

Задача А Контроль доступа

Максимальное время работы на одном тесте:	3 секунды
Максимальный объем используемой памяти:	256 мегабайт

Вася разрабатывает новый веб-сервер. В настоящее время он работает над функцией, обеспечивающей поддержку списков контроля доступа. Список контроля доступа позволяет ограничить доступ к некоторым ресурсам веб-сайта, основываясь на основании IP-адреса запрашивающей стороны.

Каждый IP-адрес — это 4-байтный номер, который записан байт за байтом в десятичной записи. Байты разделены точками следующим образом: "byte0.byte1.byte2.byte3" (кавычки добавлены для ясности). Каждый байт записывается как десятичное число от 0 до 255 (включительно) без ведущих нулей. IP-адреса организованы в IP-сети. IP сети описывается двумя 4-байтовыми числами - сетевым адресом и маской сети. И сетевой адрес и маска сети записаны в той же форме, что и IP-адреса. Для того чтобы понять смысл сетевого адреса и маски сети рассмотрим их двоичное представление. Двоичное представление IP адреса, сетевого адреса и маски сети состоит из 32 бит: 8 бит для byte0 (от старших к младшим), затем по 8 бит для byte1, 8 бит для byte2 и 8 бит для byte3.

IP сеть содержит 2^N IP-адресов, где $0 \leq N \leq 32$. В маске сети первые $32-N$ бита установлены в единицы, и последние N бит установлены в ноль. Сетевой адрес имеет произвольные $32-N$ первых бит, а последние N бит установлен в ноль. IP сеть содержит все IP-адреса, первые $32-N$ бит которых равны $32-N$ первых бит сетевого адреса с произвольными N последними битами. Например, IP сеть с сетевым адресом 194.85.160.176 и сетевая маска 255.255.255.248 содержит 8 IP-адресов, с 194.85.160.176 по 194.85.160.183 (включительно).

IP сети, как правило, обозначается как "byte0.byte1.byte2.byte3/S", где "byte0.byte1.byte2.byte3" — сетевой адрес, S — это число бит, установленных в единицу в маске сети. Например, IP сети из предыдущего абзаца обозначается как 194.85.160.176/29. Список контроля доступа содержит упорядоченный список правил. Каждое правило имеет одну из следующих форм:

deny from <IP network> — запрещает доступ к ресурсу для любого IP из заданной сети.

deny from <IP address> — запрещает доступ к ресурсу для указанного IP-адреса.

allow from <IP network> — разрешает доступ к ресурсу для любого IP из заданной сети.

allow from <IP address> — разрешает доступ к ресурсу для указанного IP-адреса.

Когда кто-нибудь обращается к какому-либо ресурсу, первым делом проверяется IP-адрес обращающегося по списку контроля доступа. Правила проверяются в том порядке, в котором они перечислены, и применяется первое выполняющееся правило. Если ни одно из правил не соответствует IP-адресу запрашивающей стороны, то доступ предоставляется.

По известному списку контроля доступа и списку запросов от IP-адресов, необходимо выяснить для каждого из запросов будет ли предоставлен доступ к ресурсу.

Входные данные

Первая строка ввода содержит число N — количество правил в списке контроля доступа ($0 \leq N \leq 100\,000$). Следующие N строк содержат правила по одному в строке. IP сеть всегда записывается как "byte0.byte1.byte2.byte3/S". Следующая строка содержит число M — количество IP адресов, которые следует проверить ($0 \leq M \leq 100\,000$). Следующие M строк содержат по одному IP адресу в строке.

Выходные данные

Для каждого из M IP-адресов выведите «А», если доступ будет предоставлен, и «D» иначе. Все символы следует выводить слитно, не разделяя пробелами.

Пример

Входные данные	Выходные данные
4 allow from 10.0.0.1 deny from 10.0.0.0/8 allow from 192.168.0.0/16 deny from 192.168.0.1	ADAAA
5 10.0.0.1 10.0.0.2 194.85.160.133 192.168.0.1 192.168.0.2	

Задача В Информационный стенд

Максимальное время работы на одном тесте:	3 секунды
Максимальный объем используемой памяти:	256 мегабайт

В холле 179 школы, есть информационный стенд размером $H \times W$ (H – высота, W – ширина). На этом стенде размещается информация о кружках, изменениях в расписании, победах в олимпиадах, а также другая важная информация.

Первого сентября стенд был пуст. Одно за другим на нем начали появляться объявления, которые потом не снимались.

Каждое объявление представляет собой полоску бумаги единичной высоты. i -ое объявление представляет собой прямоугольник размером $1 \times W_i$.

Когда кто-либо вешает объявление на стенд, то он старается повесить объявление как можно выше. Среди возможных верхних мест всегда выбирается самое левое. Если подходящего места для объявления не нашлось, то не объявление не вешается на стенд. Ваша задача состоит в том, чтобы для каждого объявления определить, как высоко оно будет располагаться (в каком ряду).

Входные данные

Первая строка содержит три целых числа H , W и N ($1 \leq H, W \leq 10^9$; $1 \leq N \leq 200\,000$) — размеры стенда и количество объявлений.

Каждая из следующих N строк содержит по одному целому числу W_i ($1 \leq N \leq 10^9$) — ширину i -го объявления.

Выходные данные

Для каждого из объявлений (в порядке следования во входном файле) выведите номер ряда, в котором оно будет размещено. Ряды занумерованы от 1 до H сверху-вниз. Если объявление разместить нельзя — выведите «-1».

Пример

Входные данные	Выходные данные
3 5 5	1
2	2
4	1
3	3
3	-1
3	

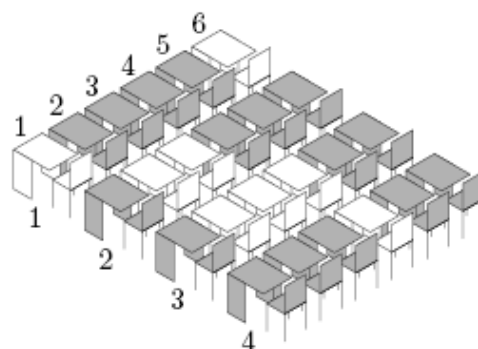
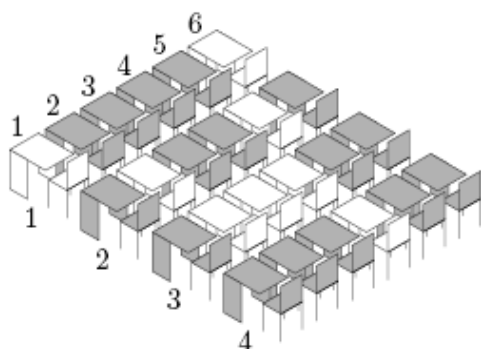
Задача С Класс

Максимальное время работы на одном тесте:	3 секунды
Максимальный объем используемой памяти:	256 мегабайт

Профессор Странненький действительно ведет себя странно. На каждой лекции он определяет заполненность класса и если она, по его мнению, мала, то он отчисляет одного из студентов.

Заполненность класса профессор определяет как минимум из строчной и столбцовой заполненности. Строчная заполненность это максимальное количество студентов в одном ряду, а столбцовая — максимальное количество студентов в одном столбце.

Например, есть 16 студентов. На левом рисунке (занятые парты выделены серым) строчная заполненность равна 5 (4 ряд), а столбцовая 3 (1, 3, 5 и 6 столбцы). Таким образом, заполненность класса равна 3. Если же студенты переседают так, как показано на рисунке справа, то строковая заполненность станет равна 4 (5 строка) и столбцовая заполненность также будет равна 4.



Студенты профессора Странненького хотят знать рассадку, которая максимизирует заполненность класса.

Входные данные

Первая строка входного файла содержит три числа N , R и C — количество студентов, строк и столбцов в классе ($1 \leq R, C \leq 100, 1 \leq N \leq R \times C$).

Выходные данные

В первой строке выведите максимальную заполненность класса.

