

Riterių turnyras

Milano kunigaikštis Lodovico Sforza 1491 metais ruošdamasis vestuvėms su Beatrice d'Este paprašė Leonardo suplanuoti vestuves, kurių metu vyktų tris dienas truksiantis riterių turnyras. Deja, populiariausias riteris vėluoja...

Turnyras

Turnyro pradžioje N riterių susirikiuoja viena eile, pozicijose, sunumeruotose nuo 0 iki $N - 1$. Turnyras sudarytas iš atskirų turų. Turnyro vedantysis pradeda *turą* paskelbdamas dvi pozicijas S ir E (kur $0 \leq S < E \leq N - 1$). Visi riteriai, kurių pozicijos yra nuo S iki E (imtinai) dalyvauja tame ture, laimėtojas grįžta į savo poziciją eilėje ir tęs turnyrą, o pralaimėjusieji apleidžia lauką. Tuomet riteriai susiglaudžia: užima eilėje pozicijas 0 iki $N - (E - S) - 1$ nekeisdami tarpusavio stovėjimo tvarkos ir nepalikdami tuščių pozicijų.

Turnyro vedantysis skelbia kitą turą ir taip tęsia tol, kol lieka vienas riteris.

Leonardo žino, kad visų riterių pajėgumai skiriasi: juos numeruokime nuo 0 (silpniausias) iki $N - 1$ (stipriausias). Jis taip pat tiksliai žino, kokius turus (turą nusako du skaičiai) kokia tvarka skelbs vedantysis ir kiek iš viso turų bus (pažymėkime jų skaičių C). Juk vis tik jis yra Leonardo... ir jis yra tikras, kad kiekvieną turą laimės pajėgiausias riteris.

Pavėlavęs riteris

Eilėje jau stovi $N - 1$ riterių, o paskutinytis, populiariausias riteris (jo pajėgumas yra R) dar neatvyko. Siekdamas išnaudoti šio riterio populiarumą, Leonardo nori riteriui parinkti tokią poziciją, kad šis laimėtų didžiausią galimą turų skaičių. Atkreipiame dėmesį, kad turai, kuriuose šis riteris nedalyvauja, nedomina. Domina tik tie turai, kuriuose jis dalyvauja ir laimi.

Pavyzdys

Tarkime, viso yra $N = 5$ riterių ir $N - 1$ riterių jau sustojo į eilę ir jų pajėgumai atitinkamai yra $[1, 0, 2, 4]$. Pavėlavusio riterio pajėgumas $R = 3$. Turnyro vedantysis numatė $C = 3$ turus ir skelbs turus (t.y. pozicijas (S, E)) tokia tvarka: $(1, 3)$, $(0, 1)$, $(0, 1)$.

Jei Leonardo pavėlavusiam riteriui parinks pirmą poziciją, tuomet riterių pajėgumai bus tokie: $[3, 1, 0, 2, 4]$. Pirmame ture dalyvaus riteriai stovintys pozicijose 1, 2, 3, jų pajėgumai atitinkamai yra 1, 0, 2. Turą laimės riteris, kurio pajėgumas 2. Likę trys riteriai sustos į eilę ir jų pajėgumai atitinkamai yra $[3, 2, 4]$. Tolesniame ture dalyvaus riteriai, kurių pajėgumai yra 3 ir 2 (stovintys pozicijose 0 ir 1). Laimės riteris, kurio pajėgumas $R = 3$, ir turnyre liks du riteriai, kurių pajėgumai yra $[3, 4]$. Paskutinį turą (skelbiamos pozicijos 0, 1) laimės riteris, kurio pajėgumas 4. Taigi, pavėlavęs riteris laimės tik vieną (antrąjį) turą.

