

Задача А. Интерактивный замок

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1 секунда
Ограничение по памяти:	64 мегабайта

В настоящее время приобрели популярность интерактивные кодовые замки. Для того, чтобы протестировать сообразительность своих сотрудников, такой замок установили на дверь в офисе. Теперь работники в панике пытаются открыть его, но ничего не выходит.

Кодовый замок состоит из T секций, содержащих целые числа X_i ($10^2 \leq X_i \leq 10^4$), для каждого из которых необходимо отгадать делитель (кроме единицы) следующим образом:

- Вы называете какое-либо целое число (предполагаемый делитель);
- если выбранное Вами число является делителем числа X , то Вы получаете сообщение «YES» и переходите к отгадыванию следующего числа;
- если выбранное Вами число не является делителем числа X , то Вы получаете сообщение «NO», от числа X отнимается выбранное вами число и вы продолжаете отгадывать делитель уже измененного числа.

После вычитания отгадываемое число должно оставаться положительным, а также нельзя называть один и тот же делитель для текущего отгадываемого числа более одного раза. В случае соблюдения всех ограничений и удачного отгадывания хотя бы одного делителя для каждого из чисел дверь будет открыта.

Сотрудники бились над этой задачей в течение всего дня. Теперь им нужен тот, кто мог бы помочь им открыть дверь. Способны ли Вы на это?

Формат входных данных

В первой строке задается число T — количество чисел в кодовом замке ($1 \leq T \leq 500$).

Далее в качестве ответов на попытки угадать делитель необходимо считывать сообщения «YES» или «NO». Такое сообщение будет приходить после каждой попытки угадывания.

После каждой попытки угадать не забывайте выводить символ новой строки и делать flush-операцию (сброс буфера, например, `fflush(stdout)` в C++, `System.out.flush()` в Java, и `flush(output)` в Pascal).

В случае если Вы нарушите какие-либо ограничения протокола общения с интерактором, Ваша программа будет завершена автоматически.

Формат выходных данных

Необходимо выводить предполагаемые делители для каждого из чисел кодового замка. Таких попыток может быть несколько для каждого числа в зависимости от успешности отгадывания делителя.

Примеры

стандартный ввод	стандартный вывод
3	2
YES	2
NO	3
YES	2
YES	

Замечание

В тесте приведен пример корректной работы для кодового замка, состоящий из трех секций с числами 100, 101 и 102.

Задача В. Скип

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	3 секунды
Ограничение по памяти:	128 мегабайт

В последнее время крупные магазины все чаще стали предлагать своим покупателям различные дисконтные программы. И если в некоторых магазинах условия получения скидки запутанны и сложны, то в гипермаркете «Скип» все предельно ясно: каждый K -й товар в чеке достается покупателю бесплатно.

Студент всегда не прочь сэкономить. Но голова студента забита совсем другими вещами, поэтому он вспоминает о действующей системе скидок только тогда, когда все его N товаров уже выложены на конвейерную ленту перед кассой.

Кассир берёт товары с ленты по очереди. В этот момент студент может попросить кассира пока не пробивать взятый им товар, и переложить его в конец ленты. Кассир и вся очередь терпеливо ждут, пока студент переложит товар в конец ленты, после чего ситуация повторяется снова. Если студент решает не перекладывать товар, то он пробивается в чек и исчезает с ленты. Если какой-то товар студент уже перекладывал, то, чтобы не нервировать очередь за собой, он в любом случае разрешает кассиру пробить его в чек. Известно, что все товары, приобретаемые студентом, различны, хотя и могут иметь одинаковую стоимость.

Рассмотрим пример. Пусть стоимости товаров на ленте в порядке от ближайшего к кассиру до самого дальнего составляют 4, 1, 3 и 2 рубля, соответственно. Тогда при $K = 2$ студенту придется заплатить всего 7 рублей. Однако если бы товары лежали на конвейерной ленте в порядке 1, 3, 2, 4, то студент заплатил бы всего 3 рубля. Чтобы из первой ситуации получить вторую, студент может переложить товар стоимостью 4 рубля в конец ленты, сэкономив таким образом 4 рубля.

