

## Задача А. Карточки

Имя входного файла: *стандартный ввод*  
Имя выходного файла: *стандартный вывод*  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 512 мегабайт

На столе в ряд разложены  $n$  карточек. На каждой карточке написано три различных буквы. *Степень сходства* двух карточек будем называть количество одинаковых букв на них; например, степень сходства карточек «ABC» и «ACD» равна двум, а степень сходства «ABC» и «DFE» — нулю.

Разрешается удалять карточку со стола, если степень сходства её левого и правого соседей не меньше двух. После удаления карточки левый и правый её соседи сами становятся соседями. Крайние карточки удалять нельзя.

Найдите максимальное количество карточек, которые можно удалить со стола.

### Формат входных данных

В первой строке задано целое число  $n$  — количество карточек ( $2 \leq n \leq 50$ ). В следующих  $n$  строках заданы сами карточки в порядке их следования в ряду. Каждая из этих строк состоит ровно из трёх заглавных букв латинского алфавита.

### Формат выходных данных

В первой строке выведите одно число — максимальное количество карточек, которые можно удалить со стола.

### Примеры

<i>стандартный ввод</i>	<i>стандартный вывод</i>
5 ABC CDE ABC CDE ABC	3
5 ABC CDE EFG GHI IJK	0
5 ADR FDB ABC CDE ABD	3

## Задача В. Свёртка

Имя входного файла: *стандартный ввод*  
Имя выходного файла: *стандартный вывод*  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 512 мегабайт

Рассмотрим все подмножества множества  $U = \{0, 1, 2, \dots, n-1\}$ . Каждому подмножеству  $A = \{a_1, a_2, \dots, a_k\}$  соответствует уникальное целое число, равное  $p(A) = \sum_{i=1}^k 2^{a_i}$ . Функцию  $F$  от  $n$ -элементного множества будем задавать массивом целых чисел  $f$  длины  $2^n$  так, что значение функции  $F(A)$  равно  $f[p(A)]$ .

Вам даны две функции  $F$  и  $G$ , нужно найти функцию  $H$  такую, что

$$H(A) = \sum_{B \cup C = A} F(B)G(C).$$

### Формат входных данных

В первой строке заданы два целых числа  $n$  и  $t$  ( $1 \leq n \leq 16$ ,  $1 \leq t \leq 100$ ). Здесь  $n$  — размер множества  $U$ , а  $t$  — количество тестовых случаев. Во второй строке заданы целые числа  $a$  и  $b$ , каждое от 1 до  $10^9$ . Эти числа используются в следующем генераторе псевдослучайных чисел:

```
1. unsigned int cur = 0; // беззнаковое 32-битное число
2. unsigned int nextRand16() {
3.     cur = cur * a + b; // вычисляется по модулю 232
4.     return cur / 216; // целое число от 0 до 216 - 1
5. }
```

Тестовые случаи генерируются последовательно. В каждом из них сперва генерируются по порядку элементы массива  $f$  (значения функции  $F$ ), а затем генерируются по порядку элементы массива  $g$  (значения функции  $G$ ). Каждое следующее целое число генерируется вызовом функции `nextRand16()`.

### Формат выходных данных

В ответ на каждый тестовый случай выведите в отдельной строке одно целое число:

$$\left( \sum_A H(A) \cdot (p(A) + 1) \right) \bmod 2^{32}.$$

### Примеры

<i>стандартный ввод</i>	<i>стандартный вывод</i>
3 2 30 239017	2723387430 3167905008
16 2 239 17	551267264 1632349120

### Пояснения к примерам

Массивы в первом тесте из примера:

$f_1$ : 3, 113, 3395, 36331, 41370, 61471, 9130, 11774

$g_1$ : 25547, 45526, 55066, 13590, 14501, 41817, 9356, 18543

$h_1$ : 76641, 8167827, 273846333, 5284992017, 1656829263, 11450721456, 3699971823, 14260048942

$f_2$ : 32024, 43238, 51978, 52034, 53714, 38578, 43250, 52338

$g_2$ : 62834, 50034, 59250, 8050, 44914, 36722, 53106, 20338

$h_2$ : 2012196016, 6482475400, 8243104152, 15561662464, 7225902008, 16869349792, 22350138288, 44342816072

## Задача С. Деление перестановок

Имя входного файла: *стандартный ввод*  
Имя выходного файла: *стандартный вывод*  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 512 мегабайт

*Перестановкой* числа  $n$  назовём число, содержащее в точности цифры, из которых состоит число  $n$ , но, возможно, в другом порядке. Две перестановки считаются различными, если они различны как числа. К примеру, различные перестановки числа  $n = 313$  — это числа 133, 313 и 331.

