

Задача A. Grid Coloring

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

У Snuke есть табличка с H рядами и W столбцами. Snuke хочет покрасить эту табличку в цвета $1, 2, \dots, N$, чтобы выполнялись следующие условия:

- Для каждого i ($1 \leq i \leq N$), существует ровно a_i клеток покрашенных в цвет i . Гарантируется, что $a_1 + a_2 + \dots + a_N = H \cdot W$.
- Для каждого i ($1 \leq i \leq N$), все клетки цвета i связаны. Это значит, что из каждой клетки цвета i мы можем добраться до любой клетки цвета i последовательно перемещаясь в соседнюю по стороне клетку цвета i .

Найдите способ покрасить таблицу, чтобы условия выполнялись. Можно показать, что решение всегда существует.

Формат входных данных

Первая строка ввода содержит 2 целых числа H и W ($1 \leq H, W \leq 100$)
Вторая строка содержит одно целое число N ($1 \leq N \leq H \cdot W$)
Третья строка содержит N чисел a_i ($1 \leq a_i$) ($\sum_{i=1}^N a_i = H \cdot W$)

Формат выходных данных

Выведите способ покрасить табличку, который удовлетворяет условиям.
Вывод должен содержать H строк, в каждой из которой должно быть W чисел.

Примеры

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

Задача В. Красивый путь

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

В стране Деда Мороза есть n домов и m двусторонних дорог между ними. Дед Мороз захотел проехать по магазинам, чтобы купить себе новогодние подарки. К сожалению, олени Деда Мороза очень требовательные, поэтому они согласны помочь ему, если путь будет удовлетворять следующим условиям:

- Путь проходит через каждый дом не более одного раза.
- Путь проходит через каждую дорогу не более одного раза.
- Путь содержит хотя бы 2 дома.
- Путь проходит через каждый дом, который соединен дорогой с началом или концом пути.

К счастью, олени знают рамки и не просят невозможного, так что такой путь всегда существует. Дед Мороз просит вас найти любой путь, удовлетворяющий требованиям оленей. Обратите внимание, что Дед Мороз сам может выбрать, где он начнет покупки, и где закончит.

Гарантируется, что в стране Деда Мороза от каждого дома по дорогам можно добраться до любого другого дома. Между двумя домами существует не более одной дороги.

Формат входных данных

В первой строке находятся 2 целых числа n, m ($2 \leq n \leq 10^5, 1 \leq m \leq 10^5$) — количество домов и дорог соответственно.

В следующих m строках находятся по 2 целых числа a_i, b_i ($1 \leq a_i < b_i \leq n$) — описание дорог страны.

Дорога с номером i соединяет дома a_i и b_i .

Формат выходных данных

В первой строке выведите одно целое число — количество домов на пути.

Во второй строке через пробел выведите номера домов в порядке их появления на пути.

Примеры

стандартный ввод	стандартный вывод
5 6 1 3 1 4 2 3 1 5 3 5 2 4	4 2 3 1 4
7 8 1 2 2 3 3 4 4 5 5 6 6 7 3 5 2 6	7 1 2 3 4 5 6 7

Задача С. Лосяш и дерево

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

В одной научной статье Лосяш увидел необычное дерево из N вершин, в каждой вершине записано одно число — 0 или 1.

Изучая интересные свойства этого дерева, Лосяш задался целью найти такой путь в этом дереве, что xor значений в вершинах, лежащих на этом пути равен xor значений в вершинах, которые являются соседними к какому-либо из концов пути, но не лежат на пути.

Помогите ему найти его жизненный путь!

Обратите внимание, что не запрещается выводить путь, концы которого совпадают.

Формат входных данных

В первой строке вводится целое число N ($2 \leq N \leq 200\,000$).

В следующей строке содержатся N целых чисел a_1, \dots, a_N — значения, записанные в вершинах ($a_i = 0$ или 1)

В каждой из следующих $N - 1$ строк вводится два числа u_i, v_i — очередное ребро дерева ($1 \leq u_i, v_i \leq N$).

Формат выходных данных

Если подходящего пути не существует, то выведите -1 , иначе первая строка должна содержать одно целое число M ($1 \leq M \leq N$) — количество вершин в пути.

В следующей строке должны быть записаны M целых чисел — вершины в пути в порядке обхода ($1 \leq v_i \leq N$)

Примеры

стандартный ввод	стандартный вывод
4 0 1 1 1 1 2 1 3 1 4	3 2 1 3
2 0 1 1 2	-1
4 1 0 0 1 2 1 3 2 4 2	1 2

Задача D. Интересная сумма

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

Дан массив a длины n . Вы можете выбрать любой подотрезок a_l, a_{l+1}, \dots, a_r массива длины, не совпадающий со всем массивом, то есть, для которого $1 \leq l \leq r \leq n$ и $r - l + 1 < n$. Красотой выбранного подотрезка назовем значение

$$\max(a_1, a_2, \dots, a_{l-1}, a_{r+1}, a_{r+2}, \dots, a_n) - \min(a_1, a_2, \dots, a_{l-1}, a_{r+1}, a_{r+2}, \dots, a_n) + \max(a_l, \dots, a_r) - \min(a_l, \dots, a_r).$$

Найдите максимальную красоту подотрезка среди всех возможных допустимых подотрезков массива (за исключением всего массива).

Формат входных данных

Первая строка содержит одно целое число t ($1 \leq t \leq 1000$) — количество наборов входных данных. Далее следует описание наборов входных данных.

Первая строка каждого набора входных данных содержит единственное целое число n ($4 \leq n \leq 10^5$) — длину массива.

Вторая строка каждого набора входных данных содержит n целых чисел a_1, a_2, \dots, a_n ($1 \leq a_i \leq 10^9$) — элементы данного массива.

Гарантируется, что сумма n по всем наборам входных данных не превосходит 10^5 .

Формат выходных данных

Для каждого набора входных данных выведите одно число — искомое максимальное значение красоты.

Пример

стандартный ввод	стандартный вывод
4	9
8	297
1 2 2 3 1 5 6 1	0
5	14
1 2 3 100 200	
4	
3 3 3 3	
6	
7 8 3 1 1 8	

Замечание

В первом тесте из условия оптимально выбрать отрезок $l = 7, r = 8$. Красота этого отрезка равна $(6 - 1) + (5 - 1) = 9$.

Во втором тесте из условия оптимально выбрать отрезок $l = 2, r = 4$. Красота этого отрезка равна $(100 - 2) + (200 - 1) = 297$.

Задача E. Минимальная уникальная подстрока

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

Пусть у вас есть строка s из символов “0” и “1”. Будем называть строку t подстрокой строки s , если существует $1 \leq l \leq |s| - |t| + 1$, такое что $t = s_l s_{l+1} \dots s_{l+|t|-1}$. Будем называть подстроку t строки s уникальной, если существует единственное такое l .

Например, пусть $s = \text{“1010111”}$. Тогда $t = \text{“010”}$ является уникальной подстрокой s , так как существует единственное подходящее $l = 2$. Заметим, что $t = \text{“10”}$ не является уникальной подстрокой s , так как подходят $l = 1$ и $l = 3$. А, например, $t = \text{“00”}$ вообще не является подстрокой строки s , так как не существует подходящих l .

Сегодня Вася на уроке информатики решал такую задачу: дана строка из символов “0” и “1”, надо найти длину её кратчайшей уникальной подстроки. Написав решение к этой задаче, он решил его протестировать. Он просит помощи у вас.

Вам даны 2 таких целых положительных числа n и k , что $(n \bmod 2) = (k \bmod 2)$, где $(x \bmod 2)$ — это операция взятия остатка числа x при делении на 2. Найдите любую строку s состоящую из n символов “0” и “1”, такую что наименьшая длина её уникальной подстроки равна k .

Формат входных данных

В первой строке даны два целых числа n и k , разделённые пробелом ($1 \leq k \leq n \leq 100\,000$, $(k \bmod 2) = (n \bmod 2)$).

Формат выходных данных

Выведите строку s длины n , состоящую из символов “0” и “1”. Минимальная длина уникальной подстроки s должна равняться k . Среди таких строк разрешается вывести **любую**. Гарантируется, что хотя бы одна подходящая строка существует.

