

Содержание

Задача А. Наибольшая возрастающая подпоследовательность [2 секунды, 64 мегабайта]	2
Задача В. Рюкзак [2 секунды, 256 мегабайт]	3
Задача С. Рюкзак 2 [2 секунды, 256 мегабайт]	4
Задача D. Прогулка [2 секунды, 256 мегабайт]	5
Задача Е. Коды, сохраняющие порядок [2 секунды, 256 мегабайт]	7
Задача F. Ciel и гондолы [4 секунды, 512 мегабайт]	8
Задача G. Покупка участков [0.2 секунды, 256 мегабайт]	10
Задача H. Две лесопилки [0.1 секунды, 64 мегабайта]	11
Задача I. Мостостроение [2 секунды, 64 мегабайта]	12
Задача J. Конфеты и Камни [20 секунд, 45 мегабайт]	13

Задача А. Наибольшая возрастающая подпоследовательность [2 секунды, 64 мегабайта]

Дана последовательность, требуется найти её наибольшую возрастающую подпоследовательность.

Формат входных данных

В первой строке входных данных задано целое число n — длина последовательности ($1 \leq n \leq 300\,000$). Во второй строке задается сама последовательность. Числа разделяются пробелом. Элементы последовательности — целые числа, не превосходящие 10^9 по модулю.

Формат выходных данных

В первой строке выведите длину наибольшей возрастающей подпоследовательности, а во второй строке выведите через пробел самую наибольшую возрастающую подпоследовательность данной последовательности. Если ответов несколько — выведите любой.

Пример

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

Задача В. Рюкзак [2 секунды, 256 мегабайт]

Даша собралась в путешествие. Чтобы туда отправиться, она хочет собрать вещи в Рюкзачок. У неё есть n предметов, пронумерованные от 1 до n , каждый предмет имеет вес w_i и полезность в путешествии v_i .

У Рюкзачка есть максимальный вес W , который он может выдержать. Какую наибольшую по суммарной полезности группу предметов Даша может с собой унести?

Формат входных данных

Первая строка содержит целые числа n и W ($1 \leq n \leq 100$, $1 \leq W \leq 10^5$) — количество предметов и максимальная вместимость Рюкзачка.

Следующие n строк содержат целые числа w_i и v_i ($1 \leq w_i \leq W$, $1 \leq v_i \leq 10^9$) — веса предметов и их ценности.

Формат выходных данных

Выведите одно число — максимальную суммарную ценность.

Примеры

стандартный ввод	стандартный вывод
3 8 3 30 4 50 5 60	90
5 5 1 1000000000 1 1000000000 1 1000000000 1 1000000000 1 1000000000	5000000000
6 15 6 5 5 6 6 4 6 6 3 5 7 2	17

Замечание

В первом примере следует взять первый и третий предмет.

Задача С. Рюкзак 2 [2 секунды, 256 мегабайт]

Даша собралась в *ещё одно* путешествие. Чтобы туда отправится, она хочет собрать вещи в Рюкзачок.

У неё есть n предметов, пронумерованные от 1 до n , каждый предмет имеет вес w_i и полезность в путешествии v_i .

У Рюкзачка есть максимальный вес W , который он может выдержать. Какую наибольшую по суммарной полезности группу предметов Даша может с собой унести?

Формат входных данных

Первая строка содержит целые числа n и W ($1 \leq n \leq 100$, $1 \leq W \leq 10^9$) — количество предметов и максимальная вместимость Рюкзачка.

Следующие n строк содержат целые числа w_i и v_i ($1 \leq w_i \leq W$, $1 \leq v_i \leq 10^3$) — веса предметов и их ценности.

Формат выходных данных

Выведите одно число — максимальную суммарную ценность.

Примеры

стандартный ввод	стандартный вывод
3 8 3 30 4 50 5 60	90
1 1000000000 1000000000 10	10
6 15 6 5 5 6 6 4 6 6 3 5 7 2	17

Замечание

В первом примере следует взять первый и третий предмет.

Задача D. Прогулка [2 секунды, 256 мегабайт]

Собрав вещи, Даша отправилась в путешествие. Можно считать, что её путешествие проходило на ориентированном графе из n вершин, пронумерованных от 1 до n .

Начав своё путешествие в какой-то произвольной вершине, Даша k раз переходила по какому-то ориентированному ребру из текущей вершины.

Воспоминания уже стёрлись из её головы, и она не очень помнит свой маршрут. Однако ей хочется узнать, сколькими способами она могла прогуляться по графу, пройдя ровно k рёбер. Так как это количество может быть очень большим, выведите его по модулю $10^9 + 7$. Обратите внимание, что пути, насколько раз проходящие по одному и тому же ребру, тоже учитываются.

