

Задача А. Марсианский волейбол

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

Волейбольный матч на Марсе играется двумя командами до k очков, при этом для победы отрыв должен быть хотя бы на 2 очка. За каждый розыгрыш мяча одна из команд получает ровно 1 очко.

На текущий момент матча счет первой команды равен x , а счет второй команды равен y . Какое минимальное количество мячей будут разыграны до окончания матча?

Формат входных данных

Входные данные содержат несколько тестовых наборов. В первой строке задано количество тестов t ($1 \leq t \leq 5000$).

Каждый тестовый пример описывается одной строкой, содержащей три целых числа, разделенных пробелами: k , x и y ($1 \leq k \leq 100$; $0 \leq x, y \leq 100$).

Гарантируется, что счет мог быть получен при корректной незавершенной игре.

Формат выходных данных

Для каждого теста в отдельной строке выведите ответ на него — минимальное количество мячей, которые будут разыграны до окончания матча.

Пример

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

Задача В. Раскраска стены

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

Девочка Маша смотрит на стену в своей комнате. Стена выложена квадратной плиткой, но вместо некоторых плиток установлены лампы. Таким образом, можно представить, что стена представляет собой клетчатый прямоугольник размером $n \times m$, в некоторых клетках которого находятся лампы.

У Маши есть краски k различных цветов. Маша хочет покрасить все плитки на стене так, чтобы на любом вертикальном или горизонтальном отрезке плиток, ограниченном с каждого конца краем стены или лампой, цвета не повторялись. При этом лампы Маша, разумеется, красить не будет. Маше не обязательно требуется использовать все цвета.

Маша просит вас придумать, как ей покрасить стену.

Формат входных данных

Входные данные содержат несколько тестовых наборов. В первой строке задано количество тестов t .

Каждый из тестов описывается следующим образом: в первой строке описания теста содержатся три числа n , m , k ($1 \leq n, m \leq 100$, $1 \leq k \leq \max(n, m)$) — размеры стены и количество различных красок у Маши.

В следующих n строках содержатся по m чисел a_{ij} :

- $a_{ij} = 0$, если на позиции (i, j) находится лампа;
- $a_{ij} = 1$, если на позиции (i, j) находится плитка.

Суммарное количество плиток и ламп по всем тестам не превосходит 10^5 .

Формат выходных данных

Для каждого теста в отдельной строке сначала выведите ответ на него:

- NO, если не существует ни одного способа покрасить стену так, как хочет Маша.
- YES, если способ перекрасить есть. В этом случае в следующих n строках выведите по m чисел b_{ij} — цвет плитки на позиции (i, j) или 0, если на этой позиции лампа. Если подходящих раскрасок несколько, выведите любую из них.

Пример

стандартный ввод	стандартный вывод
2	YES
4 3 2	0 2 0
0 1 0	2 0 2
1 0 1	1 0 1
1 0 1	0 1 0
0 1 0	NO
3 4 2	
0 1 0 1	
1 0 1 1	
1 1 1 0	

Задача С. Магический артефакт

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

Максим играет в компьютерную игру. Она состоит из n уровней, которые пронумерованы от 1 до n . Их можно проходить в любом порядке, на i -й уровень Максим тратит a_i секунд.

Кроме того, Максим может найти магический артефакт, который усилит его персонажа. Артефакт в игре ровно один, и его местонахождение точно неизвестно — на i -м уровне он находится с вероятностью p_i . После получения артефакта играть становится существенно проще — с ним Максим проходит i -й уровень за b_i секунд ($b_i \leq a_i$). Обратите внимание, что артефакт не действует на том уровне, на котором Максим его нашёл.

Максим хочет проходить уровни в таком порядке, чтобы минимизировать математическое ожидание времени игры. Помогите ему вычислить, какого минимального математического ожидания он сможет добиться, выбрав некоторый порядок прохождения уровней. Максим выбирает порядок уровней заранее, этот порядок не должен зависеть от того, на каком уровне окажется артефакт.

