

## Задача А. Транзитивное замыкание

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 0.25 секунд  
Ограничение по памяти: 256 мегабайт

Дан ориентированный граф. Найдите его транзитивное замыкание, то есть для каждой пары вершин  $a, b$  определите, есть ли путь из  $a$  в  $b$ .

### Формат входных данных

На первой строке число вершин  $n$  ( $1 \leq n \leq 1\,000$ ). Следующие  $n$  строк имеют длину  $n$ , состоят из нулей и единиц и задают матрицу смежности графа. Единица в  $i$ -й строке,  $j$ -м столбце обозначает ребро из  $i$  в  $j$ .

### Формат выходных данных

Выведите матрицу смежности транзитивного замыкания данного графа.

### Пример

стандартный ввод	стандартный вывод
3	011
010	001
001	000
000	

## Задача В. Mumatrix

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 0.6 секунд  
Ограничение по памяти: 256 мегабайт

Дана бинарная матрица. Возвести ее в квадрат по модулю 2.

### Формат входных данных

Во входном файле число  $n$  ( $1 \leq n \leq 4000$ ). Далее  $n$  строк. В каждой  $n$  символов. Каждый — или 0, или 1.

### Формат выходных данных

В выходной одно число — количество единиц в матрице-квадрате-исходной.

### Пример

стандартный ввод	стандартный вывод
3 011 100 101	5

## Задача С. Украденный массив

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 0.8 секунд  
Ограничение по памяти: 256 мегабайт

Мальчик Петя любит массивы. Недавно ему подарили огромный массив чисел  $F$  размера  $2^n$ . Петя человек странный, и при виде массива из чисел сразу начинает считать какие-то суммы на нём. Специально для этого он купил в магазине новый пустой массив  $P$  размера  $2^n$  и начал его заполнять по следующему правилу:  $P[i] = \sum_{j \& i = j} F[j]$ . Другими словами для каждого  $j$ , такого что  $j$  — подмаска  $i$  (т.е. побитовое «И» чисел  $i$  и  $j$  равно  $j$ ), Петя прибавил  $F[j]$  к изначально нулевому значению  $P[i]$ .

Но потом случилось ужасное — массив  $F$  украли! Теперь Петя хочет разыскать массив  $F$ , но он не помнит, какие значения там были изначально. Единственное что у него есть — массив  $P$ . Помогите Пете восстановить массив  $F$ .

### Формат входных данных

В первой строке входных данных дано одно число  $n$  ( $1 \leq n \leq 20$ ).

В следующей строке даны  $2^n$  чисел,  $i$ -е из них — значение  $P[i]$  ( $-10^9 \leq P[i] \leq 10^9$ ) (нумерация ведётся с нуля).

### Формат выходных данных

В одной строке выведите  $2^n$  чисел,  $i$ -е из которых — значение  $F[i]$  (нумерация ведётся с нуля).

### Примеры

стандартный ввод	стандартный вывод
2 1 2 3 4	1 1 2 0
3 1 3 4 10 6 14 16 36	1 2 3 4 5 6 7 8

## Задача D. Переворот битов

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 1 секунда  
Ограничение по памяти: 256 мегабайт

Дана таблица  $n \times m$  из 0 и 1. За одно действие вы можете поменять все биты на противоположные (то есть все 0 на 1 и все 1 на 0) в любом столбце или в любой строке этой таблицы. Вы можете совершить  $k$  действий. Какое максимальное количество 1 вы можете получить?

### Формат входных данных

Первая строка содержит три целых числа  $n$ ,  $m$  и  $k$  ( $n \times m \leq 500$ ,  $k \leq 1000$ ).

Следующие  $n$  строк содержат описание таблицы: в  $i$ -й из этих строк содержится битовая строка длины  $m$ .

### Формат выходных данных

Выведите единственное число — ответ на задачу.

### Примеры

стандартный ввод	стандартный вывод
3 2 2 11 10 11	5
5 4 4 0011 1100 0001 0101 0010	16

## Задача Е. Симпатичные узоры

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 64 мегабайта

Компания BrokenTiles планирует заняться выкладыванием во дворах у состоятельных клиентов узор из черных и белых плиток, каждая из которых имеет размер  $1 \times 1$  метр. Известно, что дворы всех состоятельных людей имеют наиболее модную на сегодня форму прямоугольника  $M \times N$  метров.

