

Задача А. Плащ левитации

Имя входного файла: `cloak.in`
Имя выходного файла: `cloak.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Во время борьбы с очередным демоном Доктор Стрэндж сильно испачкал свой незаменимый плащ левитации. Но так как использование магии без особой нужды — моветон, Стивен будет стирать в машинке, как обычный смертный.

Однако оказалось, что стирка — далеко не самая сложная дилемма на повестке дня. Стрэндж хочет повесить плащ сушиться, но все, что оказалось у него в этом измерении - бельевая веревка, натянутая на высоте h сантиметров и протяженностью l сантиметров. Плащ же представляет из себя прямоугольник со сторонами a и b сантиметров.

Стивен хочет перекинуть плащ через веревку, но при этом очень важно, чтобы висящий плащ не коснулся пола. Стивену повезло, что веревка нерастяжимая и сохранит высоту даже с повешенным на ней плащом. Также если плащ свисает ровно до пола, касание не считается, так как он очень тонкий и площадь касания в таком случае ничтожно мала.

Руки доктора заняты плащом, а ум великими мыслями, поэтому он просит вас подсказать ему, сможет ли он повесить плащ на веревку так, чтобы тот не задел пол.

Формат входных данных

В единственной строке входного файла содержатся четыре натуральных числа h , l , a и b — высота и длина бельевой веревки, а также размеры плаща в сантиметрах. ($1 \leq h, l, a, b \leq 200$)

Формат выходных данных

В выходной файл выведите YES, если Доктор Стрэндж сможет повесить плащ, чтобы тот не касался пола и NO иначе.

Примеры

<code>cloak.in</code>	<code>cloak.out</code>
100 100 50 50	YES
4 10 10 8	YES
50 20 30 30	NO

Замечание

В первом тесте Стрэндж сможет любым способом повесить плащ, во втором — согнуть по стороне длины 8, тогда плащ хоть и будет свисать до пола, но тем не менее не будет считаться касающимся. В третьем тесте плащ повесить не получится.

Задача В. Граненые стаканы

Имя входного файла: `glasses.in`
Имя выходного файла: `glasses.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Перед доктором Стрэнджем на столе стоит несколько граненых стаканов. Каждый из них при взгляде сверху выглядит как выпуклый многоугольник, а его стенки вертикальны и перпендикулярны столу.

Перед медитацией Стрэндж хочет распределить по стаканам некоторый объем воды, причем так, чтобы высота воды во всех стаканах была одинакова. Гарантируется, что стаканы достаточно высокие, чтобы так можно было сделать.

Помогите ему определить, какой уровень воды должен быть во всех стаканах.

Формат входных данных

В первой строке находятся два целых числа n и v — количество стаканов и объем воды ($1 \leq n \leq 10^5$; $1 \leq v \leq 10^{18}$).

Далее содержится описание стаканов. Форма стакана описывается многоугольником, лежащим в основании. Описание i -го многоугольника начинается со строки, в которой находится единственное число k_i — количество углов в многоугольнике ($3 \leq k_i \leq 10^4$). Далее следует k_i строк, в каждой из которых находится по два целых числа x_{ij} и y_{ij} — координаты очередного угла многоугольника ($|x_{ij}|, |y_{ij}| \leq 10^6$). Углы даны в порядке обхода против часовой стрелки. Суммарное количество углов во всех многоугольниках не превышает 10^6 .

Так как многоугольники описывают только форму стаканов, но не их положение на столе, данные многоугольники могут пересекаться.

Формат выходных данных

В единственной строке выведите одно число — высоту уровня воды в стаканах, с абсолютной или относительной погрешностью не менее 10^{-6} .

Пример

<code>glasses.in</code>	<code>glasses.out</code>
2 10	5.00000000
3	
0 0	
1 0	
0 1	
4	
1 1	
0 2	
0 0	
1 0	

Задача С. Преследование

Имя входного файла: `kdivision.in`
Имя выходного файла: `kdivision.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Кецилий украл жезл Ватумба, и теперь Доктор Стрэндж преследует его по разным альтернативным реальностям. Буквально только что Доктор потерял его, но благодаря своим способностям он все еще может следить, где Кецилий сейчас находится.

Стрэндж записал все номера реальностей, в которых успел побывать Кецилий, и решил найти какую-нибудь закономерность его действий. Однако внезапно Доктор понял, что все номера он записал подряд без пропусков, и теперь невозможно точно восстановить номера реальностей.

Тем не менее Стивен запомнил три важных условия:

- Кецилий побывал ровно в k реальностях.
- Каждая из них имеет неотрицательный номер без лидирующих нулей (кроме реальности с номером 0).
- Из реальности номер x Кецилий всегда попадал в реальность с номером y таким, что абсолютная разность чисел x и y не меньше l , но в то же время не превосходит r .

Точно восстановить передвижения злодея невозможно, поэтому сейчас Стрэндж хочет хотя бы знать количество способов разбить получившееся число на k других, чтобы выполнялись условия, описанные выше.

Формат входных данных

В первой строке содержится число t — количество тестов ($1 \leq t \leq 100$). В i -й из следующих t строк содержится описание i -го теста.

Тест описывается четырьмя числами x, l, r, k — числом, полученным Доктором Стрэнджем, границами для абсолютной разницы и количеством реальностей, в которых побывал Кецилий ($1 \leq x \leq 10^{18}, 0 \leq l \leq r \leq 10^{18}, 1 \leq k \leq 18$).

Формат выходных данных

На каждый тест в отдельной строке выведите ответ на него — искомое количество разбиений.

Пример

<code>kdivision.in</code>	<code>kdivision.out</code>
3	1
248 16 45 2	2
248 16 46 2	0
4444 1 5 2	

Замечание

В первом тесте возможно только одно разбиение, удовлетворяющее ограничениям на абсолютную разность: сначала Кецилий мог быть в реальности номер 24, а потом в реальности номер 8.

Во втором тесте возможны два разбиения: $2 \rightarrow 48, 24 \rightarrow 8$.

Задача D. Библиотека

Имя входного файла: `library.in`
Имя выходного файла: `library.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Одним из источников знаний доктора Стрэнджа являются книги, которые он довольно часто берет в библиотеке. Все книги, которые Стрэндж берет в библиотеке, он обязательно дочитывает до конца, при этом не забывая сдать каждую из них в срок.

Однако в определенный момент количество книг, взятых доктором в библиотеке, стало быстро расти, и он не успевал следить за сроками сдачи, и в добавок к этому изменились правила сдачи книг в библиотеку. Теперь в день можно сдавать не более одной книги.

Стрэндж очень ценит свое время, и поэтому просит вас помочь ему определить, сможет ли он прочесть все взятые в библиотеке книги и сдать их в срок по новым правилам.

Книги Стрэндж умеет читать параллельно, то есть каждый день он одновременно может читать сколько угодно книг. Книга также может быть сдана и в день прочтения, и в день сдачи.

Формат входных данных

В первой строке входного файла дано число n — количество взятых в библиотеке книг ($1 \leq n \leq 10^5$).

Далее следуют n строк. Каждая i -ая строка содержит три целых положительных числа s_i, f_i, c_i — день, в который доктор взял книгу в библиотеке, день, по прошествии которого книга должна быть сдана, и количество дней, которое требуется доктору для прочтения книги, соответственно ($1 \leq s_i, f_i, c_i \leq 10^9$).

Формат выходных данных

Если доктор Стрэндж сможет сдать все прочитанные им книги в срок, в первой строке выходного файла выведите YES. Иначе, если не сможет, выведите NO.

Примеры

<code>library.in</code>	<code>library.out</code>
3 2 7 3 3 8 1 1 4 2	YES
3 1 3 1 1 3 1 1 3 1	NO

Замечание

В первом тесте из условия доктор может, например, сдать первую книгу в 5-ый день, вторую - в 6-ой, а третью - в 3-ий день.

