



Najvzácnejší chrobáčisko

Po dome Uja Blangkona behá N chrobákov, očíslovaných od 0 po $N - 1$. Každý chrobák má svoj **typ** - celé číslo medzi 0 a 10^9 , vrátane. Viacero chrobákov môže zdieľať rovnaký typ.

Majme chrobákov poskupinkovaných podľa ich typu. Definujme kardinalitu **najčastejšieho** typu chrobákov ako počet chrobákov v skupine chrobákov, ktorá má najviac chrobákov. Nápodobne, kardinalita **najvzácnejšieho** typu chrobákov je počet chrobákov v skupine chrobákov, ktorá má najmenší počet chrobákov.

Zoberme si následovných 11 chrobákov s typmi $[5, 7, 9, 11, 11, 5, 0, 11, 9, 100, 9]$. V tejto kôpke sú skupinky s najväčším počtom chrobákov skupinky typov 9 a 11 - každá z nich obsahuje 3 chrobáky. Teda v tomto prípade je kardinalita **najčastejšieho** typu chrobákov 3. Nápodobne, skupinky s najmenším počtom chrobákov sú skupinky typov 7, 0 a 100 - každá z nich obsahuje len 1 chrobáka.

Ujo Blangkon ale nevie rozoznať jednotlivé typy chrobákov, má veľmi zlý zrak. Má ale aj Chrobákožrút 3000. S jeho pomocou vie stlačením jediného tlačidla zistiť kardinalitu **najčastejšieho** typu chrobákov vo vnútri Chrobákožrúta. Chrobákožrút podporuje tri operácie:

1. Vcucnutie chrobáka
2. Vypľutie chrobáka
3. Stlačenie tlačidla

Každá operácia môže byť použitá najviac 40 000 krát.

Vašou úlohou je za pomoci Chrobákožrúta zistiť kardinalitu **najvzácnejšieho** typu chrobákov naprieč všetkými N chrobákmi nachádzajúcimi sa v dome Uja Blangkona. Aby toho nebolo málo, v niektorých podúlohách Vaše skóre závisí na maximálnom počte vykonaných operácií jednotlivých typov (viď obmedzenia podúloh).

Implementačné Detaily

Implementujte nasledovnú funkciu:

```
int min_cardinality(int N)
```

- N : počet chrobákov.
- Táto funkcia by mala navrátiť kardinalitu **najvzácnejšieho** chrobáka.
- Táto funkcia bude zavolaná práve raz.

Vyššie popísaná funkcia môže volať aj nasledovné funkcie:

```
void move_inside(int i)
```

- i : index chrobáka, ktorého vcucne Chrobákožrút. Hodnota i musí byť medzi 0 a $N - 1$, vrátane.
- Ak sa tento chrobák už nachádza v Chrobákožrútovi, volanie tejto funkcie nezmení obsah Chrobákožrútovhovho vrecka. Napriek tomu sa toto volanie stále počíta k celkovému počtu volaní.
- Táto funkcia môže byť zavolaná maximálne 40 000 krát.

```
void move_outside(int i)
```

- i : index chrobáka, ktorý má byť vyplutý z Chrobákožrúta. Hodnota i musí byť medzi 0 a $N - 1$, vrátane.
- Ak sa tento chrobák nenachádza v Chrobákožrútovi, volanie tejto funkcie nezmení obsah Chrobákožrútovhovho vrecka. Napriek tomu sa toto volanie stále počíta k celkovému počtu volaní.
- Táto funkcia môže byť zavolaná maximálne 40 000 krát.

```
int press_button()
```

- Táto funkcia vracia kardinalitu **najčastejšieho** typu chrobákov, ktoré sa v tejto chvíli nachádzajú vo vrecku Chrobákožrúta.
- Táto funkcia môže byť zavolaná maximálne 40 000 krát.