Даны числа  $n$  и  $m$ . Найдите количество перестановок числа  $n$ , делящихся на  $m$ .

### Формат входных данных

В первой строке заданы через пробел два целых числа  $n$  и  $m$  ( $1 \leq n \leq 10^{15}$ ,  $1 \leq m \leq 50$ ,  $n$  состоит только из цифр 1–9).

### Формат выходных данных

В первой строке выведите одно число — количество перестановок числа  $n$ , делящихся на  $m$ .

### Примеры

<i>стандартный ввод</i>	<i>стандартный вывод</i>
133 7	1
2753 5	6
1112225 5	20

## Задача D. Игра

Имя входного файла: *стандартный ввод*  
Имя выходного файла: *стандартный вывод*  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 512 мегабайт

Два игрока играют в следующую игру:

- Дана строка.
- Игроки по очереди стирают один символ с начала или конца строки, после чего дописывают его в конец результата.
- Цель первого игрока состоит в том, чтобы результирующая строка была как можно больше (в лексикографическом порядке), а цель второго — чтобы она была как можно меньше.

Определите полученный результат в предположении, что оба игрока играют оптимально.

### Формат входных данных

В единственной строке записана строка, состоящая из строчных латинских букв, длиной не более 100 символов.

### Формат выходных данных

Выведите единственную строку — полученный результат при оптимальной игре обоих игроков.

### Примеры

<i>стандартный ввод</i>	<i>стандартный вывод</i>
abcde	eadbc
aaaca	aaca



## Задача F. Максимальное среднее

Имя входного файла: *стандартный ввод*  
Имя выходного файла: *стандартный вывод*  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 512 мегабайт

На доске записана последовательность натуральных чисел. За один ход разрешается выбрать два соседних числа в последовательности и заменить их на их среднее арифметическое. Будем делать ходы до тех пор, пока на доске не останется одно число. Какое максимальное число может получиться?

### Формат входных данных

В первой строке входного потока задано целое число  $n$  — количество чисел в последовательности ( $1 \leq n \leq 100$ ). Во второй строке заданы  $n$  целых чисел  $a_1, a_2, \dots, a_n$  через пробел — сама последовательность ( $1 \leq a_i \leq 10\,000$ ).

### Формат выходных данных

Выведите единственное число — максимальное число, которое может получиться в результате. Ответ считается верным, если он отличается от точного ответа не более чем на  $10^{-6}$  по абсолютной или относительной погрешности.

### Пример

<i>стандартный ввод</i>	<i>стандартный вывод</i>
3 1 2 3	2.25

## Задача G. Разрезание многоугольника

Имя входного файла: *стандартный ввод*  
Имя выходного файла: *стандартный вывод*  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 512 мегабайт

Определите количество способов разрезать выпуклый многоугольник с  $n$  сторонами на  $k$  частей. Разрезы разрешено проводить только от вершины к вершине, таким образом, что многоугольник делится ровно на две части. Вершины считаются пронумерованными различными числами, порядок разрезов не важен.

Например, квадрат можно разрезать тремя способами. Два из них — произвести разрез по какой-либо из диагоналей (таким образом получается две части), и один — вообще не производить разрезов (получается одна часть).

Так как количество способов может быть очень большим, то нужно вывести ответ по модулю  $10^9$  (то есть последние 9 цифр).

### Формат входных данных

В единственной строке заданы два целых числа  $n$  и  $k$  ( $3 \leq n \leq 100$ ,  $1 \leq k \leq 100$ ).

### Формат выходных данных

Выведите количество способов по модулю  $10^9$ . Если многоугольник невозможно разрезать на  $k$  частей, выведите  $-1$ .

### Примеры

<i>стандартный ввод</i>	<i>стандартный вывод</i>
4 2	2
100 1	1
6 4	14
31 20	956146480
3 4	-1

## Задача Н. Лыжная трасса

Имя входного файла: *стандартный ввод*  
Имя выходного файла: *стандартный вывод*  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 512 мегабайт

Лыжница Аня находится на старте лыжной трассы. Трасса представляет собой участок плоскости, на котором введена декартова система координат. Координаты измеряются в метрах. Старт соответствует прямой  $y = 0$ , а финиш — прямой  $y = n$ . На прямых  $y = 1, y = 2, \dots, y = n$  расположены ворота. Каждые ворота состоят из двух флажков, расстояние между которыми равно  $d$  метрам. Ворота характеризуются координатой левого края: на каждой прямой  $y = k$  эта координата равна  $x_k$ .