В следующих R строках выведите оптимальную рассадку студентов. Описание одной строки должно состоять из символов '.' или '#', где '.' - это свободная парта, а '#' - занятая. В случае если существует несколько решений — выведите любое из них.

Пример

Входные данные	Выходные данные
16 4 6	4 .####. #...## #...## ###.##

Задача D Экономический кризис

Максимальное время работы на одном тесте:	3 секунды
Максимальный объем используемой памяти:	256 мегабайта

Экономический кризис вынуждает центробанки многих стран выделять большие пакеты экономической помощи инвестиционным и сберегательным банкам для поддержания ликвидности и сохранения кредитных рынков.

Центробанк Флатландии планирует выделить N пакетов помощи. Каждый пакет характеризуется его суммой A_i .

Банки предоставили запросы на помощь. На данный момент поступило M запросов. Каждый запрос характеризуется его продолжительностью B_i дней.

Законы Флатландии требуют, чтобы при возвращении пакета помощи банк каждый день возвращал центробанку одинаковую целую сумму флатландских рублей. Таким образом, пакет с суммой A и продолжительностью B можно выделить только в случае если A делится на B .

По известной информации о пакетах помощи и запросах подсчитайте количество возможных пар пакет-запрос.

Входные данные

Первая строка содержит целое число N – количество пакетов помощи ($1 \leq N \leq 100\,000$). Вторая строка содержит N целых чисел: $a_1, a_2, \dots, a_n$ ($1 \leq a_i \leq 10^6$).

Вторая строка содержит целое число M – количество запросов ($1 \leq M \leq 100\,000$). Вторая строка содержит M целых чисел: $b_1, b_2, \dots, b_n$ ($1 \leq b_i \leq 10^6$).

Выходные данные

Выведите количество возможных пар.

Пример

Входные данные	Выходные данные
4 3 4 5 6 4 1 1 2 3	12

Комментарий к примеру

Пары могут быть следующие: (3, 1) дважды как (a_1, b_1) и (a_1, b_2) , (3, 3), (4, 1) дважды, (4, 2), (5, 1) дважды, (6, 1) дважды, (6, 2) и (6, 3).

Задача Е Заколдованное зеркало

Алисе нравятся две вещи — ее зеркало и ее кубики. Кубики были предназначены для изучения детьми алфавита, то есть на их сторонах написаны некоторые буквы. Алиса любит играть с кубиками возле зеркала.

Когда Алиса учила алфавит, она заметила, что с ее зеркалом что-то не так! Кубик в зеркале мог показывать не ту букву, что изображена на нем, но всегда одну и ту же. Алиса придумала новую игру, пытаясь составить смешные слова из реальных кубиков и из кубиков в зеркале одновременно.

Игра имела следующие правила. Алиса выкладывала из кубиков слово S_1 . Эти же буквы в зеркале показывали слово S_2 , которое могло отличаться от отражения S_1 , так как зеркало было заколдованным. Но длина каждого слова равнялась N .

Затем Алиса проделывала следующие шаги. Она выбирала два кубика i и j и меняла их местами. В зеркале при этом менялись изображения кубиков $N - i + 1$ и $N - j + 1$ соответственно.

Цель игры — получить из слова S_1 слово T_1 , которое будет выглядеть в зеркале как слово T_2 . Алиса не знает, когда это возможно, а когда нет. Помогите ей ответить на этот вопрос.

Входные данные

Во входном файле находятся 4 слова S_1 , S_2 , T_1 and T_2 , каждое в отдельной строке. Все слова имеют одну и ту же длину N ($1 \leq N \leq 100$) и состоят только из заглавных латинских букв.

Выходные данные

Выведите Yes или No в зависимости от ответа на вопрос задачи.

