

## Задача А. Вирівняти масив

Назва вхідного файлу: `standard input`  
Назва вихідного файлу: `standard output`  
Ліміт часу: 0.25 seconds  
Ліміт використання пам'яті: 256 megabytes

Для масиву цілих чисел  $A$ , розмірності  $N$ , визначте мінімальну кількість елементів, які потрібно видалити, щоб залишити лише елементи однакового значення.

### Формат вхідних даних

Перший рядок містить ціле число  $N$  ( $1 \leq N \leq 10^5$ )

У другому рядку містяться цілі числа  $A_i$  ( $1 \leq A_i \leq 10^4$ ), які розділяються пропуском.

### Формат вихідних даних

Вивести шукану мінімальну кількість.

### Приклад

standard input	standard output
5 3 3 2 1 3	2

## Задача В. Архіватор

Назва вхідного файлу:	standard input
Назва вихідного файлу:	standard output
Ліміт часу:	0.25 seconds
Ліміт використання пам'яті:	256 megabytes

Як відомо, всі дані в комп'ютері кодуються числами, а за допомогою програм-архіваторів можна зробити так, щоб ці дані займали менше місця на носіях (наприклад, на флешці). Один з методів ущільнення (архівування) даних полягає в тому, що в послідовності чисел відшукують підпоследовність, яка складається з однакових чисел, і замінюють її двома числами: перше означає кількість повторень, а друге — число, яке повторюється.

Наприклад, заархівувавши за цим алгоритмом таку послідовність

35 12 27 27 27 27 27 27 27 27 15 83 83 83 83 83,

отримаємо

1 35 1 12 8 27 1 15 5 83,

що можна прочитати, як один раз 35, один раз 12, вісім разів 27, один раз 15, п'ять разів 83.

Складіть програму, яка за введеним рядком чисел сформує архівну послідовність чисел. Формат вхідних даних

Формат вихідних даних У вихідний потік вивести рядок чисел, відокремлених одним пропуском - архівну послідовність чисел. Числа не повинні перевищувати 255. Примітка. Послідовність з більше ніж 255 однакових чисел кодується декількома «блоками» по 255, після яких, можливо, йде «блок» із залишком.

### Формат вхідних даних

Вхідний потік містить від 1 до 65535 цілих чисел  $a_i$  ( $0 \leq a_i \leq 255$ ), відокремлених одним пропуском - послідовність для архівування.

### Формат вихідних даних

У вихідний потік вивести рядок чисел, відокремлених одним пропуском - архівну послідовність чисел. Числа не повинні перевищувати 255.

### Приклад

standard input	standard output
35 27 27 27 27 15 83 83 83	1 35 4 27 1 15 3 83

### Зауваження

Послідовність з більше ніж 255 однакових чисел кодується декількома «блоками» по 255, після яких, можливо, йде «блок» із залишком.

## Задача С. Ланцюги

Назва вхідного файлу: `standard input`  
Назва вихідного файлу: `standard output`  
Ліміт часу: `0.25 seconds`  
Ліміт використання пам'яті: `256 megabytes`

Фірма виготовляє різнокольорові ланцюги. Кожна ланка може бути пофарбована в один із  $k$  кольорів. Замовник при оформленні залишає довжину бажаного ланцюга  $n$  та пари із номерами ланок, які обов'язково повинні бути пофарбовані різними кольорами. Потрібно визначити кількість ланцюгів, які можуть бути виготовлені враховуючи побажання замовника.

### Формат вхідних даних

В першому рядку задаються 3 числа:  $n, k, m$  ( $1 \leq n, k \leq 8$ ,  $0 \leq m \leq 10$ ), де  $m$  - кількість пар ланок.

Далі в  $m$  рядках задаються пари ланок  $u$  і  $v$  ( $1 \leq u, v \leq n$ ,  $u \neq v$ ).

### Формат вихідних даних

Одне число - відповідь на задачу.

### Приклади

<code>standard input</code>	<code>standard output</code>
2 2 1 1 2	2
4 4 0	256

### Зауваження

У першому прикладі таких ланцюгів сворити можна лише два: із кольорами 1-2 та 2-1.

У другому прикладі немає обмежень, тому всі можливі ланцюги будуть підходящими, а таких всього 256.

## Задача D. Мандрівники

Назва вхідного файлу: `standard input`  
Назва вихідного файлу: `standard output`  
Ліміт часу: 0.25 seconds  
Ліміт використання пам'яті: 256 megabytes

$N$  мандрівників опинились вночі перед мостом. Пройти міст можна лише з освітленням. На щастя у одного із мандрівників у рюкзаку завалявся ліхтарик. Міст витримує лише двох осіб. Якщо йде група із двох, то вона рухається із швидкістю найповільнішого серед них. Швидкість руху кожного мандрівника відома. Необхідно визначити найменший час із яким можна всім перетнути міст.

### Формат вхідних даних

В першому рядку задано  $N$  ( $1 \leq N \leq 10^5$ ) - кількість мандрівників.

В наступному рядку через пробіл задані швидкості руху кожного у хвилинах. Швидкість не перевищує  $10^6$ .

### Формат вихідних даних

Одне число - відповідь на задачу у хвилиnach.

### Приклад

standard input	standard output
4	60
5 10 20 25	

### Зауваження

Нехай ліхтар у першого мандрівника. Він бере другого із собою і переходить на іншу сторону. На це в них йде 10 хвилин. Потім перший повертається назад сам, проходить ще 5 хвилин (всього пройшло 15хв). Далі третій і четвертий мандрівники переходять міст за 25 хвилин (всього пройшло 40хв), передають ліхтар другому мандрівнику і той переходить міст (всього пройшло 50хв), забирає першого мандрівника і вже остаточно всі переходять міст (всього 60хв).

## Задача Е. Судоку

Назва вхідного файлу: standard input  
Назва вихідного файлу: standard output  
Ліміт часу: 0.25 seconds  
Ліміт використання пам'яті: 256 megabytes

Вам задано деяке поле в головоломці судоку. Ваша задача - розв'язати її. Судоку - головоломка в якій необхідно заповнити числами від 1 до 9 поля таким чином, щоб у кожному стовпчику та рядку, а також у квадратах 3x3 були усі числа від 1 до 9.

### Формат вхідних даних

Задано 9 рядків по 9 символів, серед яких можуть бути цифри 1..9 та символ порожньої клітинки. Символ "." - означає порожню клітинку.

### Формат вихідних даних

Вивести 9 рядків по 9 символів із розв'язаною головоломкою.

### Приклад

standard input	standard output
53..7....	534678912
6..195...	672195348
.98....6.	198342567
8...6...3	859761423
4..8.3..1	426853791
7...2...6	713924856
.6....28.	961537284
...419..5	287419635
....8..79	345286179

### Зауваження

Пояснення до приладу

5	3			7					5	3	4	6	7	8	9	1	2			
6			1	9	5				6	7	2	1	9	5	3	4	8			
	9	8						6				1	9	8	3	4	2	5	6	7
8				6					3	8	5	9	7	6	1	4	2	3		
4			8		3				1	4	2	6	8	5	3	7	9	1		
7				2					6	7	1	3	9	2	4	8	5	6		
	6						2	8		9	6	1	5	3	7	2	8	4		
			4	1	9				5	2	8	7	4	1	9	6	3	5		
				8				7	9	3	4	5	2	8	6	1	7	9		