Однако при составлении финансового плана у директора этой организации появилось целых две серьезных проблемы: во первых, каждый новый клиент очевидно захочет, чтобы узор, выложенный у него во дворе, отличался от узоров всех остальных клиентов этой фирмы, а во вторых, этот узор должен быть симпатичным.

Как показало исследование, узор является симпатичным, если в нем нигде не встречается квадрата  $2 \times 2$  метра, полностью покрытого плитками одного цвета.

Для составления финансового плана директору необходимо узнать, сколько клиентов он сможет обслужить, прежде чем симпатичные узоры данного размера закончатся. Помогите ему!

### Формат входных данных

На первой строке входного файла находятся два положительных целых числа, разделенные пробелом —  $M$  и  $N$  ( $1 \leq M \times N \leq 30$ ).

### Формат выходных данных

Выведите в выходной файл единственное число — количество различных симпатичных узоров, которые можно выложить во дворе размера  $M \times N$ . Узоры, получающиеся друг из друга сдвигом, поворотом или отражением считаются различными.

### Примеры

стандартный ввод	стандартный вывод
1 1	2
1 2	4
4 1	16
2 3	50

## Задача F. Путьевые строки

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1 секунда
Ограничение по памяти:	256 мегабайт

Рассмотрим ориентированный граф  $G$ , имеющий  $n$  вершин, пронумерованных натуральными числами от 1 до  $n$ . В графе  $G$  возможно наличие нескольких дуг между одной и той же парой вершин, а также дуг, ведущих из вершины в нее саму. Каждая дуга графа помечена некоторой буквой латинского алфавита. Каждому пути в графе  $G$  можно поставить в соответствии строку, состоящую из букв, написанных на последовательно проходимых в соответствии с этим путем дугах. Эта строка называется путевой меткой пути. Назовем строку  $S$  путевой строкой графа  $G$ , если в нем существует путь, путевая метка которого равна  $S$ .

Ваша задача посчитать остаток от деления на 1 000 000 количества путевых строк графа  $G$ , состоящих ровно из  $L$  символов.

### Формат входных данных

В первой строке входных данных записаны целые числа  $n, m, L$  ( $1 \leq n \leq 10$ ,  $1 \leq m \leq 10\,000$ ,  $1 \leq L \leq 100$ ), равные количеству вершин и ребер графа  $G$ , а также длине путевых строк, которые нужно искать. Следующие  $m$  строк задают дуги графа  $G$ . Каждая из этих строк содержит два натуральных числа  $a, b$  ( $1 \leq a, b \leq n$ ) и маленькую латинскую букву  $c$ , что означает наличие дуги из вершины  $a$  в вершину  $b$ , помеченной символом  $c$ . Элементы каждой строки отделены друг от друга пробелами.

### Формат выходных данных

Единственная строка выходных данных должна содержать одно число, равное остатку от деления количества путевых строк длины  $L$  в графе  $G$  на 1 000 000.

### Пример

стандартный ввод	стандартный вывод
4 4 100 1 2 a 2 3 b 3 4 a 4 1 b	2

## Задача G. Свёртка

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1 секунда
Ограничение по памяти:	256 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

стандартный ввод	стандартный вывод
3 2 30 239017	2723387430 3167905008
16 2 239 17	551267264 1632349120

### Замечание

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 Н. Мать драконов

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 0.3 секунд  
Ограничение по памяти: 256 мегабайт

В Королевстве Ланнистеров  $n$  замков и несколько стен, соединяющих два замка, никакие два замка не соединены более, чем одной стеной, ни одна стена не соединяет замок с собой.

Сир Джейме Ланнистер узнал, что Дейенерис Таргариен собирается атаковать его королевство. Он хочет защитить свои владения. У него есть  $k$  литров странной жидкости. Он хочет распределить эту жидкость между замками так, чтобы каждый замок содержал некоторое количество жидкости (возможно, нулевое или нецелое количество литров). После этого стабильность стены, соединяющей замки  $a$  и  $b$ , содержащие  $x$  и  $y$  литров жидкости, соответственно, равна  $x \cdot y$ .