Примеры

стандартный ввод	стандартный вывод
4 4	1111
5 3	01010
7 3	1011011

Замечание

В первом тесте легко видеть, что единственной уникальной подстрокой строки $s = \text{“1111”}$ является вся строка s , длина которой 4.

Во втором тесте у строки $s = \text{“01010”}$ минимальной по длине уникальной подстрокой является строка $t = \text{“101”}$ длина которой 3.

Во третьем тесте у строки $s = \text{“1011011”}$ минимальной по длине уникальной подстрокой является строка $t = \text{“110”}$ длина которой 3.

Задача F. Игра со стеком

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

У Вани есть натуральное число n . Он придумал следующую игру. У него есть стек, в котором изначально лежит единственное число 1. Он может выполнять следующую операцию: пусть сейчас в стеке лежит s целых чисел a_1, a_2, \dots, a_s , в порядке от самого первого элемента стека к самому последнему. Ваня может выбрать два целых числа $1 \leq l \leq r \leq s$ и положить число $a_l + a_{l+1} + \dots + a_r$ в конец стека.

Ваня хочет сделать последнее число в стеке равным n . Определите минимальное количество операций, которое для этого требуется и постройте любой возможный пример с этим количеством операций.

Формат входных данных

В первой строке находится единственное целое число t ($1 \leq t \leq 1000$) — количество тестовых случаев.

В следующих t строках находится по одному целому числу n ($1 \leq n \leq 10^{18}$) — число Вани.

Формат выходных данных

Выведите ответы на тестовые случаи в порядке их следования во входных данных. Для тестового случая выведите ответ в следующем формате:

В первой строке выведите единственное целое число q ($0 \leq q \leq 1000$) — минимальное количество операций со стеком, которое нужно сделать, чтобы последнее число стало равно n . Гарантируется, что за ≤ 1000 операций возможно сделать последнее число равным n . В следующих строках выведите по два целых числа — описания операций. В i -й строке выведите два целых числа l_i, r_i ($1 \leq l_i \leq r_i \leq i$). Во время i -й операции в конец стека кладется число $a_{l_i} + a_{l_i+1} + \dots + a_{r_i}$. После выполнения предоставленных операций последнее число стека должно быть равно n .

Все тестовые случаи независимы.

Пример

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

Задача G. Восстановление дерева

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1 секунда
Ограничение по памяти:	512 мегабайт

Fishingprince любит деревья. Дерево — это связный неориентированный граф без циклов.

У Fishingprince'a есть дерево на n вершинах. Его вершины пронумерованы числами от 1 до n . Пусть $d(x, y)$ равно длине кратчайшего расстояния между вершинами x и y , если считать длину каждого ребра равной 1.

К сожалению, Fishingprince потерял своё дерево. Тем не менее какая-то информация о нём сохранилась. А именно, для каждой тройки чисел x, y, z ($1 \leq x < y \leq n$, $1 \leq z \leq n$) известно, выполнено ли равенство $d(x, z) = d(y, z)$.

Помогите ему восстановить структуру дерева или сообщите, что дерева, удовлетворяющего всем ограничениям, не существует.

Формат входных данных

Во входных данных находятся несколько наборов входных данных. В первой строке находится одно целое число t ($1 \leq t \leq 200$) — количество наборов входных данных. Далее следуют наборы входных данных.

Первая строка набора входных данных содержит целое число n ($2 \leq n \leq 100$) — количество вершин в дереве.

Далее следует $n - 1$ строк. Из этих $n - 1$ строк i -я строка содержит $n - i$ последовательностей символов длины n , состоящих из 0 и 1. Если k -й символ в j -й последовательности i -й строки равен 0, то $d(i, k) \neq d(i + j, k)$; если же k -й символ в j -й последовательности i -й строки равен 1, то $d(i, k) = d(i + j, k)$.

Гарантируется, что в каждом тесте:

- есть не более 2 наборов входных данных с $n > 50$;
- есть не более 5 наборов входных данных с $n > 20$.

Формат выходных данных

Для каждого набора входных данных:

- выведите **No**, если искомого дерева не существует;
- иначе в первой строке выведите **Yes**. Затем выведите $n - 1$ строк. Каждая из них должна содержать два целых числа x, y ($1 \leq x, y \leq n$), означающих ребро между вершинами x и y в дереве. Если существуют несколько решений, выведите любое из них.

