

## Задача А. Фільми

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

Дедпул переглядає фільми про Росомаху з різних всесвітів на сайті фільмів мультивсесвіту. Відомо, що всього є  $n$  фільмів, і на кожній сторінці сайту відображається  $k$  різних фільмів (на останній все, що залишилось). Дедпул не має багато часу, тому відразу переходить на останню сторінку і дивиться всі фільми з неї.

Кожен фільм триває рівно  $s$  хвилин. Скільки часу він витратив на перегляд фільмів?

### Формат вхідних даних

Перший рядок містить три цілі числа  $n$ ,  $k$ ,  $s$  ( $1 \leq n, k, s \leq 10^4$ ) — кількість фільмів, кількість фільмів на одній сторінці, тривалість одного фільму.

### Формат вихідних даних

Виведіть одне число  $t$ , кількість часу, яку Дедпул витратить на перегляд фільмів.

### Приклади

| <code>standard input</code> | <code>standard output</code> |
|-----------------------------|------------------------------|
| 20 3 60                     | 120                          |
| 65 10 30                    | 150                          |
| 100 20 90                   | 1800                         |

### Зауваження

У першому прикладі ми маємо 20 фільмів і на кожній сторінці, окрім останньої, є по 3 фільми, тому на останній сторінці знаходиться 2 фільми, перегляд тих двох фільмів займе всього 120 хвилин.

У другому прикладі є загалом 65 фільмів і по 10 фільмів на сторінку, тому на останній буде 5 фільмів і перегляд займе 150 хвилин.

У третьому прикладі є 100 фільмів, по 20 фільмів на сторінку, тому на останній сторінці буде 20 фільмів і Дедпул переглядатиме їх 1800 хвилин.

## Задача В. Катамарани

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

Група з  $n$  людей планує покататися на катамаранах.

Вам, як лідеру групи, доручили замовити катамарани. Ви знаєте, що один катамаран витримує вагу не більшу за 100 кілограмів, а також знаєте, скільки важить кожен член групи.

Також вам відомо, що у вашій групі людина може важити: або 20, або 40, або 60, або 80, або 100 кілограмів.

Для того, щоб витратити якомога менше грошей, ви вирішили написати програму, що порахує мінімальну потрібну кількість катамаранів.

### Формат вхідних даних

Перший рядок містить одне ціле число  $n$  ( $1 \leq n \leq 1000$ ) — кількість людей у групі.

Другий рядок містить  $n$  цілих чисел  $a_1, a_2, \dots, a_n$  ( $a_i \in \{20, 40, 60, 80, 100\}$ ) — вага кожної людини.

### Формат вихідних даних

Виведіть одне ціле число — мінімальну потрібну кількість катамаранів.

### Приклади

| standard input   | standard output |
|------------------|-----------------|
| 4<br>20 40 80 80 | 3               |
| 4<br>20 40 20 20 | 1               |

### Зауваження

У першому прикладі ми можемо посадити в один катамаран перших двох людей, в другий посадити третю людину і в третій катамаран четверту людину. І ми не можемо посадити всіх у два катамарани, бо 2 не може сидіти ні з 3, ні 4, і 3 з 4 теж не можуть бути разом.

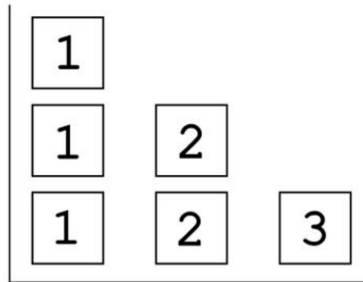
У другому прикладі ми можемо посадити всіх в один катамаран, бо сума їх ваг рівна 100 кілограмів, тобто катамаран їх витримує.

## Задача С. Діагональні числа

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

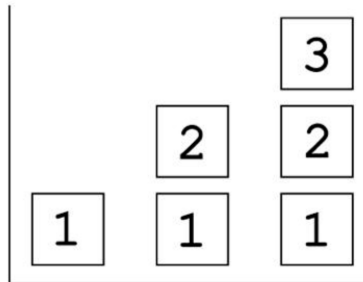
Васильку подарували кубики, на яких було написано числа від 1 до  $n$ . На одному кубіку було написано число  $n$ , на двох було написано  $n - 1$ , і так далі, на  $n - 1$  кубіку було написано число 2, і на  $n$  кубиках було написано число 1.