Ваша задача — найти максимальную возможную сумму стабильностей стен, которую Сир Джейме Ланнистер сможет достичь

### Формат входных данных

Первая строка содержит два целых числа  $n$  и  $k$  ( $1 \leq n \leq 40$ ,  $1 \leq k \leq 1000$ ).

Далее следует  $n$  строк. В  $i$ -й из них содержится  $n$  целых чисел  $a_{i,1}, a_{i,2}, \dots, a_{i,n}$  ( $a_{i,j} \in \{0, 1\}$ ). Если замки  $i$  и  $j$  соединены стеной,  $a_{i,j} = 1$ . В противном случае оно равно 0.

Гарантируется, что  $a_{i,j} = a_{j,i}$  и  $a_{i,i} = 0$  для всех  $1 \leq i, j \leq n$ .

### Формат выходных данных

Выведите одно число — максимальную возможную сумму стабильностей стен, которую Сир Джейме Ланнистер сможет достичь.

Ваш ответ будет считаться правильным, если его абсолютная или относительная точность не превосходит  $10^{-6}$ .

А именно, если ваш ответ равен  $a$ , а ответ жюри равен  $b$ , то ваш ответ будет зачтен, если  $\frac{|a-b|}{\max(1,b)} \leq 10^{-6}$ .

### Примеры

стандартный ввод	стандартный вывод
3 1 0 1 0 1 0 0 0 0 0	0.250000000000
4 4 0 1 0 1 1 0 1 0 0 1 0 1 1 0 1 0	4.000000000000

### Замечание

В первом примере, если замки 1, 2, 3 содержат 0.5, 0.5, 0 литров жидкости, соответственно, ответ равен 0.25.

Во втором примере, если замки 1, 2, 3, 4 содержат 1.0, 1.0, 1.0, 1.0 литров жидкости, ответ равен 4.0.



## Задача I. Два капитана

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1 секунда
Ограничение по памяти:	256 мегабайт

Как известно, у Черной Жемчужины два капитана: капитан Джек Воробей и Барбосса. Корабль содержит ровно  $n$  пушек, расположенных в ряд. Во время боя оба капитана раз в минуту одновременно дают команды своим матросам. Команды бывают следующих видов:

- `send l r` — послать своих матросов стрелять из пушек с номерами от  $l$  до  $r$  включительно
- `back l r` — отозвать всех своих матросов с пушек с номерами от  $l$  до  $r$  включительно. Если на каких-то пушках из этого отрезка нет матросов, подчиняющихся этому капитану, то с такими пушками ничего не происходит
- `gun` — принести еще одну бутылку рома

Каждая команда выполняется мгновенно, после чего сражение идет еще минуту до следующей команды. Если в какой-то момент у одной и той же пушки окажутся матросы, подчиняющиеся разным капитанам, они подерутся и убьют друг друга. Эта ситуация не устраивает никого из капитанов, и поэтому они обратились к вам с просьбой помочь им в решении этой проблемы.

Перед началом очередного сражения капитан Джек Воробей и Барбосса составили планы своих действий. Известно, что план капитана Джека Воробья состоит из  $m_1$  команд, а план Барбоссы — из  $m_2$  команд. В начале  $i$ -ой минуты боя каждый капитан дает своим матросам  $i$ -ую команду из своего плана, если в нем есть хотя бы  $i$  команд. Вам поручили исправить планы так, чтобы все матросы остались живы. Единственная доступная вам модификация плана сражения — вставка нескольких команд `gun` в любые места. Понятно, что капитаны не очень любят менять свои планы, поэтому суммарное количество команд, добавленных Вами в оба плана, должно быть минимально.

### Формат входных данных

В первой строке дано число  $n$  — количество пушек на корабле ( $1 \leq n \leq 10^9$ ).

Во второй строке задано число  $m_1$  — количество команд в плане Джека Воробья ( $1 \leq m_1 \leq 3000$ ). В следующих  $m_1$  строках перечислены команды из плана Джека Воробья. Команды заданы так, как они описаны выше. Для всех команд, использующих  $l$  и  $r$ , верно, что  $1 \leq l \leq r \leq n$ . Гарантируется, что последняя команда в плане — `back 1 n`.