Зная цены товаров, их первоначальный порядок на ленте, а так же число K , помогите студенту определить минимальное количество денег, необходимое для приобретения всех товаров.

Формат входных данных

В первой строке заданы числа N ($1 \leq N \leq 500$) и K ($1 \leq K \leq N \leq 500$).

В следующей строке через пробел задано N чисел A_i ($1 \leq A_i \leq 10^6$) — стоимости товаров, приобретаемых студентом, в первоначальном порядке от ближайшего к кассиру до самого дальнего.

Формат выходных данных

Выведите единственное число — минимальную сумму денег, за которую студент может приобрести все товары.

Примеры

стандартный ввод	стандартный вывод
4 2 4 1 3 2	3
7 3 1 4 1 2 5 1 1	6
9 3 1 1 100 1 100 1 1 100 1	6

Задача С. Последовательность дробей

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта

Сегодня на занятиях по олимпиадному программированию была рассмотрена последовательность дробей, генерируемая следующим алгоритмом:

- Рассмотрим все пары чисел (A, B) таких, что $1 \leq A \leq B \leq 10^9$;
- Оставим из них только те, для которых $\text{НОД}(A, B) = 1$;
- Отсортируем их по возрастанию B , а при равных B — по возрастанию A ;
- Запишем все пары в виде дробей A/B .

Первые несколько дробей, генерируемых этим алгоритмом, выглядят следующим образом:
 $1/1, 1/2, 1/3, 2/3, 1/4, 3/4, 1/5, 2/5, 3/5, 4/5, 1/6, 5/6, 1/7, \dots$

Для того, чтобы занять детей после обеда, вожатая Анна придумала следующую игру:

- Каждый ребёнок получает целое число K ;
- В ответ ребёнок должен сообщить K -ую дробь в списке, сгенерированном по описанному выше алгоритму.

Поскольку дети хотят как можно быстрее пойти играть в футболчик, они решили написать программу, которая находит ответ для любого загаданного Анной K .

Вперёд! :)

Формат входных данных

В единственной строке ввода содержится целое число K — число, которое загадала Анна ($1 \leq K \leq 10^{11}$).

Формат выходных данных

Выведите дробь A/B , которая имеет порядковый номер K в сгенерированной последовательности. Гарантируется, что такая дробь существует.

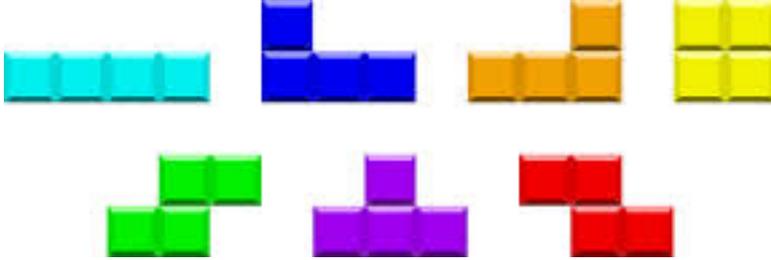
Пример

стандартный ввод	стандартный вывод
3	1/3

Задача D. Кирпичики-тетрамино

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

У Вас есть неограниченное число кирпичиков-тетрамино следующих типов:



Вам необходимо, используя любые из них, замостить прямоугольное поле размером $N \times M$. После этого в поле не должно остаться свободных мест, а кирпичики не должны выходить за границы поля или перекрывать друг друга.

Формат входных данных

В единственной строке заданы два числа N и M — размеры поля ($1 \leq N, M \leq 10$). Гарантируется, что размеры поля таковы, что его можно замостить с помощью кирпичиков-тетрамино.

Формат выходных данных

Выведите поле, замощенное кирпичиками-тетрамино, в виде N строк по M строчных латинских букв ('a' .. 'z'). Каждому использованному в решении кирпичику-тетрамино должна соответствовать своя уникальная буква.

Если существует несколько решений, выведите любое.

Примеры

стандартный ввод	стандартный вывод
4 2	bb cb cb cc
2 6	aaaabb ccccbb

Задача E. Мамонты

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта

В поисках ответа, от чего же вымерли мамонты, учёные обнаружили пещеру, содержащую в качестве наскальных рисунков целые числа. После долгих и бурных совещаний учёный Юрий предположил, что мамонты провели всё оставшееся в своей жизни время в тщетных попытках посчитать количество делителей у числа, являющегося произведением всех этих чисел.