Математическим ожиданием случайной величины называется сумма по всем возможным исходам произведения вероятности этого исхода на значение величины. В данном случае исходы соответствуют тому, на каком уровне окажется артефакт, а значение величины — время прохождения, если артефакт оказался на этом уровне, а Максим проходит уровни в выбранном порядке.

Формат входных данных

Входные данные содержат несколько тестовых примеров. В первой строке задано одно число t — количество тестов ($1 \leq t \leq 1000$).

Каждый из тестов описывается следующим образом: в первой строке задано число n — количество уровней ($1 \leq n \leq 10^5$).

В следующих n строках описаны уровни. Каждый из них задаётся тремя целыми числами: a_i , b_i , x_i — время прохождения уровня до и после нахождения артефакта и значение, которое помогает вычислить вероятность найти артефакт на этом уровне. Вероятность вычисляется по формуле $p_i = x_i / 10^7$ ($1 \leq b_i \leq a_i \leq 10^5$; $0 \leq x_i \leq 10^7$; сумма всех x_i равна 10^7).

Сумма n по всем тестам не превосходит $5 \cdot 10^5$.

Формат выходных данных

Для каждого теста выведите одно число — математическое ожидание времени игры при оптимальном выборе порядка прохождения уровней. Ответ будет считаться правильным, если его абсолютная или относительная погрешность не превышает 10^{-6} .

Пример

стандартный ввод	стандартный вывод
2	16
3	10.25
10 5 10000000	
5 3 0	
7 3 0	
4	
3 1 2500000	
4 1 2500000	
10 1 2500000	
2 1 2500000	

Задача D. Менеджер памяти

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

Петя активно ведет разработку своего собственного менеджера памяти MEM 2.0 для хранилища на базе магнитных 7D блоков. Однако он столкнулся с проблемой оптимальной выдачи данных из хранилища пользователям.

Всего в менеджере Пети хранится n блоков с данными, пронумерованных от 1 до n , и имеется q запросов о выдаче данных с одного или нескольких из них. Запросы необходимо обработать в том порядке, в котором они поступили.

Для выдачи данных у Пети есть k указателей, каждый из которых указывает на один из блоков. Изначально Петя может поставить указатели на любой набор блоков.

Петин менеджер умеет мгновенно выдавать данные из любого числа блоков, если на каждый из них указывает хотя бы один указатель. Если же это не так, менеджер сначала переставляет указатели, чтобы условие выполнялось, а затем выдает данные — время выполнения этой операции для i -го запроса составляет s_i миллисекунд, можно переставить любое число указателей.

Петя хочет реализовать переставление указателей в менеджере так, чтобы суммарное время ответа на все запросы пользователей было минимальным. Запросы необходимо обработать в том порядке, в котором они поступили, менять местами запросы нельзя. Помогите ему.

Рассмотрим приведенные в примере тесты.

В первом тесте Петя изначально может поставить указатели на блоки 1, 2 и 4 — после этого первые два запроса выполнятся мгновенно. Перед третьим запросом за $s_3 = 1$ миллисекунду он может поставить указатели на блоки 2, 3 и 5, а перед четвертым запросом за $s_4 = 1$ миллисекунду он может поставить указатели на блоки 1, 3 и 5. Суммарное время на все запросы будет равно $s_3 + s_4 = 2$ миллисекунды.

Во втором тесте Пете невыгодно выполнять первые два запроса мгновенно (ставя изначально указатели на блоки 1, 2 и 4), потому что потом ему придется потратить 10 миллисекунд, чтобы поставить указатели на другие блоки. Поэтому оптимальная стратегия для Пети — изначально поставить указатели на блоки 1, 2 и 3, перед вторым запросом на блоки 1, 3, 4 за $s_2 = 1$ миллисекунду, а перед последним запросом поставить указатели на блоки 1, 3 и 5 за $s_4 = 3$ миллисекунды. Итого суммарное время ответа на запросы $s_2 + s_4 = 4$ миллисекунды.