Во следующей строке задано число  $m_2$  — количество команд в плане Барбоссы ( $1 \leq m_2 \leq 3000$ ). В следующих  $m_2$  строках перечислены команды из плана Барбоссы. Команды заданы так, как они описаны выше. Для всех команд, использующих  $l$  и  $r$ , верно, что  $1 \leq l \leq r \leq n$ . Гарантируется, что последняя команда в плане — `back 1 n`.

### Формат выходных данных

В единственной строке выведете минимальное количество дополнительных команд.

### Система оценки

Первая группа тестов стоит 20 баллов. Баллы за эту группу начисляются только при прохождении всех тестов группы. Для всех тестов этой группы выполнено условие,  $(n, m_1, m_2 \leq 20)$ .

Вторая группа тестов стоит 35 баллов. Баллы за эту группу начисляются только при прохождении всех тестов группы. Для всех тестов этой группы выполнено условие,  $(n, m_1, m_2 \leq 300)$ .

Третья группа тестов стоит 25 баллов. Баллы за эту группу начисляются только при прохождении всех тестов группы. Для всех тестов этой группы выполнено условие,  $(n, m_1, m_2 \leq 1000)$ .

Четвертая группа тестов стоит 20 баллов. Каждый тест этой группы стоит определенное количество баллов.

## Пример

стандартный ввод	стандартный вывод
3	3
4	
send 1 1	
send 2 2	
back 1 1	
back 1 3	
5	
send 2 3	
send 1 1	
back 2 2	
run	
back 1 3	

## Задача J. Ним

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

Сергей Х. и Сергей Б. собираются поиграть в Ним. Сначала они выставляют  $k$  стопок камней, содержащие  $a_1, a_2, \dots, a_k$  камней соответственно. Затем они по очереди ходят, причем Сергей Х. начинает первым.

В свой ход игрок выбирает любую непустую кучку и забирает оттуда любое ненулевое количество камней, если на момент хода все кучки пустые, то у игрока нет хода, и он проигрывает.

Из особой любви к простым числам, они решили сделать каждое  $a_i$  от 0 до  $A$ , где  $a_i$  — простое число. По данным  $k$  и  $A$  найдите число начальных состояний, дающих победу Сергею Б., если оба игрока играют оптимально. Выведите ответ по модулю  $10^9 + 7$ .

### Формат входных данных

Единственная строка входных данных содержит два числа:  $k$  и  $A$  ( $1 \leq k \leq 10^9$ ,  $2 \leq A \leq 5 \cdot 10^4$ ).

### Формат выходных данных

Выведите по модулю  $10^9 + 7$  число начальных состояний, обеспечивающих выигрыш Сергею Б.

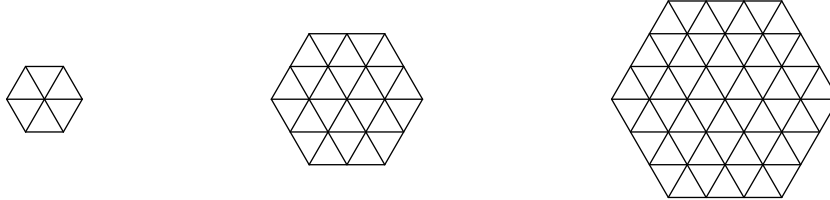
### Примеры

стандартный ввод	стандартный вывод
3 7	6
4 13	120

## Задача К. Шестиугольные домино-ромбы

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 256 мегабайт

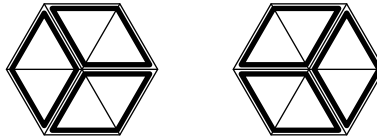
Правильный шестиугольник со стороной длины  $n$  разделен на  $6n^2$  единичных треугольников.



Его требуется полностью покрыть без наложений и пересечений домино-ромбами — фигурами, составленными из двух единичных треугольников с общей стороной.



Требуется посчитать число способов покрыть шестиугольник таким образом. Например, два способа покрыть шестиугольник со стороной 1 приведены на рисунке.



### Формат входных данных

Входной файл содержит число  $n$  ( $1 \leq n \leq 7$ ).

