

Piigivõitluse turniir

Kui Milaano hertsog Lodovico Sforza 1491. aastal Beatrice d'Este naiseks võttis, laskis ta Leonardol koordineerida pulmapidustusi, mille hulka kuulus ka tervelt kolm päeva kestnud piigivõitluse turniir.

Turniir

Turniiril rivistatakse N rüütli sirgesse ritta ja nummerdatakse nende kohad rivis 0 kuni $N-1$. Turniiri iga *vooru* alguses hõikab kohtunik välja kahe koha numbrid S ja E ($0 \leq S < E \leq N-1$). Kõik rüütli kohadelt S kuni E (kaasa arvatud) võistlevad selles voorus. Vooru võitja läheb rivisse oma kohale tagasi, kõik ülejäänud langevad turniirilt välja ja lahkuvad. Seejärel nihkuvad kõik allesjäänud rüütli omavahelist järjekorda säilitades rivi alguse poole, hõivates kohad 0 kuni $N-(E-S)-1$. Seejärel hõikab kohtunik välja järgmise vooru ja kõik kordub, kuni alles on jäänud vaid üks rüütel.

Leonardo teab kõigi rüütli (paarikaupa erinevaid) tugevusi, mida ta tähistab arvudega 0 (kõige nõrgem) kuni $N-1$ (kõige tugevam). Lisaks teab ta, millised kohtadepaarid kohtunik igaühes turniiri C voorust välja kuulutada plaanib (ta on ju geeniaalne Leonardo...) ja on kindel, et igas voorus võidab tugevaim rüütel.

Hilineja

$N-1$ rüütli on juba kohal ja rivis, ainult kõige populaarsem rüütel on veel puudu. Tema tugevus on R ja ta jääb natuke hiljaks. Turniiri atraktiivsuse huvides tahab Leonardo paigutada selle rüütli rivisse nii, et ta võidaks turniiril maksimaalse arvu voorusid. Tasub tähele panna, et meid huvitavad ainult voorud, mille see rüütel võidab ja me ei hooli voorudest, kus ta üldse ei osale.

Näide

Olgu $N = 5$ ja kohaloleva $N-1$ rüütli tugevused rivistuse järjekorras $[1, 0, 2, 4]$. Siis peab hilineja tugevus olema $R = 3$. Olgu kohtunikul plaanis korraldada turniir kolmes voorus ja hõigata (S, E) paaridena välja vastavalt $(1, 3)$, $(0, 1)$ ja $(0, 1)$.

Kui Leonardo paigutaks hilineja rivi algusse, oleks tugevuste jada $[3, 1, 0, 2, 4]$. Esimeses voorus võistleks kohtadel 1, 2 ja 3 olevad rüütli, kelle tugevused on vastavalt 1, 0 ja 2; seega oleks vooru võitja rüütel 2 ja uus jada $[3, 2, 4]$. Järgmises voorus võistleks rüütli 3 ja 2 (kohtadelt 0 ja 1); rüütel $R = 3$ võidaks ja uus jada oleks $[3, 4]$. Viimases voorus (jälle kohtade 0 ja 1 vahel) võidaks rüütel 4. Seega saaks hilineja võita ainult ühe korra (teises voorus).

Kui Leonardo paigutaks hilineja rüütli 1 ja 0 vahele, oleks jada $[1, 3, 0, 2, 4]$. Siis võistleks esimeses voorus rüütli 3, 0 ja 2 ning rüütel $R = 3$ võidaks. Järgmine voor algaks jadaga $[1, 3, 4]$ ning rüütli 1 ja 3 vaheliste võitluse võidaks jälle rüütel $R = 3$. Enne viimast vooru oleks jada $[3, 4]$

ja rüütel 4 võidaks. Seega saaks hilineja sel juhul võita kahes voorus, mis on ka parim võimalik tulemus, kuna kõigis kolmes voorus ei saaks ta võita mitte mingi paigutusega.

Ülesanne

Kirjutada programm, mis valib hilinejale parima võimaliku positsiooni (sellise, mis maksimeerib tema võitude arvu, nagu Leonardo tahab). Täpsemalt tuleb realiseerida funktsioon `GetBestPosition(N, C, R, K, S, E)`, kus:

- N on rüütlike arv;
- C on turniiri voorude arv ($1 \leq C \leq N-1$);
- R on hilineja tugevus; kuna kõigi rüütlike tugevused on erinevad ja esitatud täisarvudena $0, \dots, N-1$, saaks hilineja tugevuse teistest tuletada, aga siiski on see ka ilmutatult antud;
- K on $N-1$ elemendiga massiiv, mis annab juba ravis olevate rüütlike tugevused;
- S ja E on C elemendiga massiivid; iga $i = 0 \dots C-1$ (kaasa arvatud) korral kutsub kohtunik $(i+1)$. voorus võistleva rüütli kohtadelt $S[i]$ kuni $E[i]$ (kaasa arvatud) ja võib eeldada, et $S[i] < E[i]$.

Funktsioonile antav kohtuniku kutsete jada on korrektne: iga $E[i]$ on väiksem kui $(i+1)$. vooru alguseks alles olevate rüütlike arv ja kõigi C vooru järel jääb alles ainult üks rüütel.

`GetBestPosition(N, C, R, K, S, E)` peab tagastama parima positsiooni P ($0 \leq P \leq N-1$), kuhu Leonardo peaks hilineja paigutama. (Positsioon P on hilineja kohanumber jadas pärast tema lisamist; kohtade numeratsioon algab nullist. Teisisõnu, P on jadas hilineja positsioonist eespool seisvate rüütlike arv. $P = 0$ tähendaks, et hilineja peab minema rivi algusse, ja $P = N-1$, et ta peab minema rivi lõppu.)

Alamülesanne 1 [17 punkti]

Võib eeldada, et $N \leq 500$.

Alamülesanne 2 [32 punkti]

Võib eeldada, et $N \leq 5\,000$.

Alamülesanne 3 [51 punkti]

Võib eeldada, et $N \leq 100\,000$.

Realisatsioon

Lahendusena tuleb esitada üks fail nimega `tournament.c`, `tournament.cpp` või `tournament.pas`. See fail peab realiseerima eelpool kirjeldatud alamprogrammid vastavalt alltoodud signatuuridele.

C/C++ programmid

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

Pascali programmid

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

Need alamprogrammid peavad töötama nagu eelpool kirjeldatud. Muidugi võib nende realiseerimiseks kirjutada ka teisi sisemisi alamprogramme. Lahendus ei tohi pöörduda standardsisendi, standardväljundi ega ühegi teise faili poole.

Lokaalne hindaja

Lokaalne hindaja ootab sisendit järgmises vormingus:

- 1. rida: N, C, R;
- read 2, ..., N: K[i];
- read N+1, ..., N+C: S[i], E[i].

Aja- ja mälupiirangud

- Ajapiirang: 1 sekund.
- Mälupiirang: 256 MiB.