Príklad

Predstavme si prípad so 6 chrobákmi typov [5, 8, 9, 5, 9, 9]. Zavoláme:

```
min_cardinality(6)
```

Táto funkcia by mohla postupne volať funkcie `move_inside`, `move_outside`, a `press_button` nasledovným spôsobom:

Volanie	Návratová hodnota	Chrobáky vo vrečku	Typy chrobákov vo vrečku
		{}	[]
move_inside(0)		{0}	[5]
press_button()	1	{0}	[5]
move_inside(1)		{0,1}	[5,8]
press_button()	1	{0,1}	[5,8]
move_inside(3)		{0,1,3}	[5,8,5]
press_button()	2	{0,1,3}	[5,8,5]
move_inside(2)		{0,1,2,3}	[5,8,9,5]
move_inside(4)		{0,1,2,3,4}	[5,8,9,5,9]
move_inside(5)		{0,1,2,3,4,5}	[5,8,9,5,9,9]
press_button()	3	{0,1,2,3,4,5}	[5,8,9,5,9,9]
move_inside(5)		{0,1,2,3,4,5}	[5,8,9,5,9,9]
press_button()	3	{0,1,2,3,4,5}	[5,8,9,5,9,9]
move_outside(5)		{0,1,2,3,4}	[5,8,9,5,9]
press_button()	2	{0,1,2,3,4}	[5,8,9,5,9]

V tomto bode máme dost' informácií na odvodenie faktu, že kardinalita najvzácnejšieho chrobáčieho typu je 1. Teda volanie `min_cardinality(6)` vráti hodnotu 1.

V tomto príklade sme zavolali funkciu `move_inside` 7-krát, `move_outside` 1-krát a `press_button` 6-krát.

Obmedzenia

- $2 \leq N \leq 2000$

Podúlohy

1. (10 bodov) $N \leq 200$
2. (15 bodov) $N \leq 1000$
3. (75 bodov) Žiadne ďalšie obmedzenia.

Ak v ktoromkoľvek testovacom vstupe počet volaní funkcií `move_inside`, `move_outside` alebo `press_button` presiahne povolené limity, alebo návratová hodnota `min_cardinality` bude

nesprávna, skóre vášho riešenia pre danú podúlohu bude 0.

V podúlohe 3 navyše platí, že niektoré riešenia dostanú len čiastočný počet bodov. Presný počet bodov, ktorý za túto podúlohu vaše riešenie dostane, je určený nasledovne:

Nech q je **maximum** z počtov volaní funkcií `move_inside`, `move_outside` a `press_button`.

Ďalej, nech m je maximálna hodnota $\frac{q}{N}$ naprieč všetkými testovacími príkladmi v tejto podúlohe.

Body za podúlohu 3 budú priradené na základe nasledovnej tabuľky:

Obmedzenia	Body
$20 < m$	0 (reportované ako "Output isn't correct" v CMS)
$6 < m \leq 20$	$\frac{225}{m-2}$
$3 < m \leq 6$	$81 - \frac{2}{3}m^2$
$m \leq 3$	75

Lokálny testovač

Nech T je pole N celých čísel, pričom číslo $T[i]$ predstavuje typ chrobáka číslo i .

Lokálny testovač číta vstup v nasledujúcom formáte

- riadok 1: N
- riadok 2: $T[0] T[1] \dots T[N - 1]$

Ak lokálny testovač detekuje porušenie protokolu, výstup bude `Protocol Violation: <MSG>`, kde `<MSG>` bude jedno z nasledovných:

- `invalid parameter`: parameter funkcií `move_inside(x)` alebo `move_outside(x)` bola neplatná hodnota, teda x nebolo medzi 0 a $N - 1$, vrátane.
- `too many calls`: počet volaní *niektorej* z funkcií `move_inside`, `move_outside` alebo `press_button` prekročil 40 000.

Inak bude výstup lokálneho testovača v nasledovnom formáte:

- riadok 1: návratová hodnota z `min_cardinality`
- riadok 2: q