### Формат выходных данных

Выведите число способов покрыть шестиугольник домино-ромбами.

### Примеры

стандартный ввод	стандартный вывод
1	2
2	20

## Задача L. Необычная сортировка

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 3 секунды  
Ограничение по памяти: 256 мегабайт

Вам дана последовательность различных целых чисел  $a_1, a_2, \dots, a_n$ . Все числа в последовательности лежат в отрезке  $[0, 2^k - 1]$ , где  $k$  дано.

Давайте определим функцию  $f(x)$  для числа  $x$ , которое лежит в отрезке  $[0, 2^k - 1]$  как количество инверсий в последовательности  $a_1 \oplus x, a_2 \oplus x, \dots, a_n \oplus x$ .

Теперь давайте отсортируем все целые числа из отрезка  $[0, 2^k - 1]$  по возрастанию значений функции  $f$ , а затем по возрастанию самих чисел.

Вам дана позиция  $p$ . Найдите  $p$ -е число в этом порядке сортировки всех целых чисел из отрезка  $[0, 2^k - 1]$ .

### Формат входных данных

В первой строке находится единственное целое число  $t$  ( $1 \leq t \leq 20$ ) — количество тестовых случаев. Далее находится описание  $t$  тестовых случаев в следующем формате:

Первая строка содержит три целых числа  $n, k, p$ , разделенных пробелом ( $1 \leq n \leq 10^6, 1 \leq k \leq 30, 1 \leq p \leq 2^k$ ) — количество чисел в последовательности, параметр  $k$  и заданная позиция.

В следующей строке находятся  $n$  различных целых чисел  $a_1, a_2, \dots, a_n$ , разделенных пробелами ( $0 \leq a_i < 2^k$ ).

Сумма всех  $n$  во всех тестовых случаях не превосходит  $10^6$ .

### Формат выходных данных

Для каждого тестового случая выведите единственное целое число —  $p$ -е целое число в порядке сортировки всех целых чисел из отрезка  $[0, 2^k - 1]$ , описанном в условии задачи.

### Пример

стандартный ввод	стандартный вывод
2	4
4 3 5	2
2 0 3 7	
2 2 1	
2 0	

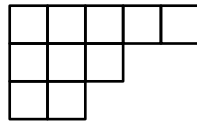
## Задача М. Увидеть Юнга и умереть

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	2 секунды
Ограничение по памяти:	256 мегабайт

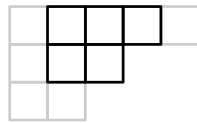
Диаграммы Юнга используются для того, чтобы изобразить разбиение числа на слагаемые. Разбиение числа  $n$  на слагаемые представляет собой сумму вида  $n = m_1 + m_2 + \dots + m_k$ , где  $m_1 \geq m_2 \geq \dots \geq m_k$ .

Диаграмма состоит из  $n$  квадратиков, организованных в виде  $k$  рядов, где  $k$  количество слагаемых в разбиении. Ряд, соответствующий числу  $m_i$ , содержит  $m_i$  квадратиков. Все ряды выровнены по левому краю и упорядочены от более длинного к более короткому.

Например, диаграмма Юнга, приведенная на рисунке, соответствует разбиению  $10 = 5 + 3 + 2$ .



Иногда можно вписать одну диаграмму Юнга в другую. Диаграмму  $X$  можно вписать в диаграмму  $Y$ , если можно удалить некоторые квадратики из диаграммы  $Y$  так, чтобы получилась диаграмма  $X$ . Отметим, что разрешается только удалять некоторые квадратики, вращать или отражать диаграмму не разрешается. Например, диаграмма для разбиения  $5 = 3 + 2$  может быть вписана в диаграмму для разбиения  $10 = 5 + 3 + 2$ , как показано на рисунке.



С другой стороны, диаграмму для разбиения  $8 = 4 + 4$  нельзя вписать в диаграмму для разбиения  $10 = 5 + 3 + 2$ .

Для заданного  $n$  найдите такое разбиение  $n$  на слагаемые, что в соответствующую ему диаграмму Юнга можно вписать максимальное количество различных диаграмм.

Например, в диаграмму для разбиения  $10 = 5 + 3 + 2$  можно вписать 36 различных диаграмм. Однако это не максимальное значение. В диаграмму для разбиения  $10 = 4 + 2 + 2 + 1 + 1$  можно вписать 41 диаграмму Юнга.

### Формат входных данных

Входной файл содержит целое число  $n$  ( $1 \leq n \leq 100$ ).

### Формат выходных данных

На первой строке выходного файла выведите максимальное число диаграмм Юнга, которые можно вписать в некоторую диаграмму, соответствующую разбиению на слагаемые числа  $n$ .

На второй строке выведите одно или более целых чисел — количество квадратиков в каждом из рядов оптимальной диаграммы.

### Пример

стандартный ввод	стандартный вывод
10	41 4 3 2 1

## Задача N. Максимумы

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 6 секунд  
Ограничение по памяти: 512 мегабайт

К сожалению, у Дани не хватило времени, чтобы написать нескучное условие к этой задаче.

Вам дан массив  $a_0, \dots, a_{n-1}$ , который задаётся числами  $n, a_0, b, c, d$  следующим образом:

$$b_i = (b_{i-1} \cdot c + d) \bmod 2^{31} \text{ для } i \geq 1$$

$$a_i = (a_{i-1} + 1 - 2 \cdot ((b_i \bmod 239179) \bmod 2)) \text{ для } i \geq 1$$

Обратите внимание, что два соседних числа отличаются либо на  $+1$ , либо на  $-1$ .

Ответьте на  $n$  запросов,  $i$ -й запрос — максимум на отрезке  $[\min(l_i, r_i), \max(l_i, r_i)]$  для  $i = 0 \dots n-1$ .

Пусть  $ans_i$  — ответ на  $i$ -й запрос. Будем считать, что  $ans_{-1} = 0$ . Вам задаётся число  $x_0$ . Далее,  $l_i$  и  $r_i, x_i$  вычисляются так:

$$l_i = (x_i + ans_{i-1}) \bmod n \text{ для } i \geq 0$$

$$r_i = (l_i + i) \bmod n \text{ для } i \geq 0$$

$$x_i = (x_{i-1} \cdot 1103515245 + 12345) \bmod 2^{31} \text{ для } i \geq 1$$

При взятии по модулю обратите внимание на то, что  $ans_i$  бывают отрицательными.

### Формат входных данных

Единственная строка ввода содержит шесть целых чисел  $n, a_0, b, c, d, x_0$ .

- $1 \leq n \leq 2 \cdot 10^7$
- $-10^9 \leq a_0 \leq 10^9$
- $0 \leq b, c, d, x_0 \leq 2^{31} - 1$

### Формат выходных данных

Выведите сумму ответов на все запросы.

### Примеры

стандартный ввод	стандартный вывод
4 1 2 3 4 0	11
100500 -1 23 45 67 89	-8614564

## Задача О. Максимумы возвращаются

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 6 секунд  
Ограничение по памяти: 512 мегабайт

Вам дан массив  $a_0, \dots, a_{n-1}$ , который задаётся числами  $n, a_0, b_0, c, d$  следующим образом:

$$b_i = (b_{i-1} \cdot c + d) \bmod 2^{31} \text{ для } i \geq 1$$

$$a_i = (a_{i-1} + 1 - 2 \cdot ((b_i \bmod 239179) \bmod 2)) \text{ для } i \geq 1$$

Обратите внимание, что два соседних числа отличаются либо на  $+1$ , либо на  $-1$ .

Ответьте на  $n$  запросов,  $i$ -й запрос — поиск количества максимумов на отрезке  $[\min(l_i, r_i), \max(l_i, r_i)]$  для  $i = 0 \dots n - 1$ . Пусть  $ans_i$  — ответ на  $i$ -й запрос. Будем считать, что  $ans_{-1} = 0$ . Вам задаётся число  $x_0$ . Далее,  $l_i$  и  $r_i, x_i$  вычисляются так:

$$l_i = (x_i + ans_{i-1}) \bmod n \text{ для } i \geq 0$$

$$r_i = (l + i) \bmod n \text{ для } i \geq 0$$

$$x_i = (x_{i-1} \cdot 1103515245 + 12345) \bmod 2^{31} \text{ для } i \geq 1$$

### Формат входных данных

Единственная строка ввода содержит шесть целых чисел  $n, a_0, b, c, d, x_0$ .