Формат входных данных

Первая строка содержит целые числа n и k ($1 \leq n \leq 50$, $1 \leq k \leq 10^{18}$) — количество вершин и длина пути.

Затем для каждого i от 1 до n следует строка, содержащая целые числа $a_{i,1}, a_{i,2}, \dots, a_{i,n}$ ($0 \leq a_{i,j} \leq 1$, $a_{i,i} = 0$).

В случае, если из вершины i есть ориентированное ребро в j , то $a_{i,j} = 1$, иначе $a_{i,j} = 0$.

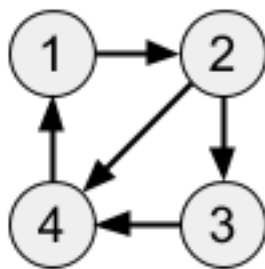
Формат выходных данных

Выведите одно число — количество возможных путей по модулю $10^9 + 7$.

Примеры

стандартный ввод	стандартный вывод
4 2 0 1 0 0 0 0 1 1 0 0 0 1 1 0 0 0	6
3 3 0 1 0 1 0 1 0 0 0	3
6 2 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 1 0 0 0 0 0 0	1
1 1 0	0

Замечание



Картинка иллюстрирует граф в первом примере.

Возможные пути Даши выглядят следующим образом:

- $1 \rightarrow 2 \rightarrow 3$
- $1 \rightarrow 2 \rightarrow 4$
- $2 \rightarrow 3 \rightarrow 4$
- $2 \rightarrow 4 \rightarrow 1$
- $3 \rightarrow 4 \rightarrow 1$
- $4 \rightarrow 1 \rightarrow 2$

Задача Е. Коды, сохраняющие порядок [2 секунды, 256 мегабайт]

Двоичный код — это код, где каждому символу сопоставляется последовательность из единиц и нулей. Код называется префиксным, если ни одно кодовое слово не является префиксом другого. Код называется сохраняющим порядок, если лексикографический порядок кодовых слов совпадает с алфавитным порядком символов.

Рассмотрим текст над алфавитом, содержащим n символов, в котором a_1 раз встречается первый символ, a_2 раз встречается второй символ, ..., a_n раз встречается n -й символ. Длина текста после кодирования его префиксным кодом, где первому символу сопоставлена строка длины l_1 , второму — строка длины l_2 , и т. д., будет равна $a_1l_1 + a_2l_2 + \dots + a_nl_n$.

Требуется найти сохраняющий порядок префиксный код, минимизирующий длину закодированного текста.

Формат входных данных

Первая строка содержит число n — число символов в алфавите ($2 \leq n \leq 2000$). Следующая строка содержит n целых чисел — сколько раз каждый символ встречается в тексте: a_1, a_2, \dots, a_n . Числа положительные и не превосходят 10^9 .

Формат выходных данных

Выведите n двоичных последовательностей — искомый код.

Пример

codes.in	codes.out
5	00
1 8 2 3 1	01
	10
	110
	111

Задача F. Ciel и гондолы [4 секунды, 512 мегабайт]

Лиса Ciel зашла в парк аттракционов. И вот, она в очереди на колесо обозрения. В очереди стоит n людей (хотя нет, скорее лис): мы будем считать, что первая лиса стоит в начале очереди, а n -ая лиса стоит в хвосте очереди.

Всего имеется k гондол, мы распределяем лис по гондолам следующим образом:

- Когда подплывает первая гондола, q_1 лис переходят из начала очереди в подплывшую гондолу.
- Затем, когда подплывает вторая гондола, q_2 лис из начала оставшейся очереди переходит в эту гондолу.
- ...
- Оставшиеся q_k лис идут с последнюю (k -ую) гондолу.

Обратите внимание, что числа q_1, q_2, \dots, q_k должны быть положительными. Из условия следует, что $\sum_{i=1}^k q_i = n$ и $q_i > 0$.

Вы знаете как лисам не хочется задерживаться в гондолах с незнакомцами. Итак, Ваша задача — найти оптимальный способ размещения (то есть определить оптимальную последовательность q), чтобы угодить всем. Для каждой пары лис i и j задано значение u_{ij} , обозначающее степень незнакомости. Можете считать, что $u_{ij} = u_{ji}$ для всех i, j ($1 \leq i, j \leq n$) и что $u_{ii} = 0$ для всех i ($1 \leq i \leq n$). Тогда значение незнакомости в гондоле определяется как сумма значений незнакомости между всеми парами лис, которые находятся в этой гондоле. Общее значение незнакомости определяется как сумма значений незнакомости по всем гондолам.