Чтобы пройти трассу, Ане нужно последовательно пройти сквозь все ворота. Другими словами, пересекать каждую прямую вида  $y = k$  можно лишь в точках  $(x, y)$  таких, что  $x_k \leq x \leq x_k + d$ . Движение между двумя соседними воротами ничем не ограничено.

Аня может начать движение из любой точки прямой  $y = 0$  и закончить движение в любой точке прямой  $y = n$ , находящейся внутри последних ворот. Найдите минимальную длину маршрута, по которому можно пройти трассу. Размером Ани можно пренебречь и считать её точкой.

### Формат входных данных

В первой строке входного файла заданы через пробел два целых числа  $n$  и  $d$  — количество ворот и ширина ворот в метрах ( $1 \leq n \leq 100, 2 \leq d \leq 10$ ). В следующей строке заданы  $n$  целых чисел  $x_1, x_2, \dots, x_n$  через пробел — координаты левого края первых, вторых,  $\dots$ , последних ворот трассы ( $0 \leq x_k \leq 1000$ ).

### Формат выходных данных

Выведите в выходной файл одно число — минимальную длину пути по трассе с точностью не менее шести знаков после запятой.

### Примеры

<i>стандартный ввод</i>	<i>стандартный вывод</i>
3 2 1 2 1	3
5 2 0 2 5 7 10	9.94427190999916

## Задача I. Саммит

Имя входного файла: *стандартный ввод*  
Имя выходного файла: *стандартный вывод*  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 512 мегабайт

Саммит (от англ. *summit*) – встреча на высшем уровне, то есть переговоры глав государств и/или правительств.

---

материал из Википедии – свободной энциклопедии

На дворе 2100 год. На саммит по поводу поддержки стран Нью-Африки собрались главы нью-европейских стран. Всем известно, что доход Одной Небольшой Страны (ОНС) на текущий год составляет  $C$  местных бананов (да, да, в 2100 году валютой являются бананы).

Президент Одной Крупной Страны (ОКС) предложил каждый год увеличивать доход ОНС на такое количество бананов, какова сумма цифр в числе, равном текущему доходу. Но критически настроенный президент Соединенных Штатов Китая (СШК) быстро осознал, что скоро доход этой страны превысит доход СШК. Но на сколько? Вам, как главному программисту СШК, нужно срочно написать программу, которая посчитает доход этой страны через  $N$  лет.

### Формат входных данных

В первой строке содержатся числа  $C$  и  $N$  ( $1 \leq C, N \leq 10^{14}$ ).

### Формат выходных данных

В первой строке вывода должно содержаться одно число – доход ОНС через  $N$  лет.

### Пример

<i>стандартный ввод</i>	<i>стандартный вывод</i>
2 2	8

## Задача J. Беру? Нет, не беру!

Имя входного файла: *стандартный ввод*  
Имя выходного файла: *стандартный вывод*  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 512 мегабайт

Саша и Петя играют в весёлую игру. Петя рисует на листе бумаги таблицу  $m \times n$  и заполняет её целыми числами. После этого Саша ставит свою фишку на клетку  $(1,1)$  и может каждым своим ходом двигать её вправо или вниз. Кроме того, если Сашина фишка находится на числе большем, чем все числа, которые он брал до этого, Саша может сказать «беру», и тогда Петя записывает на бумажку то число, на котором стоит Сашина фишка (если Саша не брал ранее ни одного числа, то он может сказать «беру» на любом числе). Игра заканчивается, когда Сашина фишка больше не может ходить, и количество очков, которые набрал Саша, зависит от того, сколько чисел он взял. Помогите Саше «взять» как можно больше чисел!

### Формат входных данных

Входные данные состоят из одного или нескольких тестовых случаев (не более пяти). Каждый тестовый случай начинается строкой, в которой записаны числа  $m$  и  $n$  ( $1 \leq m, n \leq 100$ ). Далее следуют  $n$  строк по  $m$  чисел в каждой — таблица, которую нарисовал Петя. Все числа не превосходят по модулю 10 000.

Входные данные завершаются набором с  $m = n = 0$ .

### Формат выходных данных

Для каждого из наборов входных данных выведите в первой строке максимальное количество чисел  $M$ , которые может взять Саша. Во второй строке выведите один из возможных вариантов Сашиних действий — строку из символов R, D и T, где R означает ход вправо ( $x \leftarrow x + 1$ ), D обозначает ход вниз ( $y \leftarrow y + 1$ ) и T обозначает взятие числа, на котором в данный момент стоит фишка.

Разделяйте вывод для разных наборов входных данных одной пустой строкой.

### Пример

<i>стандартный ввод</i>	<i>стандартный вывод</i>
2 2	1
1 1	TRD
1 1	
2 3	3
1 2	TRTDDT
1 1	
1 3	
0 0	