

Задача А. Ошибка (D)

Имя входного файла: `mistake.in` или стандартный ввод
Имя выходного файла: `mistake.out` или стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Вася решил поупражняться в арифметике. Он выбрал число 1000 и начал прибавлять к нему или отнимать от него тройку. Результат каждой операции он записывал на бумажке. Мимо проходил его друг Коля и заметил ошибку, но где именно он не сказал. Помогите Васе найти порядковый номер первой операции, в которой он допустил ошибку.

Формат входных данных

В i -ой строке входного файла содержится целое число x_i — результат, который получил Вася после i -ой операции ($-10^9 \leq x_i \leq 10^9$). Гарантируется, что Вася совершил не более ста операций, и ошибся хотя бы один раз

Формат выходных данных

Выведите номер операции, в которой Вася впервые допустил ошибку. Операции нумеруются с единицы.

Пример

<code>mistake.in</code> или стандартный ввод	<code>mistake.out</code> или стандартный вывод
1003 1000 997 993	4
1003 1000 998 1001	3

Задача В. Переправа через реку (D)

Имя входного файла: `river.in` или стандартный ввод
Имя выходного файла: `river.out` или стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Трое загадочных друзей Тимофей, Валентин и Дмитрий очень любят путешествовать. Однажды, во время путешествия по очень опасным джунглям загадочной страны Кайнозонии, они наткнулись на глубокую реку, кишашую пираньями. К счастью для них, на берегу реки находятся три одноместных лодки. На каждой из лодок написана её грузоподъёмность — максимально возможная масса человека, который может переплыть реку на этой лодке и не утопить её. Кроме того, каждый из наших героев знает свою массу. Помогите им определить, смогут ли они выбрать каждый по одной лодке так, чтобы одновременно переправиться на другой берег и не утонуть.

Формат входных данных

В первой строке входного файла содержатся три целых числа w_t, w_v, w_d — массы Тимофея, Валентина и Дмитрия соответственно ($1 \leq w_t, w_v, w_d \leq 1000$). Во второй строке входного файла содержатся три целых числа m_1, m_2, m_3 — грузоподъёмности первой, второй и третьей лодки соответственно ($1 \leq m_1, m_2, m_3 \leq 1000$).

Формат выходных данных

В единственной строке выходного файла выведите «YES» (без кавычек), если ребята могут пересечь реку и не утонуть, и «NO» иначе.

Примеры

<code>river.in</code> или стандартный ввод	<code>river.out</code> или стандартный вывод
3 2 1 2 4 2	YES
3 2 1 1 1 1	NO

Замечание

В первом примере, один из возможных вариантов: Тимофей переплывёт реку на второй лодке, Валентин — на первой лодке, а Дмитрий — на третьей лодке. Во втором примере ребята не смогут переплыть на другой берег и не утонуть.

Задача С. Касса (D, C')

Имя входного файла: cash.in или стандартный ввод
Имя выходного файла: cash.out или стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Оля решила стать предпринимателем, и недавно открыла свой магазин «Н-аудио». Магазин работает уже в течение целых n дней, и в конце каждого дня Оля приходит, и записывает на листочек количество денег в кассе магазина. При этом, в конце дня деньги из кассы не изымаются, то есть, после первого дня в кассе хранится выручка за первый день, после второго дня выручка за первый и второй дни и так далее. Оле стало интересно, в какой из n дней выручка магазина была максимальна. Если таких дней несколько, то Олю интересует первый такой день.

Формат входных данных

В первой строке входного файла находится целое число n — количество дней, в течение которых работал магазин ($1 \leq n \leq 10^5$). В i -ой из следующих n строчек содержится целое число a_i — описание баланса рублей в кассе магазина после i -ого дня ($0 \leq a_i \leq 10^9$). Гарантируется, что деньги из кассы никогда не забирали, то есть $a_i \geq a_{i-1}$, для любого i от 2 до n .

Формат выходных данных

В единственной строке выходного файла выведите два целых числа — номер дня, в котором выручка магазина была максимальна, и величину этой выручки. Если таких дней несколько, то выведите самый ранний из них. Дни в магазине нумеруются с единицы.

Пример

cash.in или стандартный ввод	cash.out или стандартный вывод
3 4 5 17	3 12
5 1 1 3 5 5	3 2

Задача D. Интересные числа (C', C)

Имя входного файла: `funny-numbers.in` или стандартный ввод
Имя выходного файла: `funny-numbers.out` или стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Маленький Вася очень любит числа, а особенно сильно он любит интересные числа. Вася считает число x интересным, если сумма квадратов его цифр делится на число 7. Например, число 123 — интересное, потому что $1^2 + 2^2 + 3^2 = 14$ делится на 7, а число 16 — нет, потому что $1^2 + 6^2 = 37$ не делится на 7.

