Opisi algoritama

IBT 2010., 1. kolo, juniori P1

1 Trojica

Za svaki od unesenih trokuta računamo opseg i površinu kako je opisano u zadatku. Želimo li izbjeći korjenovanje, usporedit ćemo kvadrat opsega i kvadrat površine - ako su jednaki, ispisujemo redni broj trokuta.

2 Bojanje

Za svaku riječ potrebno je pronaći koliko se puta koji broj pojavljuje u njoj. Svaku riječ rješavamo tako da prođemo po svakoj znamenki u njoj i povećamo vrijednost za 1 na poziciji (u pomoćnom nizu) koja odgovara vrijednosti te znamenke. Nakon toga samo nađemo prvu najveću vrijednost u pomoćnom nizu.

3 DNK

Ukupan broj molekula duljine N (bez obzira da li eksplodiraju) iznosi 2^N (jer za svaku od N pozicija imamo 2 mogućnosti - staviti X ili Y). Lakše je izračunati broj molekula koje *neće* eksplodirati (označimo taj broj s $f(N)$), pa će konačno rješenje biti $2^N - f(N)$.

Uzmimo kao početne uvjete $f(2) = 3$ i $f(3) = 5$. Kako izračunati $f(K)$, za $K > 3$?

• Ako je K-ti (posljednji) znak molekule jednak X, prethodni mora biti Y (inače bi molekula eksplodirala), a prvih $K - 2$ znakova može biti bilo koja neeksplodirajuća molekula duljine $K - 2$, za što postoji $f(K - 2)$ mogućnosti.

• Ako je K-ti (posljednji) znak molekule jednak Y, prvih $K - 1$ znakova može biti bilo koja neeksplodirajuća molekula duljine $K - 1$, za što postoji $f(K - 1)$ mogućnosti.

Dakle, vrijedi $f(K) = f(K-2)+f(K-1)$. Koristeći ovu relaciju (kao kod Fibonaccijeva niza) redom računamo $f(4), f(5), ..., f(N)$ (pamteći uvijek samo ostatak pri dijeljenju sa 1 000 000 007).

4 Katapult

Neka je $f(K)$ traženi broj otoka koji ćemo vidjeti, krenemo li s otoka K. Neka je $s(K)$ otok u koji dolazimo izravno iz K. Lako je naći sljedeću relaciju:

$$f(K) = \begin{cases} f(s(K)) + 1, & \text{ako nećemo ponovno doći u } K\text{, tj. } K \text{ ne pripada ciklusu,} \\ \text{duljina ciklusa,} & \text{ako se } K \text{ nalazi u ciklusu.} \end{cases}$$

Vidimo da najprije trebamo pronaći cikluse, a nakon toga za vrhove koji se ne nalaze u ciklusima rekurzivno (s memoizacijom) računati $f(K)$ pomoću gornje relacije.

Kako pronaći cikluse? Algoritam je sljedeći: krećemo na putovanje iz nekog još neposjećenog otoka (dok takav postoji). Putovanje prekidamo u ova dva slučaja:

• Došli smo do otoka koji smo u istom putovanju već bili posjetili. Tada je on član ciklusa, pa otoke u tom ciklusu označimo i izbrojimo.

• Došli smo do otoka koji smo bili posjetili u nekom prethodnom putovanju. To znači da otoci kojima smo upravo putovali ne pripadaju ciklusu.

U oba slučaja prolazit ćemo samo po još neposjećenim otocima (i to najviše dva puta), pa složenost algoritma (kao i složenost opisanog memoizacijskog računanja $f(K)$) iznosi $O(N)$.

5 Stonoga

Pretpostavimo da želimo dodati K slova na kraj danog stringa. Palindrom tada mora imati $N+K$ slova, a budući da je K manji od N, prva polovina tog palindroma nam je poznata. Druga polovina palindroma mora biti jednaka obrnutoj prvoj, pa je ona jednoznačno određena.

Dakle, za svaki K potrebno je naći string koji predstavlja poznati dio druge polovine palindroma i usporediti ga s njemu odgovarajućim dijelom prve polovine palindroma - ako su oni jednaki, palindrom je moguće načiniti. Nije teško izvesti formule koje daju pozicije ovih stringova (valja uzeti u obzir ima li palindrom parno, ili neparno mnogo slova).

Još je potrebno brzo usporediti ove stringove. U tu svrhu koristimo poznatu *hash* funkciju, koja svakom stringu pridružuje cijeli broj i tako uspoređivanje stringova svodi na uspoređivanje dvaju brojeva.

Kako brzo izračunati *hash* nekog podstringa? Treba uočiti da, znamo li $hash(S_1)$ i $hash(S_1 + S_2)$, možemo izračunati $hash(S_2)$. Dakle, ako odmah nakon unosa izračunamo *hahseve* svih prefiksa (stringova koji započinju prvim slovom danog stringa), možemo izračunati *hash* bilo kojeg podstringa znajući samo njegovu početnu i krajnju poziciju. Isto možemo činiti i za obrnuti string (sjetimo se, u svakom koraku trebamo usporediti jedan "obični" i jedan obrnuti podstring).

Složenost algoritma iznosi $O(N)$.

Komentare i pitanja uputite na frane.kurtovic@gmail.com *ili na* askurdija@gmail.com.