Він розмістив кубики у квадратну рамку без верхньої межі, так що всі кубики розміщені під діагоналлю, що проходить з лівого верхнього кута до правого нижнього кута. У першому стовпчику зліва є  $n$  кубиків, на яких записане число 1; у другому стовпчику  $n - 1$  кубик, на яких записане число 2... в  $n$ -му стовпчику один кубик, на якому записане число  $n$ . Тобто, в  $i$ -му стовпчику  $n - i + 1$  кубик, на яких записане число  $i$ .



Початковий стан

Васильку захотів переставити їх так, щоб всі кубики були розташовані під **протилежною діагоналлю**. Так, щоб у першому рядку знизу було  $n$  кубиків, на яких було записане число 1; у другому рядку має бути  $n - 1$  кубик, на яких було записане число 2... в  $n$ -му рядку був один кубик, на якому було записане число  $n$ . Тобто, в  $i$ -му рядку має бути  $n - i + 1$  кубик, на яких має бути записане число  $i$ .



Кінцевий стан

Також він вирішив, що буде переставляти кубики лише з вершини одного стовпця на вершину іншого (не обов'язково сусіднього). Допоможіть йому зробити це в найефективніший спосіб — знайдіть мінімальну кількість перестановок, які йому потрібні, та виведіть, які перестановки потрібно зробити Васильку.

Ви можете отримати часткові бали, детальніше перегляньте нижче.

### Формат вхідних даних

Перший рядок містить одне ціле число  $n$  ( $3 \leq n \leq 1000$ ) — розміри квадрата.

### Формат вихідних даних

У першому рядку виведіть одне ціле число  $k$  ( $1 \leq k \leq 10^6$ ) — мінімальна кількість перестановок, за які ви можете переставити кубики.

У кожному з наступних  $k$  рядків виведіть два цілі числа  $a$  та  $b$  ( $1 \leq a, b \leq n$ ;  $a \neq b$ ), де  $a$  — номер стовпчика, з якого ви берете кубик, а  $b$  — номер стовпчика, на який ви ставите кубик.

## Система оцінювання

1. (12 балів):  $n = 4$ ;
2. (20 балів):  $n = 5$ ;
3. (20 балів):  $n = 6$ ;
4. (20 балів):  $n \leq 10$ ;
5. (20 балів):  $n \leq 100$ ;
6. (8 балів): без додаткових обмежень.

Ви можете отримати половину балів за кожен блок, якщо для кожного тесту блока виведете правильне  $k$ . Ви також отримаєте чверть балів, якщо ви виведете неправильне  $k$ , але інструкції будуть вірними, також в цьому випадку  $k \leq 10^6$ .

Зверніть увагу, що для того, щоб отримати половину балів, вам потрібно вивести правильний  $k$  і

- або більше нічого не виводити; тобто, вивести лише  $k$ , але без інструкцій;
- або вивести повністю всі  $k$  інструкцій, де всі номери стовпчиків від 1 до  $n$ . Будь-яких додаткових вимог до них немає.

Якщо ви виведете, наприклад, лише кілька інструкцій, або за багато інструкцій, або числа, що не є номерами стовпчиків тощо, то ви отримаєте 0 балів.

**Часткові бали не даються за приклад з умови, бо він оцінюється в 0 балів.**

Вихідні дані для половини балів, коли  $n = 3$ , можуть виглядати так:

8

або так:

8

1 2

2 1

1 2

2 1

1 2

2 1

1 2

2 1

Вихідні дані для чверті балів, коли  $n = 3$ , можуть виглядати так:

10

1 3

3 1

3 2

1 3

2 1

2 3

1 3

2 3

1 2

3 2

Якщо ви виведете кількість інструкцій, яка не рівна  $k$ , то ви отримаєте 0 балів:

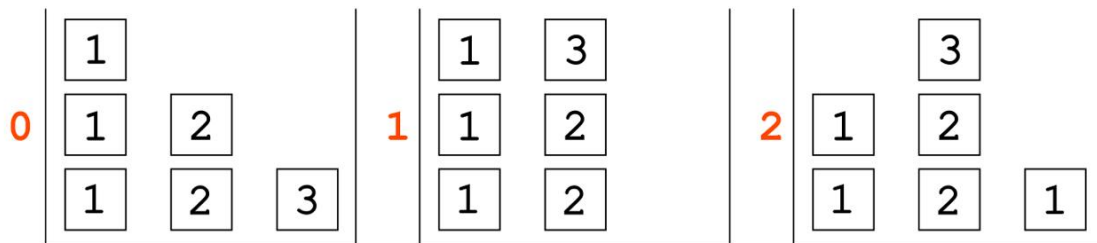
8  
1 3  
3 1  
3 2  
1 3  
2 1  
2 3  
1 3  
2 3  
1 2  
3 2

### Приклад

| standard input | standard output   |
|----------------|---|
| 3              | 8<br>3 2<br>1 3<br>2 1<br>2 3<br>1 3<br>2 3<br>1 2<br>3 2 |

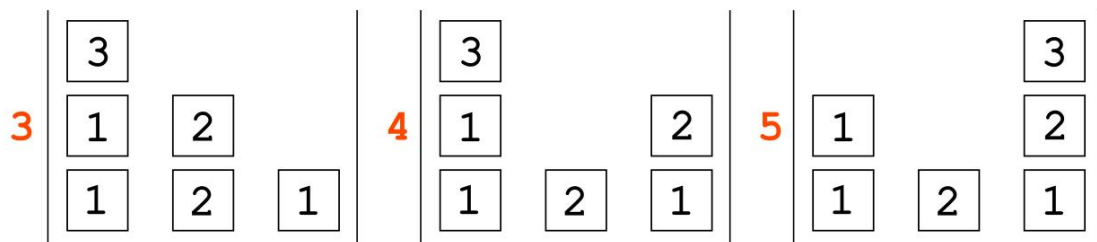
### Зауваження

Спершу ми розставимо останній стовпчик. Тому єдиним першим ходом, щоб переставити кубики за мінімальну кількість ходів, може бути, перенести 3 з третього стовпчика до другого, бо інакше ми не зможемо поставити 1 на початок третього стовпчика, якщо будемо переносити не 3 або не в другий стовпчик.



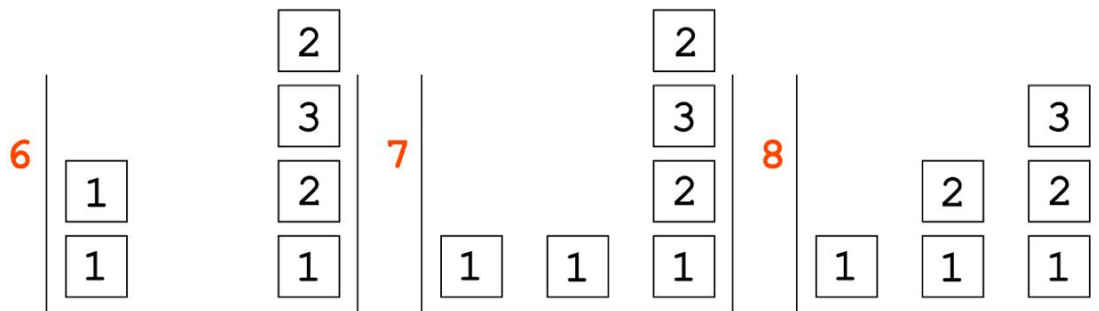
Перші 3 стани

Після того як ми поставили 1, нам треба поставити 2 і 3 на своє місце в третьому стовпчику, знову єдиний спосіб це зробити спершу перенести 3 в перший стовпчик, бо інакше ми не зможемо поставити 2 в третій наступним ходом, а потім і саму 3 поставити на своє місце.



Наступні 3 стани

Після цього ми хочемо правильно поставити другий стовпчик, і щоб 1 було першим числом, спершу треба забрати 2, яку ми можемо забрати лише в останній стовпчик. Після того як ми поставили 1, нам залишається лише повернути 2 в другий стовпчик, щоб отримати потрібний нам стан кубиків.



Кінцеві 3 стани

## Задача D. Непарні рядки

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

Одного разу s1mple підійшов до Кості, відомого розв'язувача задач, і сказав:

*«Якщо хочеш стати кращим, тобі потрібна постійна практика. Ось задача для тренування:»*

Дано матрицю  $a$  розміром  $n \times m$  ( $n$  — кількість рядків;  $m$  — кількість стовпчиків), кожен елемент якої 0 або 1. Відомо, що в кожному стовпчику є рівно  $c_i$  одиничок. У кожному стовпці можна довільно розставити елементи в середині цього стовпчика. Треба максимізувати кількість рядків із непарною кількістю одиничок і знайти таку матрицю.

Костя мовчки кивнув, сів за стіл і почав працювати, адже знав: кожне тренування наближає до майстерності.

Костя не впорався із задачею і просить вас допомогти йому її розв'язати.

**Ви можете отримати часткові бали, якщо знайдете лише число, а не матрицю. Детальніше нижче.**

### Формат вхідних даних

Перший рядок містить два цілі числа  $n$  та  $m$  ( $1 \leq n \cdot m \leq 10^6$ ) — розміри матриці.

Другий рядок містить  $m$  цілих чисел  $c_1, c_2, \dots, c_m$  ( $0 \leq c_i \leq n$ ) — кількість одиничок у кожному стовпчику.

### Формат вихідних даних

У першому рядку виведіть одне ціле число  $t$  ( $0 \leq t \leq n$ ) — кількість рядків в матриці з непарною сумою.

У кожному з наступних  $n$  рядках виведіть по  $m$  цілих чисел  $a_{ij}$  ( $0 \leq a_{ij} \leq 1$ ) — числа матриці.

### Система оцінювання

- (10 балів):  $n, m \leq 5$ ;
- (8 балів): кількість одиничок в матриці не більша за  $n$ ;
- (20 балів): кількість одиничок в кожному стовпці не більша за  $n/2$ ;
- (14 балів):  $n, m \leq 50$ ;
- (14 балів):  $n \leq 3\,000$ ;
- (14 балів):  $n \cdot m \leq 3 \cdot 10^5$ ;
- (20 балів): без додаткових обмежень.

Ви можете отримати половину балів за кожен блок, якщо для кожного тесту блока виведете правильний  $t$ .

Зверніть увагу, що для того, щоб отримати часткові бали, вам потрібно вивести правильний  $t$  і

- або більше нічого не виводити; тобто, вивести лише  $t$ , але не матрицю;
- або вивести повністю матрицю, яка складається з 0 та 1, яка необов'язково правильна. Наприклад, що складається з одних нулів.

Якщо ви виведете, наприклад, лише кілька рядків, або за багато рядків, числа крім 0 та 1 тощо, то ви отримаєте 0 балів.

Вихідні дані для половини балів, для другого прикладу, можуть виглядати так:

2

або так:

```
2
0 0 0 0
0 0 0 0
0 0 0 0
0 0 0 0
```

але він не може виглядати так:

```
2
0 0 0 0
0 0 0 0
0 0 0 0
0 0 0 0
0 0 0 0
```

і не може виглядати так:

```
2
10 0 0 0
0 0 0 0
0 0 0 0
0 0 0 0
```

## Приклади

| standard input | standard output   |
|----------------|---|
| 8 4<br>6 1 6 1 | 6<br>1 1 1 0<br>1 0 1 1<br>1 0 1 0<br>1 0 1 0<br>1 0 0 0<br>1 0 0 0<br>0 0 1 0<br>0 0 1 0 |
| 4 4<br>3 0 3 0 | 2<br>1 0 1 0<br>1 0 1 0<br>1 0 0 0<br>0 0 1 0   |
| 7 3<br>4 3 2   | 7<br>1 1 1<br>1 0 0<br>1 0 0<br>1 0 0<br>0 1 0<br>0 1 0<br>0 0 1                          |

## Зауваження

У першому прикладі перший і третій стовпчики перетинаються принаймні в 4 позиціях, тобто якби були лише ці два стовпчики, то ми б мали 4 парних рядки, але оскільки у нас ще є два стовпчики з 1 одиничкою, то ми можемо перетворити два з них на непарні, тому оптимальна відповідь буде 6 непарних.

У другому прикладі ми можемо забути про другий і четвертий стовпчики, бо в них нема одиничок, тобто вони не змінять парності жодному рядку, а перший і третій перетинаються принаймні у 2 рядках, тобто принаймні два рядки будуть парними, тому відповідь 2.



