



Մեծ մրցանակ

«Մեծ մրցանակ»-ը հայտնի հեռուստատեսային շոու է: Դուք մրցույթի երջանիկ մասնակից եք և հայտնվել եք եզրափակիչ փուլում: Դուք կանգնած եք n տուփերի շարքի առաջ, որոնք ձախից աջ համարակալված են 0 -ից մինչև $n - 1$: Յուրաքանչյուր տուփ պարունակում է մրցանակ, որը հնարավոր չէ տեսնել մինչև արկղի բացելը: Գոյություն ունեն $v \geq 2$ տարբեր տեսակի մրցանակներ: Տեսակները համարակալված են 1 -ից մինչև v . արժեքների նվազման կարգով:

1 -ին տեսակի մրցանակը ամենաթանկն է. այն ադամանդ է: Բոլոր տուփերում կա ընդամենը մեկ ադամանդ: v -րդ տեսակի մրցանակը ամենաէժանն է. այն սառնաշաքար է: Խաղն ավելի գրավիչ դարձնելու նպատակով էժան մրցանակների քանակը մեծ է քան թանկ մրցանակների քանակը: Ավելի ստույգ, բոլոր t -երի համար, $2 \leq t \leq v$, մենք գիտենք հետևյալը. Եթե $(t - 1)$ -րդ տեսակի մրցանակները k հատ են, ապա t -րդ տեսակի մրցանակների քանակը խիստ մեծ է k^2 -ից:

Ձեր նպատակն է նվաճել ադամանդը: Խաղի վերջում դուք բացելու եք տուփը և ստանալու եք այնտեղ պահվող մրցանակը: Նախքան տուփն ընտրելը դուք կարող եք խաղավար Ռամբոդին հարցեր տալ: Յուրաքանչյուր հարցի համար դուք ընտրում եք որևէ i -րդ տուփ: Որպես պատասխան Ռամբոդը ձեզ տալու է երկու ամբողջ թիվ պարունակող a զանգված: Դրանց իմաստը հետևյալն է.

- i -րդ տուփից ձախ գտնվող տուփերից ճիշտ $a[0]$ հատը պարունակում են i -րդ տուփի մրցանակից ավելի թանկ մրցանակներ:
- i -րդ տուփից աջ գտնվող տուփերից ճիշտ $a[1]$ հատը պարունակում են i -րդ տուփի մրցանակից ավելի թանկ մրցանակներ:

Օրինակ, ենթադրենք $n = 8$: Ձեր հարցի համար դուք ընտրում եք $i = 2$ արկղը: Որպես պատասխան Ռամբոդը ձեզ ասում է $a = [1, 2]$: Նշանակում է.

- 0 -րդ և 1 -ին տուփերից ճիշտ մեկը պարունակում է 2 -րդ տուփի մրցանակից ավելի թանկ մրցանակ:
- $3, 4, \dots, 7$ տուփերից ճիշտ երկուսը պարունակում են 2 -րդ տուփի մրցանակից ավելի թանկ մրցանակ: Ձեր խնդիրն է քիչ հարցեր տալով որոշել, թե որ տուփում է պահվում ադամանդը:

Իրականացման մանրամասներ

Դուք պետք է իրականացնեք հետևյալ ֆունկցիան.

```
int find_best(int n)
```

- n -ը տուփերի քանակն է:
- Այս ֆունկցիան պետք է վերադարձնի այն տուփի համարը, որտեղ պահվում է ադամանդը, այսինքն մեկ d ($0 \leq d \leq n - 1$) ամբողջ թիվ այնպես, որ d -րդ տուփը պարունակում է 1-ին տեսակի մրցանակը:

Վերևի ֆունկցիայում կարելի է կանչել հետևյալ ֆունկցիան.

```
int[] ask(int i)
```

- i -ն այն տուփի համարն է, որը դուք ընտրել եք հարցի համար: i -ի արժեքը պետք է ընկած լինի 0-ից մինչև $n - 1$ հատվածում, ներառյալ ծայրակետերը:
- Այս ֆունկցիան վերադարձնում է a զանգված, որն ունի 2 տարր: Այստեղ $a[0]$ -ն i -րդ տուփից ձախ գտնվող տուփերում ավելի թանկ մրցանակների քանակն է, իսկ $a[1]$ -ը i -րդ տուփից աջ գտնվող տուփերում ավելի թանկ մրցանակների քանակն է:

Օրինակ

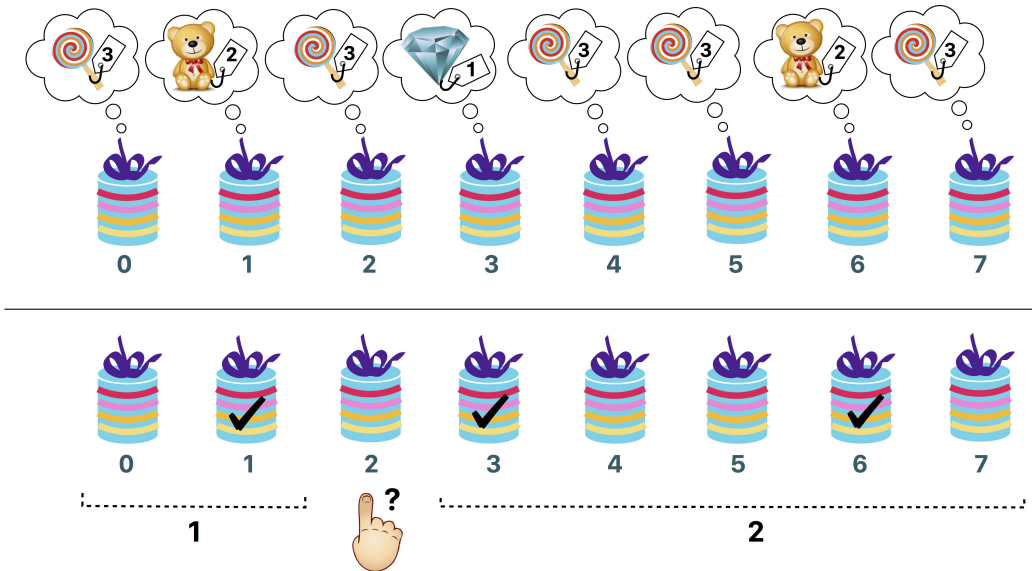
Գրեյդերն անում է հետևյալ ֆունկցիայի կանչը.

```
find_best(8)
```

Կա $n = 8$ տուփ: Ենթադրենք նրանցում պահվող մրցանակների տեսակներն են $[3, 2, 3, 1, 3, 3, 2, 3]$: Ստորև տրված են `ask` ֆունկցիայի բոլոր հնարավոր կանչերը և համապատասխան վերադարձի արժեքները:

- `ask(0)` վերադարձնում է $[0, 3]$
- `ask(1)` վերադարձնում է $[0, 1]$
- `ask(2)` վերադարձնում է $[1, 2]$
- `ask(3)` վերադարձնում է $[0, 0]$
- `ask(4)` վերադարձնում է $[2, 1]$
- `ask(5)` վերադարձնում է $[2, 1]$
- `ask(6)` վերադարձնում է $[1, 0]$
- `ask(7)` վերադարձնում է $[3, 0]$

Այս օրինակում ադամանդը գտնվում է 3-րդ արկղում: Հետևաբար `find_best` ֆունկցիան պետք է վերադարձնի 3:



Վերևի նկարը ցուցադրում է այս օրինակը: Վերևի մասում պատկերված են տուփերը և նրանցում գտնվող մրցանակների արժեքները: Ներքևի մասում ցուցադրված է `ask(2)` հարցումը: 2-րդ տուփի մրցանակից ավելի թանկ մրցանակ պարունակող տուփերի վրա նշիկ է դրված:

Սահմանափակումներ

- $3 \leq n \leq 200\,000$.
- Յուրաքանչյուր տուփում մրցանակի տեսակը 1-ից v հատվածին պատկանող թիվ է, ներառյալ ծայրակետերը:
- Կա 1 տեսակի ճիշտ մեկ մրցանակ:
- Եթե կա $t - 1$ տեսակի k մրցանակ, ապա t տեսակի մրցանակների քանակը *խիստ* մեծ է k^2 -ից, $2 \leq t \leq v$:

Ենթախնդիրներ և գնահատում

Որոշ թեստերում գրեյդերի պահվածքը հարմարվողական է: Դա նշանակում է, որ այդ թեստերում գրեյդերը չունի մրցանակների ֆիքսված հաջորդականություն: Փոխարենը գրեյդերի տված պատասխանները կարող են կախված լինել ձեր լուծման կողմից տրված հարցերից: Երաշխավորվում է, որ գրեյդերը պատասխանում է այնպես, որ յուրաքանչյուր պատասխանից հետո կա մրցանակների առնվազն մեկ հաջորդականություն, որը համատեղելի է մինչ այդ տրված բոլոր պատասխաններին:

1. (20 միավոր) Կա ճիշտ 1 ադամանդ և $n - 1$ սառնաշաքար (այսինքն, $v = 2$): Դուք կարող եք կանչել `ask` ֆունկցիան առավելագույնը 10 000 անգամ:
2. (80 միավոր) Լրացուցիչ սահմանափակումներ չկան:

Ենթախնդիր 2-ում դուք կարող եք մասնակի միավոր ստանալ: Դիցուք q -ն այս ենթախնդրի բոլոր թեստերից `ask` ֆունկցիայի ամենաշատ կանչերի քանակն է: Այդ դեպքում այս ենթախնդրի համար ձեր միավորը հաշվվում է ըստ հետևյալ

աղյուսակի.

Հարցեր	Միավոր
$10\,000 < q$	0 (CMS-ում հայտարարվում է որպես 'Wrong Answer')
$6000 < q \leq 10\,000$	70
$5000 < q \leq 6000$	$80 - (q - 5000)/100$
$q \leq 5000$	80

Գրեյդերի օրինակ

Գրեյդերի օրինակը հարմարվողական չէ: Փոխարենը այն կարդում և օգտագործում է մրցանակների տիպերի ֆիքսված p զանգված: b -րդ տուփում մրցանակի տեսակը $p[b]$ է, $0 \leq b \leq n - 1$: Գրեյդերի օրինակը մուտքային տվյալները կարդում է հետևյալ ձևաչափով.

- տող 1. n
- տող 2. $p[0] \ p[1] \ \dots \ p[n - 1]$

Գրեյդերի օրինակը տպում է մեկ տող, որը պարունակում է `find_best`-ի վերադարձի արժեքը և `ask` ֆունկցիայի կանչերի քանակը: