

Тэмцээн

Милан хотын Гүнтэн Lodovico Sforza гүнж Beatrice d'Este-тай 1491 онд хуримаа хийхдээ Леонардогоос хуримын ёслолын үеэр гурван өдөр үргэлжлэх жадны тэмцээн зохион байгуулахыг хүсчээ. Гэвч олны танил нэгэн баатар тэмцээн эхлэх мөчид хараахан ирж амжаагүй байлаа.

Тэмцээний дүрэм

N баатар оролцсон жадны тулаан дараах дүрмээр зохиогддог. Тэмцээний эхлэлд баатрууд нь 0 -оос $N-1$ гэсэн байрлалд санамсаргүйгээр эгнэн зогсдог. Тэмцээний "нэг үе" гэдэг нь S, E ($0 \leq S < E \leq N - 1$ байх) хоёр тоогоор тодорхойлогдоно. Энэхүү үед S -ээс E (энэ үеийн тулаанд S болон E дугаартай байрлалд байгаа баатрууд мөн тулалдана) хүртэлх дугаартай байранд зогсож байгаа баатрууд хоорондоо тулалдаж нэг л ялагч тодордог. Тухайн үеийн ялагч нь жагсаалдаа буцаж өөрийн байрлалд орно. Ялагдагчид тэмцээнээс хасагдаж жагсаалыг орхино. Түүний дараа үлдсэн баатрууд жагсаалын эхлэл рүү зэрэгцэн зогсоно. Өөрөөр хэлбэл дундах хоосон зайнуудаа нөхөж зогсоно. Ингэснээр тэдний жагсаал дахь дугаар нь 0 -оос $N - (E - S) - 1$ болж шинчлэгдэнэ. Тэмцээнийг нэг баатар үлдэх хүртэл энэ мэтчилэн үргэлжлүүлдэг.

Леонардо хуримын тэмцээнд 0 -оос $N-1$ гэсэн ялгаатай чансаатай N баатрууд оролцох гэж байгааг мэдэж байгаа. 0 чансаатай нь хамгийн чадал муутай нь, $N-1$ чансаатай нь хамгийн хүчтэй нь. Мөн тэрээр жадны тэмцээн C үетэйгээр зохион байгуулагдах гэж байгааг мэдэж байгаа. Тэр чинь Леонардо шүү дээ. Мөн үе бүрт хамгийн өндөр чансаатай нь тухайн үеийнхээ ялагч болно гэдгийг ч мэдэж байгаа.

Хоцорсон баатар

N баатруудын $N-1$ баатрууд нь аль хэдийн тэмцээнд хүрэлцэн ирж санамсаргүйгээр эгнэн жагссан. Гэвч олны танил нэгэн баатар хараахан ирж амжаагүй байлаа. Тэр баатар нь R чансаатай бөгөөд тэмцээнд хоцорч ирэв. Леонардо тэмцээнийг илүү сонирхолтой болгох үүднээс олны танил тэрхүү баатрыг хамгийн олон үед ялагч болгох жагсаал дахь байрлалд нь оруулахаар шийджээ. Үүнд тухайн баатрын оролцоогүй үеүдийн тоо хамаагүй зөвхөн тухайн баатрын оролцож ялагч болсон үеүдийг л тооцно гэдгийг анхаар.

Жишээ

$N=5$ баатартай үед, $N-1$ баатрууд чансаагаараа дараах байдлаар жагссан байгаа гэж үзье. $[1, 0, 2, 4]$. Мөн хоцорч ирсэн баатрын чансаа нь $R=3$. Тэмцээн нийт $C=3$ үе үргэлжилж дуусна. Тэмцээний үеүдийг тодорхойлох (S, E) гэсэн хоёр тоонууд нь $(1, 3), (0, 1), (0, 1)$ байгааг Леонардо мэдэж байгаа.

Хэрвээ Леонардо хоцорч ирсэн баатрыг жагсаалын эхний байрлалд зогсоовол нийт баатрууд чансаагаараа дараах байдлаар зогсоно [3, 1, 0, 2, 4]. Тэгэхээр тэмцээний нэгдүгээр үед (1,2,3 дугаар байрлалд байгаа) 1, 0, 2 чансаатай баатрууд оролцож 2 чансаатай баатар нь ялагч болно. Үүний дараа жагсаал [3, 2, 4] болно. Хоёрдугаар үед 3, 2 гэсэн чансаатай баатруудын хооронд тулаан явагдана. (0,1 гэсэн байрлалд байгаа баатрууд орно) Тулаанд 3 чансаатай баатар ялж жагсаалт [3, 4] болно. Тэмцээний сүүлчийн үе буюу гуравдугаар үед (0,1 байрлалд байгаа баатрууд орно) 4 чансаатай баатар ялна. Тиймээс хоцорч ирсэн баатрыг жагсаалын эхэнд байрлуулбал тэр зөвхөн 1 үед л ялалт байгуулж байна (хоёр дахь үед). Нэгдүгээр үеийн тулаанд тэрээр оролцоогүй учраас ялалт байгуулсанд тооцохгүй байгааг анхаар.

