

Задача А. Мега кузнечик

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

За один прыжок кузнечик может попасть из точки x в точки $x + 1, x + 2, \dots, x + k$.

Посчитайте количество способов кузнечика добраться из точки 0 до точки n по модулю $10^9 + 7$.

Формат входных данных

Даны два целых числа k и n ($1 \leq k \leq n \leq 10^5$) — максимальная длина прыжка кузнечика и куда он хочет попасть.

Формат выходных данных

Выведите одно число — количество способов попасть в точку n по модулю $10^9 + 7$.

Примеры

стандартный ввод	стандартный вывод
1 5	1
2 5	8
3 5	13

Задача В. Некто в Батуми

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

Некто переехал из Батуми в Москву. Нет. Это фейк. Некто остался жить в Батуми, и теперь ему надо добраться из дома до школы. Батуми — это табличка из n строчек и m столбцов (пронумерованных сверху вниз и слева направо, начиная с 1). Каждая клетка таблички представляет собой многоэтажный дом, на первом этаже которого находится ресторан сети «Нечто (и точка)».

Как известно, Некто очень любит поесть. И поэтому он обязательно поест в каждом «Нечто (и точка)», которое встретится у него на пути. Некто совсем недавно переехал в Батуми, поэтому ещё не успел исследовать все рестораны. И он очень боится отравиться, поэтому не хочет посещать рестораны, в которых он еще никогда не был. Для уже посещённых Нечто, Некто вычислил цену, за которую он поест в этом Нечто. К несчастью, в Грузии не такие большие премии за олимпиады, поэтому Некто решил начать экономить на еде. Но отказать себе в приёме пищи он не может. Единственный способ начать экономить — выбирать правильный маршрут от дома до школы.

Дом находится в левом верхнем углу Батуми, а школа — в правом нижнем. Для того чтобы не тратить впустую калории от купленной еды, Некто оптимизирует свои передвижения. В частности, он передвигается только от одного Нечто к одному из ближайших, и при том всегда движется в сторону школы. На практике это означает, что он умеет ходить вправо, вниз и вправо-вниз от текущего Нечто.

В последнее время Некто устроился экономистом на полставки, и теперь он слишком занят, чтобы самому рассчитывать свой дневной бюджет. Но он практичный человек и хочет знать, сколько ему придётся тратить денег на еду по пути от дома до школы. С таким вопросом он и обратился к вам.

Формат входных данных

В первой строке входных данных дается два натуральных числа — $n, m \leq 10^3$.

В следующих n строках дается описание города Батуми. В каждой строке для каждого дома, Нечто на первом этаже которого Некто уже посещал, содержится неотрицательное целое число $x \leq 10^6$. В противном случае там содержится число -1 . Гарантируется, что есть путь из начальной клетки в конечную.

Формат выходных данных

Выведите одно число — ответ на вопрос Некто.

Пример

стандартный ввод	стандартный вывод
4 4 3 5 -1 3 2 3 -1 4 1 2 2 4 1 -1 5 1	9

Задача С. НВП с восстановлением

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

Дана последовательность, требуется найти её наибольшую возрастающую подпоследовательность.

Формат входных данных

В первой строке входных данных задано число N — длина последовательности ($1 \leq N \leq 1000$). Во второй строке задается сама последовательность (разделитель — пробел). Элементы последовательности — целые числа, не превосходящие 10000 по модулю.

Формат выходных данных

В первой строке выведите длину НВП. В следующей строке требуется вывести наибольшую возрастающую подпоследовательность данной последовательности. Если таких подпоследовательностей несколько, необходимо вывести одну (любую) из них.

Пример

стандартный ввод	стандартный вывод
6 3 29 5 5 28 6	3 3 5 28

Задача D. НОПроблемо

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 0.5 секунд
Ограничение по памяти: 256 мегабайт

Даны две последовательности. Найдите длину их наибольшей общей подпоследовательности (подпоследовательность — это то, что можно получить из данной последовательности вычёркиванием некоторых элементов).

Формат входных данных

В первой строке входного файла записано число N — длина первой последовательности ($1 \leq N \leq 10^3$). Во второй строке записаны члены первой последовательности (через пробел) — целые числа, не превосходящие 10^4 по модулю. В третьей строке записано число M — длина второй последовательности ($1 \leq M \leq 10^3$). В четвёртой строке записаны члены второй последовательности (через пробел) — целые числа, не превосходящие 10^4 по модулю.

Формат выходных данных

В выходной файл требуется вывести единственное целое число: длину наибольшей общей подпоследовательности, или число 0, если такой не существует.

Примеры

стандартный ввод	стандартный вывод
3 1 2 3 4 2 1 3 5	2
3 1 2 3 3 1001 1002 1003	0

Задача Е. НВП

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 0.25 секунд
Ограничение по памяти: 256 мегабайт

Числовая последовательность задана рекуррентной формулой: $a_{i+1} = (k \cdot a_i + b) \bmod m$. Найдите её наибольшую возрастающую подпоследовательность. Если таких последовательностей несколько, можно вывести любую.