Формат входных данных

Входные данные содержат несколько тестовых наборов. В первой строке задано количество тестов t ($1 \leq t \leq 1000$).

Каждый из следующих t тестов описывается следующим образом: в первой строке описания теста содержатся три числа n , k , q — количество блоков в менеджере, количество указателей и количество пользовательских запросов соответственно ($1 \leq k \leq n \leq 10^5$, $1 \leq q \leq 10^6$).

В следующей строке содержится q целых чисел s_i — время, требуемое для перестановки указателей при ответе на i -й запрос ($1 \leq s_i \leq 10^4$).

В следующих q строках содержится описание запросов пользователей, i -й запрос описывается одной строкой, где первое число c_i описывает количество блоков данных, которое пользователь хочет получить ($1 \leq c_i \leq k$), а следующие c_i чисел описывают номера этих блоков $b_{i,j}$ в возрастающем порядке ($1 \leq b_{i,j} \leq n$).

Гарантируется, что сумма n во всех тестах одних входных данных не превосходит 10^5 , а также сумма c_i во всех тестах одних входных данных не превосходит 10^6 .

Формат выходных данных

Для каждого теста выведите ответ на него — минимальное суммарное время ответа на все запросы пользователей.

Пример

стандартный ввод	стандартный вывод
2	2
5 3 4	4
1 1 1 1	
1 2	
2 1 4	
2 2 3	
3 1 3 5	
5 3 4	
1 1 10 3	
1 2	
2 1 4	
2 1 3	
3 1 3 5	

Задача Е. ЛИСА

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

Илья работает в Лаборатории Изучения Строковых Алгоритмов (ЛИСА). Сейчас он пытается решить такую задачу.

Задан массив строк $s_1, s_2, \dots, s_n$ и q запросов. Каждый запрос характеризуется двумя целыми числами l_i и r_i ($1 \leq l_i \leq r_i \leq n$). Для ответа на запрос необходимо сделать следующее. Назовем строку представимой, если ее можно получить следующим образом: выберем две строки s_x и s_y , где $l_i \leq x, y \leq r_i$, выберем непустой префикс строки s_x и непустой суффикс строки s_y , возьмем их конкатенацию. Ответом на запрос является количество различных представимых строк для заданных l_i и r_i .

Например, пусть $s = [abc, ab, ac, bcac]$, выберем $l_i = 2, r_i = 3$. Представимыми будут строки:

$x = 2, y = 2$: $ab = a + b, aab = a + ab, abb = ab + b, abab = ab + ab$.

$x = 2, y = 3$: $ac = a + c, aac = a + ac, abc = ab + c, abac = ab + ac$.

$x = 3, y = 2$: $ab = a + b, aab = a + ab, acb = ac + b, acab = ac + ab$.

$x = 3, y = 3$: $ac = a + c, aac = a + ac, acc = ac + c, acac = ac + ac$.

Всего получилось 12 различных представимых строк.

Помогите Илье решить задачу достаточно быстро.

Формат входных данных

В первой строке заданы два целых числа n и q — количество слов и запросов соответственно ($1 \leq n, q \leq 10^5$).

Каждая из следующих n строк содержит непустые слова s_i , состоящие из строчных латинских букв. Суммарная длина всех слов не превосходит 10^5 .

Следующие q строк содержат пары чисел l_i, r_i ($1 \leq l_i \leq r_i \leq n$) — левая и правая граница i -го запроса соответственно.

Формат выходных данных

Выведите q строк, i -я из них должна содержать единственное целое число — ответ на i -й запрос.

Пример

стандартный ввод	стандартный вывод
4 3	20
abc	23
ab	12
ac	
bcac	
3 4	
1 3	
2 3	