Олимпиада для роботов

Жюри чемпионата по скоростному вычислению булевых функций среди роботов готовит задания для участников.

Задание для роботов представляет собой таблицу из m строк и n столбцов, каждая ячейка которой содержит целое число. Обозначим число в i-й строке, j-м столбце таблицы как $x_{i,j}$. В каждом столбце значения в ячейках таблицы образую перестановку чисел от 0 до $m-1$. Иначе говоря, числа в каждом столбце различны: если $i \neq k$, то $x_{i,j} \neq x_{k,j}$ для всех j, и выполнено условие $0 \leqslant x_{i,j} < m$.

Для каждого столбца таблицы задаётся значение *порога* — целое неотрицательное число z_j от 0 до m. В качестве аргументов булевых функций, которые будут вычислять участники олимпиады, используются значения логических выражений $x_{i,j} < z_j$. Значение такого логического выражения равно 1, если неравенство выполнено, иначе оно равно 0.

В процессе соревнования участники вычисляют значения m булевых функций — по одному для каждой строки. Каждая булева функция задаётся в виде *бесповторной монотонной линейной программы* (БМЛП).

Рассмотрим БМЛП для i-й строки таблицы. Она представляет собой последовательность из $n-1$ инструкции, пронумерованных от 1 до $n-1$, p-я инструкция задаётся тремя числами: a_p, b_p и op_p. Число op_p принимает два возможных значения: 1 для операции `and` — логическое «и», 2 для операции `or` — логическое «или». Числа a_p и b_p являются номерами аргументов p-й инструкции, выполнены неравенства $1 \leqslant a_p, b_p < n+p$.

Рассмотрим массив $val[1..2n-1]$, каждое из значений которого равно 0 или 1. Проинициализируем значения $val[1]..val[n]$ с использованием порогов, $val[j] = 1$, если $x_{i,j} < z_j$, иначе $val[j] = 0$. Значение $val[n+p]$ вычисляется с использованием p-й инструкции.

- Если $op_p = 1$, то $val[n+p] = (val[a_p]$ `and` $val[b_p])$, то есть значение $val[n+p]$ равно 1 если и только если каждое из значений $val[a_p]$ и $val[b_p]$ равно 1.
- Если $op_p = 2$, то $val[n+p] = (val[a_p]$ `or` $val[b_p])$, то есть значение $val[n+p]$ равно 1 если и только если хотя бы одно из значений $val[a_p]$ и $val[b_p]$ равно 1.

При этом программа является бесповторной, а именно все $2n-2$ значений a_p и b_p для p от 1 до $n-1$ различны. Иначе говоря, $a_p \neq b_p$, а если $p \neq q$, то $a_p \neq a_q$, $a_p \neq b_q$, $b_p \neq a_q$ и $b_p \neq b_q$.

Результатом исполнения программы является значение $val[2n-1]$.

Жюри олимпиады подготовило таблицу $x_{i,j}$, выбрало булевы функции для каждой строки и записало их в виде БМЛП. Теперь осталось выбрать значение порога для каждого столбца, чтобы получить задание для олимпиады. Жюри считает задание сбалансированным, если ровно s из m программ для строк таблицы возвращают единицу, а остальные $m-s$ возвращают ноль. Ваша задача — помочь жюри найти подходящие значения порогов.

Требуется написать программу, которая по заданным значениям в ячейках таблицы и БМЛП для строк таблицы определяет такие значения порогов z_j, чтобы значение ровно s из m заданных функций было равно 1. Можно доказать, что при описанных в условии задачи ограничениях требуемые значения порогов всегда можно подобрать.

Формат входных данных

В первой строке входных данных заданы целые числа n, m и s ($1 \leqslant n \leqslant 3 \cdot 10^5$, $1 \leqslant m \leqslant 3 \cdot 10^5$, $n \cdot m \leqslant 3 \cdot 10^5$, $0 \leqslant s \leqslant m$).

Далее следует m блоков по $n-1$ строке в каждом, каждый блок задает бесповторную монотонную линейную программу для одной строки таблицы. В каждом блоке p-я строка содержит 3 целых числа: a_p, b_p и op_p ($1 \leqslant a_p < n+p$, $1 \leqslant b_p < n+p$, гарантируется, что в одном блоке все значения a_p и b_p попарно различны, $op_p = 1$ или $op_p = 2$).