У третьому прикладі відповідь 7, бо існує матриця, яка має 7 непарних рядків і задовольняє умову, і не існує матриця, що має більше ніж 7.

## Задача Е. Три запити

|                             |                 |
|-----------------------------|-----------------|
| Назва вхідного файлу:       | standard input  |
| Назва вихідного файлу:      | standard output |
| Ліміт часу:                 | 4 seconds       |
| Ліміт використання пам'яті: | 256 megabytes   |

Дано масив  $a$  довжини  $n$  та  $q$  запитів. Також ми маємо бінарний масив  $w$  нескінченної довжини, початково всі  $w_i = 1$ .

Є три типи запитів:

- «1  $x$ » — змінюємо значення  $w_x$  на протилежне (з 1 на 0, і навпаки).
- «2  $l r$ » — потрібно порахувати кількість унікальних чисел у масиві  $a$  на відрізку  $[l, r]$  для яких  $w_{a_i} = 1$  і  $l \leq i \leq r$ .
- «3  $x t$ » — присвоїти  $a_x$  значення  $t$ .

Дайте відповідь на кожен запит другого типу.

### Формат вхідних даних

Перший рядок містить два цілі числа  $n$  та  $q$  ( $1 \leq n, q \leq 3 \cdot 10^5$ ) — довжина масиву і кількість запитів.

Другий рядок містить  $n$  цілих чисел  $a_1, a_2, \dots, a_n$  ( $1 \leq a_i \leq 10^9$ ) — значення елементів масиву.

Кожен з наступних  $q$  рядків починається з цілого числа  $type$  ( $1 \leq type \leq 3$ ) — номер типу кожного запиту:

- Якщо  $type = 1$ , запит містить одне ціле число  $x$  ( $1 \leq x \leq 10^9$ ) — змінити значення  $w_x$  на протилежне.
- Якщо  $type = 2$ , запит містить два цілі числа  $l$  та  $r$  ( $1 \leq l \leq r \leq n$ ) — порахувати кількість унікальних чисел у масиві  $a$  на відрізку  $[l, r]$  для яких  $w_{a_i} = 1$  і  $l \leq i \leq r$ .
- Якщо  $type = 3$ , запит містить два цілі числа  $x$  та  $t$  ( $1 \leq x \leq n, 1 \leq t \leq 10^9$ ) — замінити значення  $a_x$  на  $t$ .

### Формат вихідних даних

Для кожного запиту другого типу потрібно вивести кількість унікальних чисел на відрізку в окремому рядку.

### Система оцінювання

- (8 балів):  $n, q \leq 10^3$ ;
- (6 балів): лише запити 2 типу;  $n = q; l_i = 1; r_i = i$ ;
- (13 балів): лише запити 2 типу;
- (10 балів): лише запити 1 і 2 типу; всі  $a_i$  попарно різні;
- (14 балів): лише запити 1 і 2 типу; всі  $w_{a_i}$  можуть змінитися лише раз;
- (7 балів): лише запити 1 і 2 типу;
- (14 балів): лише запити 2 і 3 типу;
- (8 балів): у будь-який момент часу  $a_i \leq 100$ ;
- (10 балів):  $n, q \leq 5 \cdot 10^4$ ;

10. (10 балів): без додаткових обмежень.

### Приклад

| standard input      | standard output |
|---------------------|-----------------|
| 10 5                | 2               |
| 3 4 3 4 3 2 3 1 2 1 | 1               |
| 2 2 5               | 2               |
| 1 3                 |                 |
| 2 2 5               |                 |
| 3 4 5               |                 |
| 2 2 5               |                 |

### Зауваження

У прикладі для першого запити другого типу відрізок виглядає так  $[4, 3, 4, 3]$ , тобто тут є числа 3, 4, і  $w_3, w_4$  є рівні 1, тому відповідь 2. Після наступного запити 1 типу,  $w_3$  стає рівним 0, тому для наступного запити відповідь дорівнює 1. Після наступного запити масив почне виглядати так:  $[3, 4, 3, 5, 3, 2, 3, 1, 2, 1]$ . В останньому запиті відрізок виглядатиме так  $[4, 3, 5, 3]$ , тобто там є числа 3, 4, 5, відповідно відповідь 2, бо  $w_3 = 0$ .