Формат входных данных

Программа получает на вход пять целых чисел: длину последовательности n ($1 \leq n \leq 10^5$), начальный элемент последовательности a_1 , параметры k, b, m для вычисления последующих членов последовательности ($1 \leq m \leq 10^4, 0 \leq k < m, 0 \leq b < m, 0 \leq a_1 < m$).

Формат выходных данных

На первой строке выходного файла вы должны вывести количество чисел в найденной вами наибольшей возрастающей подпоследовательности. На следующей строке выведите элементы подпоследовательности, разделяя их пробелами.

Примеры

стандартный ввод	стандартный вывод
5 41 2 1 100	3 41 67 71
7 1 2 1 10	4 1 3 5 7
7 2 2 1 10	3 1 3 5

Задача F. Шахтеры

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

На двух угольных шахтах работают шахтеры. Добыча угля — тяжелая работа, так что шахтерам необходимо доставлять еду прямо в шахты. Каждый раз, когда очередная порция продовольствия доставляется в шахту, шахтеры производят некоторое количество угля. Существует три типа еды для шахтеров: мясо, фрукты и бублики. Шахтеры любят разнообразие в еде и будут работать продуктивнее, если их диета будет разнообразной. Точнее, каждый раз, когда в шахту доставляется новая порция продовольствия, необходимо посмотреть на две предыдущие поставки (или меньше чем две, если их еще не было) и действовать по следующему правилу:

- если вся еда была одинаковой — будет произведена одна тонна угля
- если еды была двух типов — будет произведено две тонны угля
- если трех типов — три тонны угля

Тип и порядок поставок еды известен. Однако, можно выбрать, на какую из двух шахт какую поставку отправить. Поставка не может быть разделена и обязательно должна быть доставлена на первую или вторую шахту. Две шахты не обязательно должны получить одинаковое количество поставок (например, все поставки могут идти на одну шахту). По известному порядку и типу поставок необходимо написать программу, определяющую максимально возможное количество тонн угля, которое можно добыть на обеих шахтах, при наилучшем распределении поставок между первой и второй шахтой.

Формат входных данных

Первая строка содержит число N ($1 \leq N \leq 100\,000$) — количество поставок еды.

Вторая строка содержит N символов — типы поставок в том порядке, в котором они будут осуществляться. Каждый из символов будет большой латинской буквой M (мясо), F (фрукты) или B (бублики).

Формат выходных данных

Выведите единственное число — количество тонн угля, которое можно добыть.

Примеры

стандартный ввод	стандартный вывод
6 MBMFBB	12
16 MMBMBBVBMMMMBMB	29

Задача G. Пилообразная последовательность

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 2 секунды
Ограничение по памяти: 16 мегабайт

Назовем последовательность пилообразной, если каждый её элемент либо строго больше, либо строго меньше своих соседей. По данным числам n и k определите количество пилообразных последовательностей длины n , составленных из чисел $1, \dots, k$.

Формат входных данных

Программа получает на вход два натуральных числа n и k , $1 \leq n \leq 4000$, $1 \leq k \leq 4000$.

Формат выходных данных

Необходимо вывести остаток от деления количества искомых последовательностей на $10^9 + 7$.

Примеры

стандартный ввод	стандартный вывод
3 3	10
20 3	35422

Задача Н. Лестница

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

Археологи раскопали Древний Храм, ко входу в который ведет лестница, шириной в 1 (один) метр, из ступенек различной длины и высоты. Лестница построена из каменных блоков $1 \times 1 \times 1$ метр. Археологи хотят для удобства туристов, чтобы лестница состояла из меньшего количества ступенек N . Для этого они могут также устанавливать каменные блоки $1 \times 1 \times 1$. Какое минимальное количество блоков необходимо, чтобы сделать лестницу в N ступенек, если известны начальная длина и высота каждой ступеньки? (То есть требуется выровнять какие-то ступеньки с помощью блоков, чтобы уменьшить их число). Высоты и длины ступенек новой лестницы могут различаться.

Формат входных данных

В первой строке через пробел заданы два целых числа M и N ($1 \leq N < M \leq 100$). Далее идут M строк, содержащих пару целых чисел L и H — длина и высота i -й ступеньки соответственно ($1 \leq L, H \leq 101$). Ступеньки нумеруются снизу вверх.

Формат выходных данных

Выведите единственное число — минимальное число блоков.

Пример

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

Задача I. Линар и банк

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

Линар хочет положить для сохранности в банк «Центр-инвест» n мешков с деньгами, так как там самые выгодные условия. У Линара есть бесконечное количество бронемашин, в каждую машину вмещается k мешков. В i -м мешке c_i единиц денег.

Мешки расположены подряд, когда Линар будет складывать их, он положит первые k мешков в первую бронемашину, следующие k мешков во вторую бронемашину и т.д.

Прежде чем складывать мешки Линар хочет поменять их порядок так, чтобы сумма денег в первой бронемашине была максимальна, сумма денег во второй бронемашине была максимальна среди оставшихся мешков и т.д.

Линар может выбрать мешок и поставить в любое место в последовательности.