- $1 \leq n \leq 2 \cdot 10^7$
- $-10^9 \leq a_0 \leq 10^9$
- $0 \leq b, c, d, x_0 \leq 2^{31} - 1$

### Формат выходных данных

Выведите сумму ответов на все запросы.

### Примеры

стандартный ввод	стандартный вывод
4 1 2 3 4 0	4
100500 -1 23 45 67 89	173287



## Задача Р. Максимумы наносят ответный удар

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 15 секунд  
Ограничение по памяти: 512 мегабайт

Вам дан массив  $a_0, \dots, a_{n-1}$ , который задаётся числами  $n, a_0, c, d$  следующим образом:

$$a_i = (a_{i-1} \cdot c + d) \bmod 2^{30} \text{ для } i \geq 1$$

Ответьте на  $n$  запросов,  $i$ -й запрос — максимум на отрезке  $[\min(l_i, r_i), \max(l_i, r_i)]$  для  $i = 0 \dots n-1$ .

Пусть  $ans_i$  — ответ на  $i$ -й запрос. Будем считать, что  $ans_{-1} = 0$ . Вам задаётся число  $x_0$ . Далее,  $l_i$  и  $r_i, x_i$  вычисляются так:

$$l_i = (x_i + ans_{i-1}) \bmod n \text{ для } i \geq 0$$

$$r_i = (l_i + i) \bmod n \text{ для } i \geq 0$$

$$x_i = (x_{i-1} \cdot 1103515245 + 12345) \bmod 2^{30} \text{ для } i \geq 1.$$

### Формат входных данных

Единственная строка ввода содержит шесть целых чисел  $n, a_0, c, d, x_0$ .

- $1 \leq n \leq 5 \cdot 10^7$
- $0 \leq a_0, c, d, x_0 \leq 2^{30} - 1$

### Формат выходных данных

Выведите сумму ответов на все запросы.

### Примеры

стандартный ввод	стандартный вывод
4 2 3 4 0	152
100500 23 45 67 89	107890041652944

## Задача Q. Xor-convolution

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 3 секунды  
Ограничение по памяти: 256 мегабайт

Даны два массива длиной  $2^k$ . Посчитайте их хог-свертку по модулю  $1e9+7$ .

### Формат входных данных

На первой строке дано одно число  $k$  ( $0 \leq k \leq 20$ ). На второй строке дан первый массив  $a_i$  ( $0 \leq a_i \leq 1e9+6, 0 \leq i < 2^k$ ). На второй строке дан второй массив  $b_i$  ( $0 \leq a_i \leq 1e9+6, 0 \leq i < 2^k$ ).

### Формат выходных данных

Выведите  $2^k$  чисел  $c_i$  ( $0 \leq c_i \leq 1e9+6$ ) — хог-свертку двух массивов.

### Пример

стандартный ввод	стандартный вывод
1 0 1 2 3	3 2

## Задача R. Тримино

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 3 секунды  
Ограничение по памяти: 256 мегабайт

Недавно Глеб рассыпал свои домашние игрушки. Чтобы остаться незамеченным, он решил собрать их обратно в коробку. Коробка имеет размеры  $n$  на  $m$ . Все игрушки Глеба можно представить в виде уголка из трех клеток, либо в виде трех клеток, расположенных вдоль прямой. Вам требуется выяснить, можно ли собрать все игрушки обратно в коробку так, чтобы вся коробка была покрыта игрушками и никакие две игрушки не пересекались. Дома все игрушки можно переворачивать.

### Формат входных данных

В единственной строке записаны 4 целых числа  $n, m, a, b$  ( $1 \leq n \cdot m \leq 100$ ;  $0 \leq a \leq 100$ ;  $0 \leq b \leq 100$ ) — размеры коробки и количество игрушек типа «прямая» и «уголок» соответственно.

### Формат выходных данных

Если замостить коробку данным количеством игрушек возможно, в единственной строке выведите «YES» (без кавычек).

Иначе в единственной строке выведите «NO» (без кавычек).

### Примеры

стандартный ввод	стандартный вывод
2 6 2 2	YES
6 3 1 5	NO