Однажды Вася увидел на доске число x , он сразу же захотел узнать величину минимального интересного числа, которое строго больше чем x . Так как Вася ещё слишком юн, он обратился к вам за помощью в решение этой задачи.

Формат входных данных

Во входном файле содержится единственное целое число x — число, написанное на доске ($0 \leq x \leq 10^5$).

Формат выходных данных

В единственную строку выходного файла выведите минимальное интересное число, которое строго больше чем x .

Примеры

<code>funny-numbers.in</code> или стандартный ввод	<code>funny-numbers.out</code> или стандартный вывод
1	7
0	7
35	70

Задача Е. Перестановки с другим началом (С', С)

Имя входного файла: `permutations.in` или стандартный ввод
Имя выходного файла: `permutations.out` или стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Перестановкой размера n называется упорядоченный набор из n чисел, в котором каждое число от 1 до n встречается ровно один раз. Например, $(4, 2, 3, 5, 1)$ — это перестановка размера 5.

Вам дано число n и последовательность a , в которой k натуральных чисел.

Вычислите, сколько существует перестановок размера n , которые *не начинаются* на данную последовательность.

Формат входных данных

В первой строке содержатся два натуральных числа n и k ($1 \leq n \leq 9$; $1 \leq k \leq 100$).

Во второй строке содержатся k натуральных чисел, составляющие последовательность a . Каждое из этих чисел не превышает 100.

Формат выходных данных

Выведите количество перестановок размера n , которые не начинаются на данную последовательность a .

Примеры

<code>permutations.in</code> или стандартный ввод	<code>permutations.out</code> или стандартный вывод
3 2 2 1	5
5 2 4 4	120
5 6 2 3 9 5 6 6	120

Задача F. Денежная система (C, B')

Имя входного файла: `coins.in` или стандартный ввод
Имя выходного файла: `coins.out` или стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Денежная система Небритании развивалась на протяжении многих лет.

Изначально небританцы пользовались пшиллингами 0-го уровня — обычными монетками.

Во время правления Генриха 1-го были введены пшиллинги 1-го уровня, которые равнялись 10 пшиллингам 0-го уровня.

Во время правления Генриха 2-го были введены пшиллинги 2-го уровня, которые равнялись 20 пшиллингам 1-го уровня.

Во время правления Генриха 3-го были введены пшиллинги 3-го уровня, которые равнялись 30 пшиллингам 2-го уровня.

И так далее, а именно, во время правления Генриха k -го были введены пшиллинги k -го уровня, которые равнялись $10k$ пшиллингам $(k - 1)$ -го уровня.

Сейчас в казне Небритании огромная сумма, равная n пшиллингам 0-го уровня. Запишите её фразой вида «столько-то пшиллингов такого-то уровня, столько-то пшиллингов такого-то уровня и т. д.», причём суммарное количество упомянутых вами пшиллингов различных уровней должно быть минимальным.

Формат входных данных

В первой строке содержится натуральное число n ($1 \leq n \leq 10^{15}$).

Формат выходных данных

Выведите несколько пар целых чисел, по одной на строке. При этом пара (a, b) означает фразу « a пшиллингов b -го уровня».

Номера уровней в вашей фразе должны строго убывать. Можно совсем не использовать пшиллинги какого-то уровня, в этом случае не нужно выводить про них ничего. Количество пшиллингов каждого упомянутого вами уровня должно быть положительно ($a > 0$).

Примеры

coins.in или стандартный ввод	coins.out или стандартный вывод
7777	1 3
	8 2
	17 1
	7 0
6030	1 3
	3 1

Задача G. Организация конференции (C, B')

Имя входного файла: `matrix.in` или стандартный ввод
Имя выходного файла: `matrix.out` или стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Недавно в солнечный весенний день директору Летней Флатландской Компьютерной Школы (ЛФКШ) Сергею Александровичу пришла в голову идея организовать первую флатландскую конференцию для школьников по программированию. Теперь перед ним стоит задача выбрать место проведения.

Флатландия представляет собой прямоугольник из m строк и n столбцов, в каждой из клеток которого расположен один город. Сергей Александрович уже посчитал для каждого города количество желающих принять участие в конференции. Известно, что флатландцы не любят далеко ездить, так что в каком бы городе она проводилась, в конференции смогут принять участие только школьники из самого города и соседних с ним по стороне городов. Формально говоря, если конференция проводится в городе, находящемся в i -ой строке и j -ом столбце, то в этой конференции будут участвовать школьники из городов (i, j) , $(i - 1, j)$ (при условии, что $i > 1$), $(i + 1, j)$ (при условии, что $i < m$), $(i, j - 1)$ (при условии, что $j > 1$) и $(i, j + 1)$ (при условии, что $j < n$).

