

Задача В. Масив монет і запити зважування

Назва вхідного файлу:	standard input
Назва вихідного файлу:	standard output
Ліміт часу:	1 second
Ліміт використання пам'яті:	256 megabytes

Це інтерактивна задача.

Є n монет, розташованих в ряд та пронумерованих зліва направо цілими числами від 1 до n .

Рівно k ($k < n$) з цих монет є фальшивими, інші $(n - k)$ монет — справжні. Фальшиві монети легші за справжні. Всі справжні монети мають однакову вагу, в той час як фальшиві монети можуть мати різну вагу. Також відомо, що фальшиві монети **розташовані поспіль**, тобто мають номери $p, p + 1, \dots, (p + k - 1)$.

Вам потрібно знайти номер найлівішої фальшивої монети. Ви можете користуватись зважуванням, що аналогічне зважуванню на двочашкових терезах: вибрати дві множини монет, що не перетинаються, і дізнатись, яка з множин важить більше, або що множини важать однаково.

Формат вхідних даних

У першому рядку задано три цілі числа n, k, g ($1 \leq k < n \leq 10^4, 0 \leq g \leq 6$) — загальна кількість монет, кількість фальшивих монет та номер блоку тестів відповідно.

Формат вихідних даних

Для того, щоб дати відповідь, виведіть єдиний рядок у форматі «! p », де p ($1 \leq p \leq n$) — номер найлівішої фальшивої монети.

Протокол взаємодії

Щоб виконати запит зважування виведіть «? $s_1 s_2 a_1 a_2 \dots a_{s_1} b_1 b_2 \dots b_{s_2}$ », де s_1 та s_2 позначають розміри множин які зважуються, а масиви a та b позначають номери монет, що належать першій та другій множинам відповідно.

У відповідь на запит програма журі виведе одне ціле число x ($x \in \{0, 1, 2\}$). Якщо $x = 1$, то перша множина важча за другу; якщо $x = 2$, то друга множина важча за першу; якщо $x = 0$, то множини мають однакову вагу.

Якщо запит невалідний (тобто перевищено максимальну кількість запитів або параметри запиту є невалідними), програма журі виведе -1 та припинить роботу. У такому випадку завершіть роботу програми, щоб отримати вердикт **Неправильна відповідь**.

Подбайте про виклик методу `flush` після виводу кожного рядка. Для цього можна використовувати:

- `fflush(stdout), cout << endl` або `cout.flush()` в C++;
- `System.out.flush()` в Java;
- `flush(output)` в Pascal;
- `sys.stdout.flush()` в Python;
- дивіться документації для інших мов програмування.

Система оцінки

Визначимо q як максимальну кількість запитів зважування, яку ви можете виконати у тестах певного блоку.

1. (5 балів): $n \leq 16, q = 16$;
2. (9 балів): $k = 1, q = 16$;
3. (7 балів): $k = 1, q = 11$;

4. (16 балів): $k \leq 16$, $q = 11$;
5. (9 балів): ваги всіх фальшивих монет однакові, $q = 11$;
6. (до 54 балів): $q = 300$. Нехай максимальна кількість використаних зважувань дорівнює s . Якщо $s \leq 9$, то ви отримаєте 54 бали, а інакше ви отримаєте $\lfloor 54 \cdot \max(-0.0004 \cdot s + 0.3134, 0.018 + \frac{9.0773}{s}) \rfloor$ балів.

Код на мові C++, який обчислює кількість балів за останній блок тестів залежно від кількості використаних зважувань:

```
((s <= 9) ? 54 : int(54 * (max((-0.0004 * s + 0.3134), (0.018 + 9.0773 / s))))))
```

Таблиця розподілу балів

$s \leq 17$	Кількість балів	$18 \leq s \leq 27$	Кількість балів	$28 \leq s \leq 300$	Кількість балів
≤ 9	54	18	28	28	18
10	49	19	26	29-30	17
11	45	20	25	31-42	16
12	41	21	24	43-89	15
13	38	22	23	90-135	14
14	35	23	22	136-181	13
15	33	24	21	182-227	12
16	31	25	20	228-274	11
17	29	26-27	19	275-300	10

Приклад

standard input	standard output
4 1 0	? 1 1 1 2
0	? 1 1 2 4
0	? 1 1 3 4
2	! 3