Примеры

Входные данные	Выходные данные
TEAM TIED MATE EDIT	Yes
TEAM MATE TAME MEAT	No
AAAA AAAA AAAA AAAA	Yes

Задача F Дерево Фенвика

Максимальное время работы на одном тесте:	3 секунды
Максимальный объем используемой памяти:	256 мегабайт

Дерево Фенвика — это структура данных, эффективно поддерживающая запросы о сумме префикса числового массива. Для числа t обозначим $h(t)$ максимальное значение k , такое что t делится на 2^k . Например, $h(24) = 3$, $h(5) = 0$. Обозначим $l(t) = 2^{h(t)}$, например, $l(24) = 8$, $l(5) = 1$.

Рассмотрим массив $a[1], a[2], \dots, a[n]$ целых чисел. Дерево Фенвика для этого массива — это массив $b[1], b[2], \dots, b[n]$, такой что

$$b[i] = \sum_{j=i-l(i)+1}^i a[j]$$

Таким образом

$$b[1] = a[1],$$

$$b[2] = a[1] + a[2],$$

$$b[3] = a[3],$$

$$b[4] = a[1] + a[2] + a[3] + a[4],$$

$$b[5] = a[5],$$

$$b[6] = a[5] + a[6],$$

Например, дерево Фенвика для массива $a = (3, -1, 4, 1, -5, 9)$ есть массив $b = (3, 2, 4, 7, -5, 4)$.

Назовем массив само-фенвиковским, если он совпадает со своим деревом Фенвика. Например, массив $a = (0, -1, 1, 1, 0, 9)$ таковым является.

Вам дан массив a . Вам разрешается заменять в нем некоторые элементы, не меняя их порядка, чтобы сделать из исходного массива само-фенвиковский. Количество изменений при этом должно быть минимально возможным.

Входные данные

В первой строке входных данных содержится количество чисел в массиве n ($1 \leq n \leq 100\,000$). Во второй строке находятся сами n целых чисел. Все числа по модулю не превосходят 10^9 .

Выходные данные

Выведите n чисел — элементы видоизмененного массива. Если решений несколько — выведите любое из них.

Пример

Входные данные	Выходные данные
6 3 -1 4 1 -5 9	0 -1 1 1 0 9

Максимальное время работы на одном тесте:	3 секунды
Максимальный объем используемой памяти:	256 мегабайт

Представьте теперь печатную машинку с очень острыми буквами, которые вместо печати протыкают бумагу. Ясно, что цифра 0 на такой машинке вырежет дырку в бумаге, и из нее выпадет маленький овал. То же самое случится с цифрами 4, 6, 9. А 8 вырежет уже две дырки. Остальные проткнут бумагу, но не вырежут дырку.

Входные данные

Выходные данные

Примеры

[illegible]

Задача Н Важные переключатели

Максимальное время работы на одном тесте:	3 секунды
Максимальный объем используемой памяти:	256 мегабайт

Коля купил новую материнскую плату для своего компьютера, но ему кажется, что она работает некорректно. Возможно неправильно выставлены переключатели на ее разъеме, которые могут находиться в двух состояниях — включен или выключен. Коля хочет узнать текущее состояние переключателей.

К несчастью, переключатели не являются доступными. Но Коля нашел сокет, каждый пин которого соединен с одним из переключателей через некоторую схему. Коля нашел эту схему в Интернете. Он обозначил переключатели маленькими латинскими буквами, а пины сокета — большими буквами. После этого он выписал логическую формулу для каждого пина. Согласно этой формуле включенный переключатель представляется значением *true*, а выключенный — *false*.

Коля использует в формуле следующие обозначения (операции перечислены от высшего приоритета к низшему):

- Обозначение пина — буква от 'A' до 'K';
- Скобки обозначают, что если E — это формула, то (E) — тоже формула;
- Отрицание — $\sim E$ — это формула для любой формулы E ;
- Конъюнкция — $E_1 \& E_2 \& \dots \& E_n$;
- Дизъюнкция — $E_1 | E_2 | \dots | E_n$;
- Импликация $E_1 \Rightarrow E_2 \Rightarrow \dots \Rightarrow E_n$. Импликации выполняются *справа налево*: $E_1 \Rightarrow E_2 \Rightarrow E_3$ означают $E_1 \Rightarrow (E_2 \Rightarrow E_3)$;
- Эквивалентность — $E_1 \Leftrightarrow E_2 \Leftrightarrow \dots \Leftrightarrow E_n$. Такое выражение вычисляется так: $(E_1 \Leftrightarrow E_2) \& (E_2 \Leftrightarrow E_3) \& \dots \& (E_{n-1} \Leftrightarrow E_n)$.