Сейчас Сергей Александрович хочет понять, в каких городах возможно поселить всех приезжих участников. Пока он выясняет количество доступных мест в гостиницах Флатландии, вам предстоит посчитать для каждого из возможных городов проведения, скольким школьникам потребуется предоставить жильё на время конференции.

Обратите внимание на то, что школьникам, живущим в том же городе, в котором проводится конференция, поселение не нужно.

Формат входных данных

В первой строке записаны два числа m и n ($1 \leq m, n \leq 350$) — размеры Флатландии. В каждой из последующих m строк содержатся по n чисел. В i -ой строке и j -ом столбце содержится число $a_{i,j}$ ($0 \leq a_{i,j} \leq 10000$) — количество школьников из города (i, j) , желающих принять участие в конференции.

Формат выходных данных

Выведите m строк по n чисел в каждой. Число в строке с номером i и столбце с номером j должно равняться количеству приезжих участников конференции, если местом проведения будет выбран город, с координатами (i, j) .

Примеры

<code>matrix.in</code> или стандартный ввод	<code>matrix.out</code> или стандартный вывод
3 4 1 0 3 2 5 6 1 2 1 1 0 11	5 10 3 5 8 7 11 14 6 7 13 2
1 1 5	0

Задача Н. Овечка Толя (С, В')

Имя входного файла: `sheep.in` или стандартный ввод
Имя выходного файла: `sheep.out` или стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Овечка Толя умеет клонироваться — тогда рядом с ней появляется такая же овечка, с той же логикой и привычками.

Когда овечка встречает n стогов сена, то происходит следующее:

- Если n меньше 4, то овечка выкидывает эти n стогов сена в ближайший овраг. Иначе:
- Если n делится на 5, то овечка сбрасывает $n/5$ стогов сена в ближайший овраг; клонируется; сама обрабатывает $3n/5$ стогов сена с помощью этой же процедуры, а её клон обрабатывает оставшиеся $n/5$ стогов сена с помощью этой же процедуры. Иначе:
- Овечка съедает 4 стога сена и обрабатывает оставшиеся $n - 4$ стогов с помощью этой же процедуры.

Овечка Толя однажды увидела n стогов сена — это число вам дано. Сколько стогов сена будет съедено ей и всеми её клонами при описанном процессе?

Формат входных данных

В первой строке содержится натуральное число n ($0 \leq n \leq 10^6$).

Формат выходных данных

Выведите количество стогов, съеденных в итоге Толей и клонами.

Примеры

<code>sheep.in</code> или стандартный ввод	<code>sheep.out</code> или стандартный вывод
29	8

Задача I. Опять двойка (B', B)

Имя входного файла: two-digits-two.in или стандартный ввод
Имя выходного файла: two-digits-two.out или стандартный вывод
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Пете нравится цифра 2. Он считает число *красивым*, если в его десятичной записи ровно две цифры 2. Петя хочет получить большой список красивых чисел и повесить его на стенку. Помогите Пете и выведите все красивые числа, не превосходящие n .

Формат входных данных

В первой и единственной строке записано одно целое число n ($1 \leq n \leq 10^6$).

Формат выходных данных

Выведите все целые положительные числа, не превосходящие n , в десятичной записи которых ровно две цифры 2. Числа следует выводить в порядке возрастания, по одному на строке.

Примеры

two-digits-two.in или стандартный ввод	two-digits-two.out или стандартный вывод
179	22 122
1	

Замечание

Обратите внимание, что список может быть пустым; в этом случае ничего выводить не нужно.

Задача J. Мишина машина (B', B)

Имя входного файла: coins-collecting.in или стандартный ввод
Имя выходного файла: coins-collecting.out или стандартный вывод
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Миша путешествует по стране на автомобиле. В стране есть несколько городов, соединённых дорогами, на каждой дороге может быть некоторое число ям.

Мише очень нравится ездить по дорогам, на которых много ям, — он участвует в популярном телешоу «*Неперсистентные машинки*», и за каждую посещённую яму ему платят по одной монете.

Он начинает путь в любом из городов страны и затем путешествует по дорогам между городами. По каждой дороге Миша может проехать сколько угодно раз, но ему платят за посещение любой ямы только один раз.

Помогите Мише найти самый выгодный маршрут.