Үүний оронд Ленорда хоцорч ирсэн баатрыг 1, 0 чансаа бүхий баатруудын хооронд байрлуулбал тэмцээний жагсаалт дараах байдалтай болно [1, 3, 0, 2, 4]. Энэ тохиодолд нэгдүгээр үед 3, 0, 2 чансаатай баатрууд оролцоно. Нэгдүгээр үед $R = 3$ чансаатай баатар ялна. Үүний дараа жагсаал [1, 3, 4] болно. Тэмцээний хоёрдугаар үед 1, 3 чансаатай 2 баатар тулалдаж $R = 3$ чансаатай баатар дахин ялалт байгуулна. Ингээд хамгийн сүүлчийн үе буюу гуравдугаар үе эхлэхэд [3, 4] гэсэн жагсаал бий болсон байх бөгөөд 4 чансаатай баатар ялж тэмцээн дуусна. Энэ тохиолдолд хоцорч ирсэн $R = 3$ чансаатай баатар 2 удаа ялалт байгуулж байна. Энэ нь байж болох хамгийн сайн байрлуулалт юм. Учир нь тэр хамгийн ихдээ 2 удаа л ялах боломжтой.

Бодлогын даалгавар

Таны даалгавар бол хоцорч ирсэн баатрын тулалдаанд оролцож ялагчаар тодрох үеүдийн тоог хамгийн их байлгахын тулд түүнийг тэмцээн эхлэхээс өмнө аль байрлалд оруулахыг олох програм бичих юм. Тодорхой хэлбэл `GetBestPosition(N, C, R, K, S, E)` функцийг бичих хэрэгтэй.

- N нь баатруудын тоо;
- C нь тэмцээний үеүдийн тоо ($1 \leq C \leq N - 1$);
- R нь хоцорч ирсэн баатрын чансаа. Бүх баатрууд (жагссан байгаа болон хоцорч ирсэн баатар) $0, \dots, N - 1$ гэсэн ялгаатай чансаатай.
- K нь $N - 1$ ширхэг бүхэл тоо бүхий массив бөгөөд энэ нь хоцорсоноос бусад $N-1$ баатруудын жагсаал дахь чансааны дараалал юм;
- S, E нь C хэмжээ бүхий массив: тус бүрийнх нь i нь 0 -оос $C - 1$ хооронд байна. $(i + 1)$ дугаар үеийн тулаан нь $S[i]$ -ээс $E[i]$ дугаар байрлалд байгаа бүх баатруудын хооронд болно. Бүх i -ийн хувьд $S[i] < E[i]$ байна.

Энэ функцэд ирж байгаа утгууд нь үнэн зөв байна. Өөрөөр хэлбэл $E[i]$ нь $(i + 1)$ үед тулалдах баатруудын тооноос бага байна. Мөн C дахь үеийн тулааны дараа тэмцээнд зөвхөн нэг ялагч үлдэнэ.

`GetBestPosition(N, C, R, K, S, E)` функц хамгийн сайн байрлал болох $P(0 \leq P \leq N-1)$ -г буцаах ёстой. Хэрвээ хэд хэдэн сайн байрлал байвал хамгийн багыг нь буцаана. P байрлал нь 0 -ээс эхэлж дугаарлагдсан байна. Өөрөөр хэлбэл P нь хоцорч ирсэн баатрын өмнө

зогсох баатруудын тоо. Жишээ нь хэрвээ $P=0$ бол хоцорч ирсэн баатар жагсаалын эхэнд байрлана гэсэн үг. Мөн хэрвээ $P = N-1$ бол жагсаалын төгсгөлд байрлана гэсэн үг.

Дэд даалгавар 1 [17 оноо]

$N \leq 500$ байна.

Дэд даалгавар 2 [32 оноо]

$N \leq 5\,000$ байна.

Дэд даалгавар 3 [51 оноо]

$N \leq 100\,000$ байна.

Хэрэгжүүлэлтийн деталь

`tournament.c`, `tournament.cpp` эсвэл `tournament.pas` нэртэй яг нэг файл илгээх ёстой. Энэ файл нь дээр заасан функцыг тодорхойлсон байх ёстой.

C/C++ прогорам

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

Pascal програм

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

Эдгээр функц нь дээр заасны дагуу ажиллах ёстой. Мэдээж өөрт шаардлагатай бусад функцуудыг тодорхойлж бичиж болно. Гэхдээ таны бичсэн програм ямар нэг байдлаар стандарт оролт гаралт болон бусад файльтай харьцах ёсгүй.

Жишээ graders

Жишээ grader-т оролтыг дараах байдлаар авна.

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

Хугацаа болон санах ойн хязгаарлалт

- Хугацааны хязгаарлалт: 1 second.
- Санах ойн хязгаарлалт: 256 MiB.