Коля может построить новую схему, соответствующую любой формуле. Переменными в этой формуле будут состояния пинов сокета. Для начала он хочет построить схему, в которой входными значениями будут состояния всех пинов сокета, а в качестве выхода мы получим значение единственного переключателя, который при любых входных значениях будет находиться в состоянии включено. Напишите программу, которая поможет Коле написать соответствующую формулу.

Входные данные

Первая строка входных данных содержит единственное целое число n — количество пинов у сокета ($1 < n < 10$). Следующие n строк содержат описания каждого пина. Одно описание состоит из обозначения пина и соответствующей формулы, отделенной от обозначения символами ':=' . Пин обозначен заглавной латинской буквой. Его формула расположена в одной строке и состоит из элементов 'a'..'k', '(', ')', '~', '&', '|', '=>' и ' $\Leftrightarrow$ '. Элементы формулы могут быть отделены друг от друга произвольным числом пробелов. Описание каждого пина содержит не более 1000 символов.

Выходные данные

Если требуемая схема существует, то выведите в первой строке **Yes** и **NO** в противном случае.

В первом случае в следующей строке выведите формулу для переключателя, которая всегда истинна. Имя каждого пина должно входить в нее по крайней мере один раз. Имен переключателей она содержать не должна. Длина формулы не должна превосходить 1000 символов.

Пример

Входные данные	Выходные данные
3 A := (a=>c) & (b<=>d) C:= a b B := c d	Yes C&A => B ~A

Задача I Не в билетах счастье, а в их количестве

Максимальное время работы на одном тесте:	3 секунды
Максимальный объем используемой памяти:	256 мегабайт

Люди, покупающие какие-либо билеты, часто пытаются понять, на сколько счастливый билет им попался. При этом определения счастья бывают различные. В общественном транспорте Кирова для нумерации билетов используются числа от 1 до n . Витя считает билет счастливым, если его номер делится на сумму его цифр. Помогите Вите определить количество счастливых билетов.

Входные данные

На вход подается число n ($1 \leq n \leq 10^{12}$).

Выходные данные

Выведите количество счастливых билетов в диапазоне от 1 до n .

Пример

Входные данные	Выходные данные
100	33

Задача J Ключ к успеху

Максимальное время работы на одном тесте:	3 секунды
Максимальный объем используемой памяти:	256 мегабайт

Организаторы ТВ-Шоу “Ключ к успеху” подготовили ряд призов для победителей. Если победитель набрал X очков, он должен выбрать призов в точности на X у.е. От предыдущих шоу у организаторов остались N призов стоимостью $a_1, a_2, \dots, a_n$ у.е. соответственно. К сожалению, не известно, сколько именно очков наберет победитель очередной игры. Организаторы решили купить еще M призов, чтобы максимизировать минимальную сумму очков, которую уже нельзя будет набрать имеющимися призами.

Например, если уже имеются призы в 2, 3 и 9 у.е., а покупается 2 приза, то купить надо призы стоимостью 1 и 7 долларов. Тогда победитель сможет набрать себе призы при любом счете от 1 до 22 очков.

Входные данные

В первой строке входных данных находятся два целых числа N и M — число призов у организаторов и число призов, которое они собираются докупить ($0 \leq N \leq 30$, $1 \leq M \leq 30$). Вторая строка содержит N целых чисел в диапазоне от 1 до 10^9 — стоимости имеющихся призов.

Выходные данные

Выведите M чисел — стоимости призов, которые надо купить. Числа следует выводить в порядке неубывания. Если решений несколько — выведите любое из них.

Пример

Входные данные	Выходные данные
3 2 2 3 9	1 7