При выводе **Yes** и **No** вы можете выводить каждый символ в любом регистре (верхнем или нижнем).

Пример

стандартный ввод	стандартный вывод
5	Yes
2	1 2
00	No
2	Yes
10	1 3
3	2 3
001 000	No
000	Yes
3	1 2
001 010	1 4
000	2 3
5	2 5
00000 01001 00000 01100	
00000 10000 00000	
00000 11010	
00000	

Задача Н. Подарки от Деда Ахмеда

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

В Школе Деда Ахмеда обучается $n + 1$ ученик. Ученики разбиты на k классов, и в i -м классе обучается s_i учеников. Таким образом, $s_1 + s_2 + \dots + s_k = n + 1$.

В честь скорого Дня дурака все ученики получают подарки!

Дед Ахмед планировал заказать $n + 1$ коробку с подарками. В каждой коробке будет один или несколько подарков. Дед Ахмед распределит их между классами так, чтобы были выполнены следующие условия:

1. Класс номер i должен получить **ровно** s_i коробок (чтобы каждый ученик мог открыть ровно одну коробку с подарками).
2. Суммарное количество подарков в коробках, полученных i -м классом, должно быть кратным s_i (то есть поровну распределяться между s_i учениками этого класса).

К сожалению, Дед Ахмед по невнимательности заказал всего n коробок с подарками, в i -й из которых содержится a_i подарков.

У Ахмеда есть время, чтобы купить недостающую коробку с подарками, причем количество подарков в коробке должно быть целым числом от 1 до 10^6 . Помогите Ахмеду определить, коробку с каким количеством подарков ему стоит купить, а также постройте подходящее распределение коробок по классам, или сообщите, что это невозможно.

Формат входных данных

Первая строка содержит два целых числа n и k ($1 \leq n, k \leq 200, k \leq n + 1$).

Вторая строка содержит n целых чисел a_1, a_2, \dots, a_n ($1 \leq a_i \leq 10^6$) — количество подарков в имеющихся коробках.

Третья строка содержит k целых числа s_1, s_2, \dots, s_k ($1 \leq s_i \leq n + 1$) — количество учеников в классах. Гарантируется, что $\sum s_i = n + 1$.

Формат выходных данных

Если подходящего распределения не существует, в единственной строке выведите число -1 .

Иначе, в первой строке выведите единственное число s — количество подарков в коробке, которую должен купить Дед Ахмед ($1 \leq s \leq 10^6$).

Далее в k строках выведите распределение коробок по классам. В i -й строке выведите s_i целых чисел — размеры коробок, которые попадут в i -й класс.

Если существует несколько решений, выведите любое из них.

Примеры

стандартный ввод	стандартный вывод
4 2 7 7 7 127 2 3	1 7 7 7 127 1
18 4 1 2 3 4 5 6 7 8 9 1 2 3 4 5 6 7 8 9 6 1 9 3	9 7 1 7 6 5 4 9 1 2 3 8 3 2 9 8 9 6 5 4

Замечание

В первом тестовом примере Дед Ахмед может купить всего 1 подарок. После определить в первый класс две коробки с 7 подарками. $7 + 7 = 14$ нацело делится на 2. А во второй класс достанутся коробки с 1, 7, 127 подарками. $1 + 7 + 127 = 135$ нацело делится на 3.

Во втором тестовом примере классы имеют размеры 6, 1, 9 и 3. Покажем, что имеющихся коробок достаточно, чтобы распределить в классы размеров 6, 9, 3, а в класс размера 1 можно купить коробку любого размера. В класс размера 6 отправим коробки размеров 7, 1, 7, 6, 5, 4. $7 + 1 + 7 + 6 + 5 + 4 = 30$ нацело делится на 6. В класс размера 9 отправим коробки размеров 1, 2, 3, 8, 3, 2, 9, 8, 9. $1 + 2 + 3 + 8 + 3 + 2 + 9 + 8 + 9 = 45$ нацело делится на 9. В класс размера 3 отправятся оставшиеся коробки размеров 6, 5, 4. $6 + 5 + 4 = 15$ нацело делится на 3.