Tuo tarpu, jei Leonardo įterptų pavėlavusį riterį tarp riterių, kurių pajėgumai 1 ir 0, pradinio momentu riterių pajėgumai išsidėstytų taip: [1, 3, 0, 2, 4]. Šį kartą pirmame ture dalyvautų riteriai, kurių pajėgumai 3, 0, 2, ir laimėtų pavėlavęs riteris, kurio pajėgumas $R = 3$. Tuomet susirikiuotų tokia riterių eilė: [1, 3, 4]. Tolesnį turą (kaunasi riteriai, kurių pajėgumai 1 ir 3) vėl laimėtų pavėlavęs riteris, kurio pajėgumas $R = 3$. Likę du riteriai sustotų tvarka [3, 4] ir paskutinįjį turnyrą laimėtų riteris, kurio pajėgumas 4. Pavėlavęs riteris laimėtų du turus ir tai yra didžiausias galimas turų, kuriuos gali laimėti šis riteris, skaičius.

Užduotis

Parašykite programą, kuri rastų tokia pradinę poziciją pavėlavusiam riteriui, kad dalyvaudamas turnyre jis laimėtų didžiausią galimą turų skaičių, kaip to nori Leonardo. Konkrečiau, reikia parašyti paprogramę, pavadintą `GetBestPosition(N, C, R, K, S, E)`, kur:

- N - riterių skaičius;
- C - turų skaičius turnyre ($1 \leq C \leq N - 1$);
- R - pavėlavusio riterio pajėgumas; visų riterių pajėgumai (jau stovinčių eilėje ir pavėlavusio) yra skirtingi skaičiai iš intervalo 0, ..., $N - 1$; pavėlavusio riterio pajėgumas R pateiktas atskirai, nors jis ir gali būti apskaičiuojamas iš kitų duomenų;
- K - masyvas iš $N - 1$ sveikųjų skaičių, nurodantis nepavėlavusių riterių pajėgumus tokia tvarka, kuria jie sustojo į eilę turnyro pradžioje;
- S ir E - du masyvai, kurių dydis C : kiekvienam i iš intervalo nuo 0 iki $C - 1$ imtinai, $(i + 1)$ -ojo turo metu bus pakviesti kautis riteriai, kurių pozicijos nuo $S[i]$ iki $E[i]$ imtinai. Kiekvienam i galios: $S[i] < E[i]$.

Paprogramei perduodami duomenys yra korektiški: $E[i]$ yra mažesnis už $(i + 1)$ -ame turnyre dalyvaujančių riterių skaičių, o po C turų liks lygiai vienas riteris.

`GetBestPosition(N, C, R, K, S, E)` turi grąžinti poziciją P , kur Leonardo turi įterpti pavėlavusį riterį ($0 \leq P \leq N - 1$). Jei yra kelios galimos pozicijos, grąžinkite mažiausią. Pozicija P , iš esmės, yra riterių, stovinčių eilėje prieš pavėlavusį riterį, skaičius optimaliame sprendinyje. Jei $P = 0$, tai reiškia, kad riteris stovi eilės pradžioje, o jei $P = N - 1$, tai reiškia, kad riteris stovi eilės gale.

1 užduotis [17 taškų]

$N \leq 500$.

2 užduotis [32 taškai]

$N \leq 5\,000$.

3 užduotis [51 taškas]

$N \leq 100\,000$.

Realizacija

Reikia pateikti vieną failą, pavadintą `tournament.c`, `tournament.cpp` arba `tournament.pas`. Faile turi būti realizuota aukščiau aprašyta paprogramė su tokia antrašte.

Programuojantiems C/C++

```
int GetBestPosition(int N, int C, int R, int *K, int *S, int *E);
```

Programuojantiems Paskaliu

```
function GetBestPosition(N, C, R : LongInt; var K, S, E : array of LongInt) : LongInt;
```

Šios paprogramės turi veikti taip, kaip aprašyta aukščiau. Be abejo, galite sukurti daugiau paprogramių vidiniam vartojimui. Jūsų pateiktas sprendimas neturi dirbti (skaityti ar rašyti) nei su standartiniu įvedimo/išvedimo įrenginiu, nei su jokia kitu failu.

Pavyzdinis vertintojas

Pavyzdinis vertintojas duomenis perskaito tokiu formatu:

- 1-oji eilutė: N, C, R ;
- 2-oji, ..., N -oji eilutės: $K[i]$;
- $(N + 1)$ -oji, ..., $(N + C)$ -oji eilutės: $S[i], E[i]$.

Laiko ir atminties ribojimai

- Laiko ribojimas: 1 sekundė.
- Atminties ribojimas: 256 MiB.