K(S_j, \text{down}), 1 + \max_{p=i,j}(\min K(S_p, \text{normal})))$ за най-добрите i и j
- $\min K(X, \text{up}) = \max(\min K(S_i, \text{up}), 1 + \max_{p=i}(\min K(S_p, \text{normal})))$ за най-доброто i
- $\min K(X, \text{down}) = \max(\min K(S_i, \text{down}), 1 + \max_{p=i}(\min K(S_p, \text{normal})))$ за най-доброто i

Най-добрите i и j могат да се намерят за линейно време по броя наследници. И така намирането на $\min K$ при някой избран корен става за време $O(N)$ така че общо решението става за време $O(N*N)$.

Това решение вече работи оптимално. Реално вече сме показали че това е оптималното ако обстрелваме поддърветата едно по едно с разсъжденията за това кога може да не стреляме

корена. И е очевидно че не си спестяваме оръдия ако стреляме по над едно поддърво по едно и също време.

Сега ни остава само да подобрим това решение така че сложността да стане $O(N)$. Ами има няколко начина. Интуитивния (и по неприятния за правене) е да си изберем един връх за корен и докато се спускаме по дървото да си пазим всичко над текущия връх като още едно поддърво. И така ще пробваме всички дървета да са корени само с 2 обхождания.

По-лесния начин изисква още едно наблюдение. Че всеки връх който не е листо ще ни даде оптимално решение ако бъде избран за корен. Това е така защото такъв връх има поне 2 поддървета т.е. може да се получи както обстрелването „отдолу нагоре“ така и обстрелването „отгоре надолу“, което всъщност би гарантирало че отговора ще е същия ако оптималния корен е в едно от тези 2 поддървета. А тези 2 поддървета се избират оптимално така че оптималния корен би се намирал в някое от тях.

Така най-простото решение е за X да изберем връх който не е листо и да пуснем обхождането обсъдено по-горе. Това дава време за решение $O(N)$.

Автор: Иво Дилов