



Los insectos más raros

Hay N insectos enumerados del 0 al $N - 1$ corriendo alrededor de la casa de Pak Blangkon. Cada insecto tiene un **tipo**, el cual es un entero entre 0 y 10^9 (inclusivo). Múltiples insectos pueden tener el mismo tipo.

Supongamos que los insectos son agrupados por tipo. Definimos la cardinalidad del tipo de insecto **más frecuente** como la cantidad de insectos que tenga el grupo más grande. De manera similar, la cardinalidad del tipo de insecto **más raro** es la cantidad de insectos que tenga el grupo más pequeño.

Por ejemplo, supongamos que hay 11 insectos cuyos tipos son $[5, 7, 9, 11, 11, 5, 0, 11, 9, 100, 9]$. En este caso, la cardinalidad del tipo de insecto **más frecuente** es 3. Los tipos con la mayor cantidad de insectos son el 9 y el 11, cada uno con 3 insectos. La cardinalidad del tipo de insecto **más raro** es 1. Los tipos con la menor cantidad de insectos son el 7, el 0 y el 100, cada uno con 1 insecto.

Pak Blangkon no sabe el tipo de ningún insecto. Él tiene una máquina con un solo botón que puede darle cierta información sobre los tipos de los insectos. Inicialmente la máquina está vacía. Para usar la máquina se pueden ejecutar tres tipos de operación.

1. Meter un insecto dentro de la máquina
2. Sacar un insecto afuera de la máquina
3. Presionar el botón de la máquina

Cada operación puede ser ejecutada a lo mucho 40 000 veces.

Cada vez que el botón es presionado, la máquina reporta la cardinalidad del tipo de insecto **más frecuente** considerando sólo los insectos dentro de ella.

Su tarea es determinar la cardinalidad del tipo de insecto **más raro** entre los N insectos de la casa de Pak Blangkon usando la máquina. Adicionalmente, en algunas subtareas su puntaje dependerá de la máxima cantidad de operaciones de un determinado tipo que sean ejecutadas (vea la sección Subtareas para más detalles).

Detalles de Implementación

Su tarea es implementar el siguiente procedimiento:

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

- N : La cantidad de insectos
- Este procedimiento debe devolver la cardinalidad del tipo de insecto **más raro** entre los N insectos de la casa de Pak Blangkon.
- Este procedimiento es llamado exactamente una vez.

El procedimiento anterior puede hacer llamadas a los siguientes procedimientos:

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

- i : El índice del insecto a ser metido a la máquina. El valor de i debe estar entre 0 y $N - 1$ (inclusivo).
- Si este insecto ya está dentro de la máquina, la llamada no tiene efecto en el conjunto de insectos de la máquina; sin embargo, aún será contada como una llamada ejecutada.
- Este procedimiento puede ser llamado a lo mucho 40 000 veces.

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

- i : El índice del insecto a ser sacado de la máquina. El valor de i debe estar entre 0 y $N - 1$ (inclusivo).
- Si este insecto ya está afuera de la máquina, la llamada no tiene efecto en el conjunto de insectos de la máquina; sin embargo, aún será contada como una llamada ejecutada.
- Este procedimiento puede ser llamado a lo mucho 40 000 veces.

```
int press_button()
```

- Este procedimiento devuelve la cardinalidad del tipo de insecto **más frecuente** considerando solo los insectos dentro de la máquina.
- Este procedimiento puede ser llamado a lo mucho 40 000 veces.
- El evaluador **no es adaptativo**. Esto significa que los tipos de los N insectos son fijados antes de que se llame a `min_cardinality`.

Ejemplo

Considere un escenario en el que hay 6 insectos de tipos $[5, 8, 9, 5, 9, 9]$ respectivamente. El procedimiento `min_cardinality` es llamado de la siguiente manera:

```
min_cardinality(6)
```

El procedimiento puede llamar a `move_inside`, `move_outside`, y `press_button` como se muestra a continuación.

Llamada	Valor devuelto	Insectos en la máquina	Tipos de insectos en la máquina
		{}	[]
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]

En este punto hay suficiente información para concluir que la cardinalidad del tipo de insecto más raro es de 1. Por lo tanto, el procedimiento `min_cardinality` debe devolver 1.

En este ejemplo, `move_inside` es llamado 7 veces, `move_outside` es llamado 1 vez, y `press_button` es llamado 6 veces.

Restricciones

- $2 \leq N \leq 2000$

Subtareas

1. (10 puntos) $N \leq 200$
2. (15 puntos) $N \leq 1000$
3. (75 puntos) Sin restricciones adicionales.

Si en alguno de los casos de prueba las llamadas a los procedimientos `move_inside`, `move_outside`, o `press_button` no cumplen con las restricciones descritas en los Detalles de

Implementación o el valor devuelto por `min_cardinality` es incorrecto, el puntaje obtenido por su solución en esa subtarea será de 0.

Sea q el **máximo** de los siguientes tres valores: la cantidad de llamadas a `move_inside`, la cantidad de llamadas a `move_outside`, y la cantidad de llamadas a `press_button`.

En la subtarea 3 puede obtener puntaje parcial. Sea m el máximo valor de $\frac{q}{N}$ entre todos los casos de prueba de esta subtarea. Su puntaje para esta subtarea es calculado de acuerdo a la siguiente tabla:

Condición	Puntos
$20 < m$	0 (reportado como "Output isn't correct" en CMS)
$6 < m \leq 20$	$\frac{225}{m-2}$
$3 < m \leq 6$	$81 - \frac{2}{3}m^2$
$m \leq 3$	75

Evaluador de prueba

Sea T un arreglo de N enteros donde $T[i]$ es el tipo del insecto i .

El evaluador de prueba lee la entrada con el siguiente formato:

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

Si el evaluador de prueba detecta una violación del protocolo, la salida del evaluador es `Protocol Violation: <MSG>`, donde `<MSG>` es uno de los siguientes:

- `invalid parameter`: en alguna llamada de `move_inside` o `move_outside`, el valor de i no está entre 0 y $N - 1$ (inclusivo).
- `too many calls`: la cantidad de llamadas de **alguno** de `move_inside`, `move_outside`, o `press_button` excede a 40 000.

En caso contrario, la salida del evaluador de prueba está en el siguiente formato:

- línea 1: el valor retornado por `min_cardinality`
- línea 2: q