Во втором тесте доктор не сможет сдать все три книги в срок, так как каждую из них ему нужно сдать во 2-ой или в 3-ий день, что невозможно, так как в день можно сдавать не более одной книги.

Задача Е. Доктор Стрэндж и перестановка

Имя входного файла: `pots.in`
Имя выходного файла: `pots.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

У доктора Стрэнджа есть сад, в котором в ряд выставлены n горшков с цветами. На каждом горшке написано некоторое число. На позиции номер i стоит горшок с числом a_i . Иначе говоря, горшки образуют массив a .

В выходные доктор Стрэндж сделает небольшую перестановку: некоторые два горшка, находящиеся на позициях i и j ($i \neq j$) он поменяет местами. Еще Доктор Стрэндж любит закономерности, поэтому он хочет, чтобы после перестановки на четных позициях стояли четные числа, а на нечетных — нечетные.

От вас требуется выяснить, можно ли из массива a получить массив, который удовлетворяет заданному условию, сделав ровно один обмен двух элементов, стоящих на различных позициях.

Массив a индексируется с единицы.

Формат входных данных

В первой строке находится одно натуральное число n ($2 \leq n \leq 1000$).

В следующей строке находятся n натуральных чисел a_i — числа, записанные на горшках ($1 \leq a_i \leq 10^9$).

Формат выходных данных

В единственной строке выведите i и j — номера элементов, которые нужно поменять местами, чтобы добиться заданного условия ($1 \leq i, j \leq n, i \neq j$). Если ответов несколько — разрешается вывести любой.

Если не существует способа поменять два элемента местами — выведите `-1 -1`.

Примеры

<code>pots.in</code>	<code>pots.out</code>
2 2 1	1 2
3 1 2 3	1 3
4 2 1 4 6	-1 -1

Задача F. Послание

Имя входного файла: `repetition.in`
Имя выходного файла: `repetition.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Во время одного из своих путешествий доктор Стрэндж столкнулся с серьезной задачей. Ему необходимо расшифровать послание, зашифрованное в строке s , которую он получил на электронную почту. Он знает, что послание шифруется следующим образом: некоторая строка c записывается какое-то количество раз (возможно нулевое), затем между произвольными буквами полученной строки вставляются буквы послания (возможно, вставляется сразу несколько букв) и таким образом получается строка s .

Доктор Стрэндж догадался, какая строка была взята в качестве строки c и теперь, чтобы расшифровать послание, ему необходимо определить, какое максимальное количество раз могла быть записана строка c при шифровании. Помогите ему!

Формат входных данных

В первой строке входного файла дана строка c .

Во второй строке входного файла дана строка s .

Каждая строка может состоять из строчных и заглавных букв латинского алфавита. Длина каждой строки не превышает 10^6 символов.

Формат выходных данных

Выведите единственное число — ответ на задачу.

Пример

<code>repetition.in</code>	<code>repetition.out</code>
<code>ab</code> <code>abacabaaacbb</code>	3

Задача G. Знания — сила

Имя входного файла: `strength.in`
Имя выходного файла: `strength.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Доктор Стрэндж активно изучает магию. Сегодня он наконец осознал, как распространяются темные силы. Оказывается, они распространяются с помощью так называемых «носителей силы», носителями могут быть кто угодно — люди, предметы, растения. А также каждый характеризуется своим «уровнем» — количеством новых носителей, которых он может породить. Распространение происходит по следующему незамысловатому закону:

- Изначально имеется n носителей, все имеют уровень 1.
- Каждый следующий день носитель уровня i порождает новые i носителей первого уровня, которые становятся активны только на следующий день.
- Сам же носитель переходит на новый уровень $i + 1$ (это означает, что на следующий день он породит уже $i + 1$ новых носителей) и его деятельность на текущий день прекращается.

Всего в распоряжении Стрэнджа имеется k дней. Его интересует, сколько всего носителей появится за это время. За помощью он обратился именно к вам.