Как известно, мамонты вымерли. Сможете ли вы написать программу, которая находит искомое количество, пока человечество не вымерло, как мамонты?

Формат входных данных

В первой строке содержится целое число N — количество чисел, записанных на стене пещеры ($1 \leq N \leq 10$). В следующей строке содержится N целых чисел A_i — числа, которые записаны на стене пещеры ($1 \leq A_i \leq 10^9$).

Формат выходных данных

Выведите одно число — количество делителей у числа, являющегося произведением всех чисел, записанных на стене пещеры. Гарантируется, что ответ не превысит 10^{18} .

Пример

стандартный ввод	стандартный вывод
4 2 6 10 22	40

Задача F. Благотворительность

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1 секунда
Ограничение по памяти:	64 мегабайта

Так получилось, что у нас оказался доступ к некоторым денежным ресурсам. И единственное что мы можем с ним делать это вносить благотворительные взносы. Всего есть N различных взносов, которые можно произвести. Никакой взнос нельзя использовать дважды, зато можно оформлять взносы в любом порядке. Каждый взнос требует A_i времени для того, чтобы его осуществить и ещё B_i времени, чтобы провести транзакцию, но для этого наше участие не требуется и в это время можно заниматься уже переводом другого взноса. Нам кажется, что как только пройдет первая по времени транзакция, то доступ исчезнет, но оставшиеся незавершенные транзакции все же со временем завершаться. Требуется узнать какое максимальное количество взносов можно успеть сделать прежде чем нас обнару... пройдет первая транзакция.

Формат входных данных

В первой строке задается число N — количество благотворительных взносов ($1 \leq N \leq 10^5$).

В следующей строке задается N пар чисел $A_i B_i$ — количество времени, которое потребуется для оформления взноса и для завершения транзакции соответственно ($1 \leq A_i \leq 10^5$, $1 \leq B_i \leq 10^9$).

Формат выходных данных

В единственной строке выведите максимальное количество успешно оформленных транзакций.

Пример

стандартный ввод	стандартный вывод
50	42
5 230	
6 203	
7 204	
7 223	
8 251	
5 192	
5 173	
7 251	
8 189	
7 175	
7 183	
5 237	
6 263	
7 235	
6 247	
6 220	
8 260	
7 203	
6 258	
9 231	
9 237	
5 195	
9 236	
7 189	
9 173	
8 204	
9 170	
9 269	
6 251	
6 241	
7 184	
8 211	
8 175	
6 218	
5 264	
8 256	
7 241	
5 220	
8 189	
7 212	
7 225	
5 202	
9 179	
6 170	
7 267	
5 209	
9 221	
8 209	
5 194	
5 209	

Задача G. Минимумы

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта

Недавно Славик изучил алгоритм, который может быстро находить минимум на интервале в последовательности чисел, если предварительно обработать эту последовательность. Чтобы как-то попрактиковаться в написание нового алгоритма, он решил найти сумму минимумов для всех возможных интервалов последовательности, т.е. $\sum_{i=1}^N \sum_{j=i}^N \min_{k=i}^j a_k$. Требуется помочь Славику проверить, а правильно ли он написал алгоритм и для этого вам нужно решить поставленную им задачу.

Формат входных данных

В первой строке задается число N — количество элементов последовательности ($1 \leq N \leq 10^4$).

В следующей строке задается N чисел A_i — заданная последовательность ($1 \leq A_i \leq 10^9$).

Формат выходных данных

Выведите сумму минимумов для всех интервалов последовательности.

Пример

стандартный ввод	стандартный вывод
4 1 3 2 1	14

Задача Н. Счастье

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта

Имеется дерево. Совсем обычное дерево — в нем N вершин, $N - 1$ ребро и между каждой парой вершин существует ровно один путь по этим ребрам. И вот по такому дереву целый день бегают счастье и генерирует радость. И вроде бы все ничего, но радости генерируется очень много. И нам, комиссии по учёту и распространению радости, требуется узнать сколько же радости было сгенерированно. Известно, что счастье пробежало путь между каждой парой вершин один раз в обоих направлениях. При этом количество радости сгенерированное при пробежке по ребру зависит от длины ребра и от того, под каким номером идет это ребро в списке пути, а именно равно $L_i * 2^K$, где L_i — длина ребра, а K — порядковый номер ребра (ребра в пути нумеруются с 0).

