



Ең сирек жәндіктер

Пак Блангконның үйінің айналасында жүгіріп жүрген 0 -ден $N - 1$ -ге дейін нөмірленген N жәндік бар.

Әр жәндіктің 0 мен 10^9 арасындағы бүтін сан болатын **түрі** бар. Бірнеше жәндіктің түрі бірдей болуы мүмкін.

Жәндіктер түрлері бойынша топтарға бөлінген дейік. Біз **ең жиі** жәндік түрінің кардиналдығы деп ең көп жәндіктері бар топтың ішіндегі жәндіктердің санын айтамыз. Ұқсас түрде, **ең сирек** жәндік түрінің кардиналдығы деп ең аз жәндіктері бар топтың ішіндегі жәндіктердің санын айтамыз.

Мысалға, 11 жәндік бар дейік. Олардың түрлері $[5, 7, 9, 11, 11, 5, 0, 11, 9, 100, 9]$. Осы жағдайда, **ең жиі** жәндік түрінің кардиналдығы 3 . Ең көп жәндігі бар топтар - 9 және 11 түрлері. Әрқайсысында 3 жәндік бар. **Ең сирек** жәндік түрінің кардиналдығы 1 . Ең аз жәндігі бар топтар - $7, 0$ және 100 түрлері. Әрқайсысында 1 жәндік бар.

Пак Блангкон ешқандай жәндіктің түрін білмейді. Оның жалғыз ғана батырмасы бар машинасы бар. Батырма жәндіктердің түрі жөнінде біраз ақпарат бере алады. Бастапқыда, машина бос. Машинаны қолдану үшін, үш түрлі операцияны жасауға болады:

1. Жәндікті машинаның ішіне кіргізу.
2. Жәндікті машинаның ішінен шығару,
3. Машинаның батырмасын басу.

Кез-келген операцияның түрін ең көп дегенде 40000 рет қолдануға болады.

Батырма басылған кезде, машина өзінің ішіндегі жәндіктерді ғана қарастырған кездегі **ең жиі** жәндік түрінің кардиналдығын баяндайды.

Сіздің тапсырмаңыз - **ең сирек** жәндік түрінің кардиналдығын анықтау. Сонымен қатар, кейбір бөлімшелерде сіздің ұпайыңыз белгілі бір түрдің ең көп қолданылған операция санына байланысты (көбірек ақпарат үшін Бөлімшелер секциясын көрініз).

Іске асыру мәліметтері

Сізге келесі функцияны іске асыру керек:

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

- N : жәндіктердің саны.
- Бұл функция Пак Блангонның үйіндегі бүкіл жәндіктердің арасында **ең сирек** жәндік түрінің кардиналдығын қайтару керек.
- Бұл функция тура бір рет шақырылады.

Үстінгі функция келесі функцияларды шақыра алады:

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

- i : машинаның ішіне кіргізетін жәндіктің нөмірі. i мәні 0 және $N - 1$ (қоса алғанда) арасында болу керек.
- Егер бұл жәндік машинаның ішінде болған болса, бұл шақырудың машинаның ішіндегі жәндіктерге ешқандай әсері болмайды. Бірақ, бұл бәрібір жеке шақыру ретінде саналады.
- Бұл функция ең көп дегенде $40\,000$ рет қана шақырыла алады.

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

- i : машинаның ішінен шығаратын жәндіктің нөмірі. i мәні 0 және $N - 1$ (қоса алғанда) арасында болу керек.
- Егер бұл жәндік машинаның сыртында болған болса, бұл шақырудың машинаның ішіндегі жәндіктерге ешқандай әсері болмайды. Бірақ, бұл бәрібір жеке шақыру ретінде саналады.
- Бұл функция ең көп дегенде $40\,000$ рет қана шақырыла алады.

```
int press_button()
```

- Бұл функция машинаның ішіндегі жәндіктерді ғана қарастырған кездегі **ең жиі** жәндік түрінің кардиналдығын баяндайды.
- Бұл функция ең көп дегенде $40\,000$ рет қана шақырыла алады.
- Бағалаушы **адаптивті емес**. Демек, бүкіл N жәндіктің түрі `min_cardinality` шақырылуына дейін белгіленген.