Последние m строк задают таблицу, i-я строка содержит n целых чисел, j-е из которых равно $x_{i,j}$ ($0 \leqslant x_{i,j} \leqslant m-1$, в каждом столбце все числа различны, то есть если $i \neq k$, то $x_{i,j} \neq x_{k,j}$ для всех j).

Формат выходных данных

Выведите n целых чисел — искомые значения порогов $z_1, z_2, \ldots, z_n$ $(0 \leqslant z_j \leqslant m)$. Если подходящих вариантов несколько, выведите любой из них.

Система оценки

Баллы за каждую подзадачу начисляются только в случае, если все тесты для этой подзадачи и необходимых подзадач успешно пройдены.

Подз.	**Баллы**	**Дополнительные ограничения**	**Необходимые подзадачи**	**Информация о проверке**
1	10	$n \leqslant 2$, $m \leqslant 10^3$		первая ошибка
2	10	$n \leqslant 2$, $m \leqslant 10^5$	1	первая ошибка
3	10	$n \leqslant 10$, $m \leqslant 2$		первая ошибка
4	5	$x_{i,j} = i - 1$		первая ошибка
5	5	$op_p = 1$, только операции «и»		первая ошибка
6	20	$n \leqslant 100$	1, 2, 3	первая ошибка
7	10	БМЛП для всех строк одинаковые		первая ошибка
8	30	нет	1 – 7	первая ошибка

Пример

стандартный ввод	стандартный вывод
4 3 2 1 2 1 3 4 1 5 6 2 1 2 2 3 5 1 4 6 2 1 4 1 2 3 1 5 6 2 0 1 2 2 2 2 1 0 1 0 0 1	0 1 2 3

Замечание

В примере в таблице три строки, каждой соответствует формула. Необходимо найти четыре порога так, чтобы ровно две формулы возвращали 1, а оставшаяся — 0.

Рассмотрим, как будет вычисляться массив val для первой строки.

Первые четыре значения вычисляются на основе чисел в этой строке и порогов:

- $val[1] = (x_{1,1} < z_1) = (0 < 0) = 0;$
- $val[2] = (x_{1,2} < z_2) = (1 < 1) = 0;$
- $val[3] = (x_{1,3} < z_3) = (2 < 2) = 0;$
- $val[4] = (x_{1,4} < z_4) = (2 < 3) = 1.$

Далее выполняем линейную программу для первой строки:

- $val[5] = (val[1] \texttt{ and } val[2]) = (0 \texttt{ and } 0) = 0;$

- $val[6] = (val[3] \texttt{ and } val[4]) = (0 \texttt{ and } 1) = 0$;
- $val[7] = (val[5] \texttt{ or } val[6]) = (0 \texttt{ or } 0) = 0$.

Таким образом значение булевой функции для первой строки равно 0. Кстати, если эту функцию записать формулой, то получится:

$$(((x_{1,1} < z_1) \texttt{ and } (x_{1,2} < z_2)) \texttt{ or } ((x_{1,3} < z_3) \texttt{ and } (x_{1,4} < z_4))).$$

Аналогично, булева функция для второй строки равна:

$$((((x_{2,1} < z_1) \texttt{ or } (x_{2,2} < z_2)) \texttt{ and } (x_{2,3} < z_3)) \texttt{ or } (x_{2,4} < z_4)),$$

а для третьей строки:

$$(((x_{3,1} < z_1) \texttt{ and } (x_{3,4} < z_4)) \texttt{ or } ((x_{3,2} < z_2) \texttt{ and } (x_{3,3} < z_3))).$$

При подстановке порогов $z_1 = 0,\ z_2 = 1,\ z_3 = 2,\ z_4 = 3$ получим следующие выражения.
Вторая строка:

$$((((2 < 0) \texttt{ or } (2 < 1)) \texttt{ and } (1 < 2)) \texttt{ or } (0 < 3)) = (((0 \texttt{ or } 0) \texttt{ and } 1) \texttt{ or } 1) = (0 \texttt{ or } 1) = 1,$$

Третья строка:

$$(((1 < 0) \texttt{ and } (1 < 3)) \texttt{ or } ((0 < 1) \texttt{ and } (0 < 2))) = ((0 \texttt{ and } 1) \texttt{ or } (1 \texttt{ and } 1)) = (0 \texttt{ or } 1) = 1.$$

Заметим, что это не единственный подходящий набор порогов, также подойдут, например, значения $z_1 = 0,\ z_2 = 0,\ z_3 = 3,\ z_4 = 3$.