Формат входных данных

В первой строке задаются число N — количество вершину в дереве ($1 \leq N \leq 10^5$).

В следующих $N - 1$ строках задается по три числа $A_i B_i L_i$ — номера вершин, соединенных ребром, и длина ребра соответственно ($1 \leq A_i, B_i \leq N, 1 \leq L_i \leq 10^5$).

Формат выходных данных

В единственной строке выведите количество сгенерированной радости. Её может быть настолько много, что ответ лучше вывести по модулю $10^9 + 7$.

Пример

стандартный ввод	стандартный вывод
4 3 2 1 1 3 2 2 4 1	42

Задача I. Подсуммы

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1 секунда
Ограничение по памяти:	64 мегабайта

Не все числа одинаково полезны. Если, например, вам потребуется насобирать сумму как можно больше, то вам ни к чему использовать отрицательные числа. Но может получиться так, что и выбора не останется и придется их использовать. Да и вообще воспользуемся модулем, чтобы уровнять положительный и отрицательные числа. И совсем не требуется насобираться максимальную сумму, достаточно чтобы модуль суммы был больше M . Требуется найти количество способов выбрать подотрезок в последовательности с заданным свойством.

Формат входных данных

В первой строке задаются числа N M — количество чисел в последовательности и нижний предел модуля суммы ($1 \leq N \leq 10^5$, $1 \leq M \leq 10^9$).

В следующей строке задается N чисел A_i — числа последовательности ($|A_i| \leq 10^4$).

Формат выходных данных

В единственной строке выведите искомое количество подотрезков.

Пример

стандартный ввод	стандартный вывод
10 8 -2 9 3 6 3 8 -1 10 -6 7	42

Задача J. Несоответствие

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта

Для заданной последовательности маршрутов требуется найти первое несоответствие в них. Несоответствие это если нет возможности уложить все маршруты на двумерной плоскости.

Формат входных данных

В первой строке задается число N — количество маршрутов ($1 \leq N \leq 200$). Далее задаются N маршрутов $S_i L_{ij} D_{ij} A_{ij}$ — начальный город, а затем последовательность описания городов, которые были посещены в порядке движения по маршруту в формате расстояние, направление, номер города соответственно ($1 \leq S_i, L_{ij}, A_{ij} \leq 200, D_{ij} \in \{N, S, E, W\}$). Расстояние 0 обозначает конец маршрута. Не существует маршрутов, которые содержат один и то же город более одного раза. Каждый новый маршрут начинается с новой строки. Допустимо, что два города занимают одно и то же положение.

Формат выходных данных

Выведите одно число — наибольшее целое M такое, что первые M маршрутов не противоречат друг другу.

Пример

стандартный ввод	стандартный вывод
2 1 3 N 2 2 W 3 1 S 4 0 4 3 W 1 0	1

Задача К. Разница по XOR

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта

В начилии имеется N интервалов целых чисел, которые задаются своими границами $L_i R_i$. Скажем, что XOR-разница двух интервалов это сумма попарных XOR-разниц чисел этих интервалов, а XOR-разница двух чисел — количество единичных битов в XOR этих чисел.

Требуется найти сумму попарных XOR-разниц всех интервалов.

XOR — сложение по модулю 2, логическое сложение, исключающее ИЛИ, строгая дизъюнкция, xor, поразрядное дополнение, побитовый комплемент.

Формат входных данных

В первой строке задается число N — количество интервалов ($1 \leq N \leq 10^4$).

В следующих N строках задаются пары чисел L_i и R_i — границы интервала ($1 \leq L_i \leq R_i \leq 10^{18}$).

Формат выходных данных

Выведите искомую сумму по модулю $10^9 + 7$.

Пример

стандартный ввод	стандартный вывод
3	42
4 5	
8 10	
6 7	