

Turnir

Netko se s nekim trebao vjenčati, pa su žicali Leonarda da bude wedding planner, a u sklopu vjenčanja održavao se i veliki viteški turnir koji je trajao 3 dana. Ali najpopularniji vitez je javio da će malo zakasniti...

Turnir

Na viteškom turniru, N vitezova se nalaze u ravnoj liniji i redom prema poretku označeni su brojevima od 0 do $N - 1$. Direktor turnira svaku borbu određuje na način da odredi 2 pozicije S i E (gdje je $0 \leq S < E \leq N - 1$). Svi vitezovi koji su na pozicijama od S do E (uključivo) se zatim bore: pobjednik ostaje u turniru i vraća se na svoju poziciju, a svi ostali luzeri ispadaju iz turnira i idu doma. Nakon toga, svi vitezovi ostaju u poretku u kojem su bili ali njihove oznake pozicija se renumeriraju na način da su sada označeni brojevima od 0 do $N - (E - S) - 1$. Direktor turnira tada određuje novu borbu, i tako dalje sve dok samo jedan vitez ne ostane kao pobjednik.

Za svakog viteza poznata je njegova snaga tj. snaga svakog viteza je broj između 0 (najslabiji) i $N - 1$ (najjači) i to tako da ne postoje 2 viteza s istom snagom. Također, poznati su nam točni podaci o C borbi koje će odrediti direktor turnira. U svakoj borbi, pobijedit će najjači vitez.

Vitez koji kasni

$N - 1$ od ukupno N vitezova se već nalazi poredano na polju, jedino nedostaje najpopularniji vitez čija snaga je R . Za dobrobit turnira, Leonardo želi pozicionirati viteza koji kasni na onu poziciju koja će što više povećati broj borbi u kojima će taj vitez sudjelovati. Naglašavamo da nas zanima samo broj borbi u kojima naš vitez sudjeluje, a ne zanimaju nas borbe u kojima njega nema.

Primjer

Za $N = 5$ vitezova, $N - 1$ su već poredani u liniji i imaju snage $[1, 0, 2, 4]$. Prema tome, okašnjeli vitez ima snagu $R = 3$. U tri borbe, direktor turnira zove na sljedeće borbe, označene parovima (S, E) : $(1, 3)$, $(0, 1)$, $(0, 1)$.

Ako Leonardo smjesti okašnjelog viteza na prvu poziciju, snaga vitezova će biti $[3, 1, 0, 2, 4]$. Prva borba uključuje vitezove na pozicijama 1, 2 i 3, koji imaju snage 1, 0 i 2, te pobjeđuje vitez 3. U sljedećoj borbi $(0, 1)$ vitez 1 pobjeđuje i ostaje linija $[3, 4]$. U zadnjoj borbi pobjeđuje vitez 4. Prema tome, okašnjeli vitez pobjeđuje samo jednu borbu (drugu).

Zadatak

Napišite program koji će odabrati najbolju poziciju za okašnjelog viteza, tako da broj borbi iz teksta zadatka bude maksimalan. Preciznije, morate napisati funkciju `GetBestPosition(N, C, R, K, S, E)`, gdje je:

- N je ukupni broj vitezova;
- C je broj borbi ($1 \leq C \leq N - 1$);
- R je snaga okašnjelog viteza; snage svih vitezova (uključujući okašnjelog i sve one normalne) su međusobno različiti brojevi od 0 do $N - 1$, a snaga R je zadana eksplicitno, iako se može dobiti i na temelju podataka o snagama ostalih vitezova.
- K je niz od $N - 1$ integers, snage $N - 1$ vitezova;
- S i E su dva niza duljine C : za svaki broj i između 0 i $C - 1$, uključivo, borba $i + 1$ će uključiti sve vitezove od pozicije $S[i]$ do pozicije $E[i]$, uključivo. Za svaki i , $S[i] < E[i]$

Svi pozivi vašoj funkciji su ispravno: $E[i]$ je manji od broj trenutno preostalih vitezova, a nakon svih C borbi ostat će nam točno jedan vitez.

`GetBestPosition(N, C, R, K, S, E)` mora vratiti najbolju poziciju P gdje Leonardo mora smjestiti viteza koji kasni ($0 \leq P \leq N - 1$). Ako postoji više ekvivalentnih pozicija, *vratite najmanju*. (Pozicija P označava gdje će vitez biti u konačnoj liniji. Drugim riječima, P je broj vitezova koji stoji prije okašnjelog viteza u optimalnom rješenju. $P = 0$ znači početak linije, dok $P = N - 1$ znači kraj iste.)

Podzadatak 1 [17 bodova]

- $N \leq 500$.

Podzadatak 2 [32 boda]

- $N \leq 5\,000$.

Podzadatak 3 [51 bod]

- $N \leq 100\,000$.

Implementacija

Morate predati točno jednu datoteku, pod nazivom `tournament.c`, `tournament.cpp` or `tournament.pas`. Datoteka mora implementirati gore opisani potprogram koristeći sljedeće deklaracije.

C/C++ programi

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

Pascal programi

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

Vaš potprogram se mora ponašati kao što je gore opisano. Naravno, možete slobodno implementirati druge pomoćne funkcije. Ne smijete koristiti standardni ulaz ili izlaz, kao i pisanje ili čitanje po bilo kakvim datotekama.

Primjer sustava za testiranje

Sustav za testiranje na raspologanju prima ulaz kako slijedi:

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

Vremenska i memorijska ograničenja

- Vremesko ograničenje: 1 sekunda.
- Memorijsko ograničenje: 256 MiB.