



Seltenste Insekten

In Pak Blangkons Haus laufen N Insekten mit Indizes von 0 bis $N - 1$ herum. Jedes Insekt hat eine **Art**, welche durch eine ganze Zahl zwischen 0 und 10^9 angegeben ist. Mehrere Insekten können von derselben Art sein.

Nehmen wir an, dass die Insekten nach Art gruppiert sind. Die Kardinalität (Stückzahl) der **häufigsten** Insektenart definieren wir als die Anzahl der Insekten in einer Gruppe mit der größten Anzahl an Insekten. Entsprechend ist die Kardinalität der **seltensten** Insektenart die Anzahl an Insekten in einer Gruppe mit der kleinsten Anzahl an Insekten.

Nehmen wir zum Beispiel an, dass es 11 Insekten mit Arten $[5, 7, 9, 11, 11, 5, 0, 11, 9, 100, 9]$ gibt. In diesem Fall ist die Kardinalität der **häufigsten** Insektenart 3. Die Gruppen mit der größten Anzahl an Insekten sind Art 9 und Art 11, die jeweils 3 Insekten enthalten. Die Kardinalität der **seltensten** Insektenart ist 1. Die Gruppen mit der kleinsten Anzahl an Insekten sind Art 7, Art 0 und Art 100, die aus je 1 Insekt bestehen.

Pak Blangkon kennt die Art keines der Insekten. Er hat eine Maschine mit einem einzigen Knopf, die Information über die Arten eines Teils der Insekten geben kann. Anfangs ist die Maschine leer. Beim Benutzen der Maschine können drei Typen von Handlungen ausgeführt werden:

1. Lege ein Insekt in die Maschine.
2. Nimm ein Insekt aus der Maschine.
3. Drücke den Knopf auf der Maschine.

Jeder Typ von Handlung kann höchstens 40 000 Mal ausgeführt werden.

Jedes Mal, wenn der Knopf gedrückt wird, teilt die Maschine die Kardinalität der **häufigsten** Insektenart mit, aber nur unter den Insekten in der Maschine.

Deine Aufgabe ist es, die Kardinalität der **seltensten** Insektenart unter allen N Insekten in Pak Blangkons Haus mit Hilfe der Maschine zu bestimmen. In manchen Teilaufgaben hängt deine Punktzahl zudem von der maximalen Anzahl der durchgeführten Handlungen je Typ ab (siehe Abschnitt Teilaufgaben für Details).

Implementierungsdetails

Du sollst die folgende Funktion implementieren:

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

- N : die Anzahl an Insekten.
- Diese Funktion soll die Kardinalität der **seltensten** Insektenart unter allen N Insekten in Pak Blangkons Haus zurückgeben.
- Diese Funktion wird genau einmal aufgerufen.

Die obige Funktion kann die folgenden Funktionen aufrufen:

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

- i : der Index des Insekts, das in die Maschine gelegt werden soll. Der Wert von i muss zwischen 0 und $N - 1$ (inklusive) liegen.
- Falls dieses Insekt bereits in der Maschine liegt, hat der Aufruf keine Auswirkung auf die Menge der Insekten in der Maschine. Dennoch wird er als Aufruf gezählt.
- Diese Funktion kann höchstens 40 000 Mal aufgerufen werden.

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

- i : der Index des Insekts, das aus der Maschine genommen werden soll. Der Wert von i muss zwischen 0 und $N - 1$ (inklusive) liegen.
- Falls dieses Insekt bisher nicht in der Maschine liegt, hat der Aufruf keine Auswirkung auf die Menge der Insekten in der Maschine. Dennoch wird er als Aufruf gezählt.
- Diese Funktion kann höchstens 40 000 Mal aufgerufen werden.

```
int press_button()
```

- Diese Funktion gibt die Kardinalität der **häufigsten** Insektenart zurück, wobei nur die Insekten in der Maschine einbezogen werden.
- Diese Funktion kann höchstens 40 000 Mal aufgerufen werden.
- Der Grader ist **nicht adaptiv**. Das bedeutet, dass die Arten aller N Insekten vor dem Aufruf an `min_cardinality` schon feststehen.

Beispiel

Betrachte ein Szenario, in dem es 6 Insekten der Arten `[5,8,9,5,9,9]` gibt. Die Funktion `min_cardinality` wird folgendermaßen aufgerufen:

```
min_cardinality(6)
```

Die Funktion könnte dann die Funktionen `move_inside`, `move_outside` und `press_button` wie folgt aufrufen:

Aufruf	Rückgabewert	Insekten in der Maschine	Insektenarten in der Maschine
		{}	[]
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]

Am Ende wurden genügend Informationen gesammelt, um herauszufinden, dass die Kardinalität der seltensten Insektenart 1 ist. Also sollte die Funktion `min_cardinality` die Zahl 1 zurückgeben.

In diesem Beispiel wurde `move_inside` 7 Mal aufgerufen, `move_outside` wurde 1 Mal aufgerufen und `press_button` wurde 6 Mal aufgerufen.

Beschränkungen

- $2 \leq N \leq 2000$

Teilaufgaben

1. (10 Punkte) $N \leq 200$
2. (15 Punkte) $N \leq 1000$
3. (75 Punkte) Keine weiteren Beschränkungen.

Wenn die Aufrufe an `move_inside`, `move_outside` oder `press_button` in einem der Testfälle nicht den Bedingungen aus den Implementierungsdetails entsprechen oder der Rückgabewert von `min_cardinality` `false` ist, wird die entsprechende Teilaufgabe mit 0 Punkten bewertet.

Sei q das **Maximum** der folgenden drei Werte: Die Anzahl an Aufrufen von `move_inside`, die Anzahl an Aufrufen von `move_outside` und die Anzahl an Aufrufen von `press_button`.

In Teilaufgabe 3 kannst du Teilpunkte bekommen. Sei m der größte Wert von $\frac{q}{N}$ in allen Testfällen dieser Teilaufgabe. Deine Punktzahl für diese Teilaufgabe wird nach der folgenden Tabelle berechnet:

Bedingung	Punkte
$20 < m$	0 (wird als „Output isn't correct“ im CMS angezeigt)
$6 < m \leq 20$	$\frac{225}{m-2}$
$3 < m \leq 6$	$81 - \frac{2}{3}m^2$
$m \leq 3$	75

Sample-Grader

Sei T ein Array mit N ganzen Zahlen, wobei $T[i]$ die Art von Insekt i ist.

Der Sample-Grader liest die Eingabe in folgendem Format:

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

Wenn der Sample-Grader eine Verletzung des Protokolls feststellt, wird die Ausgabe `Protocol Violation: <MSG>` sein, wobei `<MSG>` Folgendes sein kann:

- `invalid parameter`: In einem Aufruf von `move_inside` oder `move_outside` war i nicht zwischen 0 und $N - 1$ (inklusive).
- `too many calls`: **Mindestens eine** der Funktionen `move_inside`, `move_outside` oder `press_button` wurde mehr als 40 000 Mal aufgerufen.

Sonst entspricht die Ausgabe des Sample-Graders dem folgenden Format:

- Zeile 1: der Rückgabewert von `min_cardinality`
- Zeile 2: q