Формат входных данных

В первой строке содержатся два целых числа n и m — количество городов и дорог, соединяющий города, в стране соответственно ($1 \leq n \leq 10^5$, $0 \leq m \leq 10^5$).

В следующих m строках содержится по три целых числа, разделённых пробелами, — номера городов, соединённых очередной дорогой, и количество ям на этой дороге соответственно. Количество ям на каждой дороге — неотрицательное число, не превосходящее 10^6 .

Гарантируется, что никакая дорога не соединяет город с самим собой и что нет двух дорог, соединяющих одинаковые пары городов.

Вершины нумеруются с единицы.

Формат выходных данных

Выведите одно число — максимальное количество монет, которое Миша сможет получить.

Примеры

coins-collecting.in или стандартный ввод	coins-collecting.out или стандартный вывод
4 4 1 2 1 2 3 1 1 4 1 4 3 0	3
4 2 1 3 5 2 4 4	5

Замечание

Все дороги двусторонние; проехав в любую сторону по дороге, Миша посещает все ямы на этой дороге. Обратите внимание, что между некоторыми парами городов может не существовать пути по имеющимся дорогам.

Задача К. Прямоугольное поле чудес (B, A', A, AY, AA, AS)

Имя входного файла: `rectangle-sum-0.in` или стандартный ввод
Имя выходного файла: `rectangle-sum-0.out` или стандартный вывод
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Коля попал на телеигру «Прямоугольное поле чудес». В финале этой игры Коле показали прямоугольное поле размера $n \times m$ клеток, в каждое клетке которого записано целое число. Коля может заменить числа в некоторых клетках на противоположные (т.е. вместо числа x записать в клетку число $-x$). Коля выиграет автомобиль, если сумма чисел в каждой строке и в каждом столбце будет равна нулю. Помогите Коле найти нужную расстановку чисел, либо определите, что ее не существует и Коля не сможет выиграть.

Формат входных данных

В первой строке записано два целых числа n и m ($1 \leq n, m \leq 5$) — размеры поля.

В следующих n строках записано по m целых чисел, разделенных пробелами — числа, записанные в клетках поля. Все числа по модулю не превосходят 10^6 .

Формат выходных данных

Если ответ существует, в первой строке выведите YES; в следующих n строках выведите по m чисел через пробел — числа в клетках поля после изменений.

Если ответа не существует, в первой строке выведите NO.

Примеры

<code>rectangle-sum-0.in</code> или стандартный ввод	<code>rectangle-sum-0.out</code> или стандартный вывод
3 3	YES
1 1 2	1 1 -2
2 2 4	2 2 -4
3 3 6	-3 -3 6

Задача L. Британские ученые (B, A', A, AY, AA, AS)

Имя входного файла: `first-last-anagram.in` или стандартный ввод
Имя выходного файла: `first-last-anagram.out` или стандартный вывод
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайта

Согласно исследованиям британских ученых, люди способны воспринимать слова в тексте, если в каждом слове оставить на месте первую и последнюю буквы, а остальные перемешать произвольным образом; например, слово «программа» может быть прочитано даже если оно записано как «пгрроммаа» или «пморгамра».

Вам дан словарь с несколькими словами, а также некоторый текст. Для каждого слова из текста определите, можно ли его прочесть как одно из слов словаря, руководствуясь правилами, описанными выше.

Формат входных данных

В первой строке записано одно целое число n ($1 \leq n \leq 10^5$) — количество слов в словаре.

В следующих n строках записаны слова из словаря, по одному на строку. Гарантируется, что все слова в словаре различны.

В следующей строке записано одно целое число m ($1 \leq m \leq 10^5$) — количество слов в тексте.

В следующих m строках записаны слова из текста, по одному на строку.

Каждое слово состоит только из строчных букв латинского алфавита; ни в какой строке ввода нет пробелов и других разделителей. Суммарная длина всех слов не превосходит 10^5 .

Формат выходных данных

Для каждого слова из текста выведите «YES», если его можно прочесть как одно из слов словаря, и «NO» в противном случае. Ответы для слов из текста следует выводить в том же порядке, в котором слова перечислены во вводе; следует выводить по одному ответу на строку.

Примеры

<code>first-last-anagram.in</code> или стандартный ввод	<code>first-last-anagram.out</code> или стандартный вывод
4	YES
bird	NO
sun	NO
lksh	NO
summer	
4	
brid	
snu	
sommer	
sis	

Задача М. НВП на отрезке (A', A, AY, AA, AS)

Имя входного файла: `segment-lis.in` или стандартный ввод
Имя выходного файла: `segment-lis.out` или стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Вам дана числовая последовательность a_1, \dots, a_n . Напишите программу, отвечающую на запросы вида «найти длину наибольшей **строго** возрастающей подпоследовательности, все элементы которой находятся на отрезке с l_i -ого по r_i -ый элемент».