Скажите, какое минимальное кол-во действий надо сделать Линару, чтобы увезти все деньги.

Формат входных данных

Первая строка содержит два целых числа n и k ($1 \leq n, k \leq 10^5$) — количество мешков и вместительность бронемашин.

Во второй строке по n целых чисел c_i ($1 \leq c_i \leq 10^5$) — количество денег в каждом мешке.

Формат выходных данных

Скажите какое минимальное кол-во действий надо сделать Линару, чтобы увезти все деньги.

Примеры

стандартный ввод	стандартный вывод
6 2 10 4 5 3 11 100	3
5 2 100 78 72 95 2	1
6 2 72 6 52 52 99 53	3
7 2 78 29 5 93 31 10 29	2

Задача J. Футбол

Имя входного файла: `football.in`
Имя выходного файла: `football.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 64 мегабайта

Футбольный клуб «Инжир» не имеет проблем с финансированием, поэтому приобрел N полевых игроков. После долгих месяцев тренировок и тестов тренерский коллектив подробно изучил каждого игрока и собирается выбрать оптимальный стартовый состав.

Каждый игрок характеризуется тремя целыми неотрицательными числами: d_i — умение играть в защите, m_i — умение играть в полузащите и a_i — умение играть в нападении. Играть было решено по схеме 5-3-2. Это означает, что в основном составе будет 5 игроков защиты, 3 полузащитника и 2 нападающих. Цель тренерского штаба — выбрать 10 игроков так, чтобы сумма соответствующих умений была максимальна (для игрока обороны значение имеет только его умение играть в защите, аналогично для других позиций). Разумеется, каждый игрок может играть не более чем на одной позиции. Ваша задача — помочь тренерскому штабу подобрать оптимальный состав.

Формат входных данных

В первой строке входного файла находится одно целое число N — количество игроков в клубе. Следующие N строк файла содержат по три целых числа каждая d_i , m_i и a_i — умения футболиста с номером i играть в обороне, полузащите и нападении соответственно.

Формат выходных данных

Выходной файл должен содержать ровно три строки. Первая строка должна содержать пять чисел — номера игроков обороны. Вторая строка должна содержать ровно три числа — номера игроков полузащиты. Третья строка должна содержать ровно два числа — номера игроков нападения. Если оптимальных ответов несколько, можно вывести любой.

Примеры

football.in	football.out
10 1 2 3 2 3 4 3 4 5 4 5 6 5 6 7 6 7 8 7 8 9 8 9 10 9 10 1 10 1 2	10 9 8 7 6 3 4 5 1 2
10 1 5 1 3 2 2 3 3 5 2 1 3 4 4 5 2 0 2 1 1 1 4 5 1 1 3 3 4 5 4	10 5 2 6 7 1 8 9 3 4

Замечание

Задача К. Экспериментатор

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

На старости лет один профессор загорелся идеей исследования на прочность транзисторов «Кд521(2)», К сожалению, ему не удалось привлечь на помощь никого из коллег, поэтому проводить измерения придется самостоятельно. Но это не пугает профессора. В шкафу профессор обнаружил t транзисторов данной модели, оставшихся со старых времен, и решил использовать их для экспериментов.

После некоторых размышлений был выбран следующий способ проведения измерений: профессор собирается, перемещаясь по пожарной лестнице, сбрасывать транзисторы с различных этажей. Таким образом он планирует определить, при падении с какого минимального этажа транзистор разбивается. При этом профессор уверен, что транзистор не может выдержать падение с последнего этажа, однако падение с высоты человеческого роста (то есть когда профессор находится на первом этаже) не причиняет транзистору вреда. Известно, что все транзисторы абсолютно одинаковые, и если транзистор разбивается при падении с некоторого этажа, то он разбивается и при падении со всех этажей с большим номером. Разбившиеся транзисторы снова использовать нельзя, а если транзистор остался целым после падения, его можно использовать повторно. Для того, чтобы поднять оставшийся целым транзистор, профессору надо спуститься на первый этаж. Оказавшись на первом этаже, профессор может поднять все лежащие там транзисторы.

Годы профессора уже дают о себе знать, поэтому он хочет минимизировать суммарное расстояние, которое ему придется подниматься по лестнице. Но, возраст дает и определенные преимущества - сняв очки, профессор может с любого этажа определить, разбился транзистор или нет. Изначально профессор находится на первом этаже, и у него имеется m транзисторов. В доме, в котором живет профессор, n этажей. Найдите минимальное число этажей, которое профессору в худшем случае придется подниматься вверх по лестнице во время проведения экспериментов.

Формат входных данных

Во входном файле заданы два целых числа - высота дома $n(2 \leq n \leq 50)$ и количество транзисторов $m(1 \leq m \leq 10)$.

Формат выходных данных

В выходной файл выведите единственное число - минимальное расстояние в этажах, которое в худшем случае придется подниматься вверх по лестнице профессору во время эксперимента.

Примеры

стандартный ввод	стандартный вывод
2 1	0
3 1	1
18 1	136
17 3	23