Мысал

Түрлері $[5, 8, 9, 5, 9, 9]$ болатын 6 жәндік бар жағдайды қарастырайық. `min_cardinality` функция келесі жолмен шақырылады:

```
min_cardinality(6)
```

Функция келесі жолмен `move_inside`, `move_outside` және `press_button` функцияларын шақыра алады.

Шақыру	Қайтару мәні	Машина ішіндегі жәндіктер	Машина ішіндегі жәндіктердің түрлері
		{}	[]
<code>move_inside(0)</code>		{0}	[5]
<code>press_button()</code>	1	{0}	[5]
<code>move_inside(1)</code>		{0, 1}	[5, 8]
<code>press_button()</code>	1	{0, 1}	[5, 8]
<code>move_inside(3)</code>		{0, 1, 3}	[5, 8, 5]
<code>press_button()</code>	2	{0, 1, 3}	[5, 8, 5]
<code>move_inside(2)</code>		{0, 1, 2, 3}	[5, 8, 9, 5]
<code>move_inside(4)</code>		{0, 1, 2, 3, 4}	[5, 8, 9, 5, 9]
<code>move_inside(5)</code>		{0, 1, 2, 3, 4, 5}	[5, 8, 9, 5, 9, 9]
<code>press_button()</code>	3	{0, 1, 2, 3, 4, 5}	[5, 8, 9, 5, 9, 9]
<code>move_inside(5)</code>		{0, 1, 2, 3, 4, 5}	[5, 8, 9, 5, 9, 9]
<code>press_button()</code>	3	{0, 1, 2, 3, 4, 5}	[5, 8, 9, 5, 9, 9]
<code>move_outside(5)</code>		{0, 1, 2, 3, 4}	[5, 8, 9, 5, 9]
<code>press_button()</code>	2	{0, 1, 2, 3, 4}	[5, 8, 9, 5, 9]

Осы кезде, ең сирек жәндік түрінің кардиналдығы 1 екенін анықтау үшін жеткілікті мәлімет бар. Сондықтан, `min_cardinality` функциясы 1 қайтару керек.

Бұл мысалда, `move_inside` 7 рет, `move_outside` 1 рет, және `press_button` 6 рет шақырылды.

Шектеулер

- $2 \leq N \leq 2000$

Бөлімшелер

1. (10 ұпай) $N \leq 200$
2. (15 ұпай) $N \leq 1000$
3. (75 ұпай) Қосымша шектеулер жоқ.

Егер тестілік жағдайдың біреуінде `move_inside`, `move_outside`, немесе `press_button` функцияларының шақырулары Іске асыру мәліметтері секциясында баяндалған шектеулерге бағынбаса, немесе `min_cardinality`-дың қайтарылған мәні қате болса, сол бөлімше үшін сіздің шешіміздің ұпайы 0 болады.

q деп келесі үш мәнің максимумын белгілейік: `move_inside` шақырулар саны, `move_outside` шақырулар саны, және `press_button` шақырулар саны.

Үшінші бөлімшеде, сіз жартылай ұпай ала аласыз.

m деп осы бөлімшедегі бүкіл тестілік жағдайлардағы ең жоғары $\frac{q}{N}$ мәнін белгілейік. Сіздің осы бөлімшедегі ұпайыңыз келесі кесте арқылы саналады:

Шарт	Ұпайлар
$20 < m$	0 (CMS-те "Output isn't correct" деп белгіленеді)
$6 < m \leq 20$	$\frac{225}{m-2}$
$3 < m \leq 6$	$81 - \frac{2}{3}m^2$
$m \leq 3$	75

Үлгі бағалаушы

T деген N сандардың жиыны болсын. $T[i]$ - i -інші жәндіктің түрі.

Үлгі бағалаушы енгізбені келесі форматта оқиды:

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

Егер үлгі бағалаушы протокол бұзылуын байқаса, үлгі бағалаушының шықпасы `Protocol Violation: <MSG>` болады. Мұнда `<MSG>` келесілердің біреуі болады:

- `invalid parameter`: `move_inside` немесе `move_outside` шақыруында, i мәні 0 және $N - 1$ (қоса алғанда) арасында емес.
- `too many calls`: `move_inside`, `move_outside`, немесе `press_button` функцияларының кез-келгенінің шақыру саны 40 000-нан асып кетті.

Басқа жағдайда, үлгі бағалаушының шықпасы келесі форматта болады:

- жол 1: `min_cardinality` қайтару мәні
- жол 2: q