Подпоследовательностью последовательности a_1, \dots, a_n называется последовательность, которую можно получить путем удаления нескольких элементов a_i (относительный порядок оставшихся элементов менять запрещается). Так, например, последовательность (2, 4) является подпоследовательностью последовательности (1, 2, 3, 4, 5) (можно удалить элементы 1, 3 и 5), а последовательность (5, 1) — нет.

Формат входных данных

В первой строке записано целое число n ($1 \leq n \leq 3000$) — число элементов в последовательности.

Во второй строке записано n чисел, разделенных пробелами — элементы последовательности. Все элементы не превосходят по модулю 10^9 .

В третьей строке записано одно целое число q ($1 \leq q \leq 10^5$) — количество запросов.

В следующих q строках описаны запросы. Описание i -ого запроса — два числа l_i и r_i ($1 \leq l_i \leq r_i \leq n$), записанные через пробел.

Формат выходных данных

Выведите q чисел — ответы на запросы. Числа следует выводить по одному на строке в том же порядке, в котором запросы описаны во вводе.

Примеры

segment-lis.in или стандартный ввод	segment-lis.out или стандартный вывод
6	2
3 3 -5 7 4 9	1
6	1
1 4	2
1 2	2
2 3	2
1 5	
3 5	
2 5	

Замечание

Обратите внимание, что подпоследовательность должна быть **строго** возрастающей.

Задача N. Вася и дерево (A', A, AY, AA, AS)

Имя входного файла: `mst-add-edges.in` или стандартный ввод
Имя выходного файла: `mst-add-edges.out` или стандартный вывод
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Васе на день рождения подарили дерево на n вершинах, на каждом ребре написано целое число из интервала от 1 до k — *вес* ребра. К дереву прилагается много запасных ребер с весами от 1 до k ; можно считать, что для каждого значения веса доступно неограниченное число ребер с этим весом.

Вчера на кружке по информатике Вася узнал, что такое минимальное остовное дерево. Теперь он хочет добавить к имеющемуся у него дереву несколько ребер таким образом, чтобы исходное дерево было минимальным остовным деревом для получившегося графа. Чтобы не запутаться, Вася хочет, чтобы все добавленные им ребра имели один и тот же вес. Вася не добавляет ребра таким образом, чтобы в графе могли получиться петли или кратные ребра.

Помогите Васе для каждого значения веса от 1 до k определить, какое максимальное количество ребер такого веса он сможет добавить в дерево!

Формат входных данных

В первой строке записано два целых числа через пробел — n и k ($1 \leq n, k \leq 3 \cdot 10^5$).

В следующих $n - 1$ строках дано описание ребер исходного дерева. i -ое ребро описывается тремя числами v_i , u_i и w_i ($1 \leq v_i, u_i \leq n$, $1 \leq w_i \leq k$), записанными через пробел: номера вершин, соединенных i -ым ребром, и вес этого ребра. Гарантируется, что заданный граф является деревом.

Формат выходных данных

Выведите k чисел по одному на строке — максимальное количество ребер с таким значением веса, которое можно добавить в дерево согласно правилам, описанным в условии.

Примеры

<code>mst-add-edges.in</code> или стандартный ввод	<code>mst-add-edges.out</code> или стандартный вывод
4 3	0
1 2 1	1
1 3 2	3
1 4 3	
5 10	0
1 2 7	0
2 3 2	0
3 4 5	0
4 5 9	1
	1
	3
	3
	6
	6

Замечание

В первом примере нельзя добавить ни одного ребра веса 1, поскольку тогда вес минимального остовного дерева уменьшится; можно добавить одно ребро веса 2 между вершинами 2 и 3; наконец, ребра веса 3 можно провести между всеми парами вершин, которые не соединены ребрами в исходном дереве.

Деревом называется связный неориентированный граф без циклов.

Остовным деревом связного неориентированного графа называется подмножество его ребер, образующее дерево (согласно определению выше) и покрывающее все вершины (т.е. каждая вершина графа является концом одного из ребер остовного дерева).

Минимальным остовным деревом взвешенного связного неориентированного графа называется остовное дерево, сумма весов ребер которого имеет минимально возможное значение.

Петлей называется ребро, соединяющее вершину с самой собой.

Кратными ребрами называются несколько (больше одного) ребер, соединяющих одну и ту же пару вершин.