Формат входных данных

В единственной строке входного файла содержится два натуральных числа n и k — количество носителей изначально и дней соответственно ($1 \leq n \leq 1000, 1 \leq k \leq 10^5$).

Формат выходных данных

Выведите одно число — ответ на задачу. Так как ответ может получиться слишком большим, выведите его по модулю $10^9 + 7$.

Пример

<code>strength.in</code>	<code>strength.out</code>
3 2	15

Замечание

В таблице приведены данные по носителям в конце каждого дня:

день	первый уровень	второй уровень	третий уровень	всего носителей
0	3	0	0	3
1	3	3	0	2×3
2	$2 \times 3 + 3$	3	3	5×3

Задача Н. Путешествие сквозь миры

Имя входного файла: `travelling.in`
Имя выходного файла: `travelling.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Доктор Стрэндж совсем недавно обрел свои способности, но сразу же решил все структурировать. Получив доступ к бесконечному, но конечно же счетному количеству альтернативных реальностей, он решил их все занумеровать. Сделав это, он посмотрел на проделанную работу и заметил, что его нумерация обладает интересным свойством: все его самые любимые альтернативные реальности имеют номера, состоящие из одинаковых цифр и наоборот — любое число, состоящее из одинаковых цифр, соответствует одной из любимых реальностей Стрэнджа.

Разумеется, смотреть на все реальности сразу у доктора Стрэнджа возможности нет, поэтому в данный момент он смотрит на реальности с номерами $l, l + 1, l + 2, \dots, r$. Разумеется, его заинтересовал вопрос — а сколько различных любимых реальностей он сейчас видит? Дел у него много и времени считать это число самому у него нет, поэтому он попросил вас помочь ему с этой задачей.

Формат входных данных

В первой и единственной строке содержатся два числа l, r — номера крайних альтернативных реальностей, которые видит Доктор Стрэндж ($1 \leq l \leq r \leq 10^{18}$).

Формат выходных данных

В единственной строке выведите количество любимых альтернативных реальностей Стрэнджа, которые он сейчас видит.

Примеры

<code>travelling.in</code>	<code>travelling.out</code>
4 7	4
10 100	9

Замечание

В первом примере все номера состоят из одной цифры, а следовательно соответствуют любимым реальностям Стрэнджа.

Во втором примере подходящие номера — 11, 22, 33, 44, 55, 66, 77, 88, 99, которых ровно 9 штук.

Задача I. Карточный трюк

Имя входного файла: `trick.in`
Имя выходного файла: `trick.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

В детстве доктор Стрэндж увидел в передаче по телевизору замечательный карточный трюк, который ему очень хорошо запомнился.

Фокусник брал две карты размера $a \times b$ и $c \times d$ сантиметров. Далее он повторял следующее: располагал первую карту так, чтобы ее длинная сторона была горизонтальна, а вторую так, чтобы ее длинная сторона была вертикальна. Накладывал одну карту на другую так, чтобы один угол совпал. После этого та часть карт, которая была наложена, магическим образом исчезала. После этого он начинал с начала.

Так совпало, что в конце обе карты приняли форму квадратов, их размеры совпали, и после наложения они исчезли. Это и поразило маленького Стрэнджа больше всего.

Стрэндж не уверен, что правильно запомнил размеры карт, поэтому просит вас помочь проверить, мог ли фокус получиться с такими размерами, и если да, то сколько раз нужно будет наложить карты.

Формат входных данных

В единственной строке находятся 4 целых числа: a , b , c и d — размеры исходных карт ($1 \leq a, b, c, d \leq 10^{18}$).

Формат выходных данных

В первой строке выведите «YES», если фокус мог получиться, и «NO» иначе.

Если фокус мог получиться, во второй строке выведите количество раз, которое нужно наложить карты, прежде чем они исчезнут.

Примеры

<code>trick.in</code>	<code>trick.out</code>
12 6 8 9	YES 6
3 2 1 4	NO

Замечание

Пояснение к первому тесту