Помогите лисе Ciel найти минимальное возможное значение общей незнакомости при некотором оптимальном распределении лис по гондолам.

Формат входных данных

В первой строке даны два целых числа n и k ($1 \leq n \leq 4000$ and $1 \leq k \leq \min(n, 800)$) — количество лис в очереди и количество гондол. В следующих n строках записано по n целых чисел — матрица u , ($0 \leq u_{ij} \leq 9$, $u_{ij} = u_{ji}$ и $u_{ii} = 0$).

Пожалуйста, используйте методы быстрого чтения (например, для Java используйте `BufferedReader` вместо `Scanner`).

Формат выходных данных

Выведите целое число — минимальное возможное значение общей незнакомости.

Примеры

стандартный ввод	стандартный вывод
5 2 0 0 1 1 1 0 0 1 1 1 1 1 0 0 0 1 1 0 0 0 1 1 0 0 0	0
8 3 0 1 1 1 1 1 1 1 1 0 1 1 1 1 1 1 1 1 0 1 1 1 1 1 1 1 1 0 1 1 1 1 1 1 1 1 0 1 1 1 1 1 1 1 1 0 1 1 1 1 1 1 1 1 0 1 1 1 1 1 1 1 1 0	7
3 2 0 2 0 2 0 3 0 3 0	2

Замечание

В первом примере можно распределить лис вот так: 1, 2 идут в одну гондолу, 3, 4, 5 идут в другую гондолу.

Во втором примере оптимальное распределение таково: 1, 2, 3 | 4, 5, 6 | 7, 8.

Задача G. Покупка участков [0.2 секунды, 256 мегабайт]

Фермер Паша собирается купить как можно больше земли для своей фермы. Он присмотрел n ($1 \leq n \leq 50\,000$) прямоугольных участков, i -й из которых имеет размер $w_i \times h_i$, где $1 \leq w_i, h_i \leq 10^6$.

Если фермер Паша покупает участок земли, он платит один доллар за каждую единицу площади участка. Но если он покупает сразу несколько участков, то действует система скидок. А именно, он платит $\max w \cdot \max h$ за множество участков, где максимум берется по всем покупаемым участкам. Например, если фермер Паша покупает участки 5×3 и 3×5 , он заплатит $5 \times 5 = 25$ долларов.

Фермер Паша хочет купить все участки и потратить минимальную сумму. Помогите ему выбрать, как разбить участки на наборы, чтобы купить их и заплатить суммарно минимальную сумму.

Формат входных данных

Первая строка содержит число n . Следующие n строк содержат по два целых числа: размеры участков.

Формат выходных данных

Выведите одно число: минимальную сумму, которую сможет потратить фермер Паша.

Примеры

стандартный ввод	стандартный вывод
4 100 1 15 15 20 5 1 100	500

Задача Н. Две лесопилки [0.1 секунды, 64 мегабайта]

От вершины до подножья холма растет N старых деревьев. Районная администрация решила в санитарных целях срубить эти деревья, а чтобы снизить стоимость мероприятия перевезти все древесину на лесопилки.

Деревья могут быть перевезены только в одном направлении – вниз. У подножья холма находится лесопилка, а также две дополнительные лесопилки могут быть построены на холме вдоль дороги. Вам предстоит определить, где наиболее выгодно построить эти лесопилки, чтобы минимизировать стоимость транспортировки древесины. Перевозка 1 килограмма древесины на 1 метр стоит 1 копейку.

Формат входных данных

Первая строка входного файла содержит натуральное число N – количество деревьев ($1 \leq N \leq 20000$). Деревья занумерованы от 1 до N начиная с вершины холма. Следующие N линий содержат по два целых числа w_i и d_i ($1 \leq w_i, d_i \leq 10000$) – вес дерева номер i и расстояние между деревьями d_i и d_{i+1} . Последнее из этих чисел d_n задает расстояние от нижнего дерева до лесопилки.

Формат выходных данных

Выведите единственное число – минимальную стоимость.

Примеры

стандартный ввод	стандартный вывод
9 1 2 2 1 3 3 1 1 3 2 1 6 2 1 1 2 1 1	26

Задача I. Мостостроение [2 секунды, 64 мегабайта]

Давным давно, в 2009-м году...

В деревне Зайкино регулярно идут проливные дожди, в результате чего речка Дубровка, которую обычно можно просто перешагнуть, выходит из берегов. Чтобы можно было перейти разлившуюся реку, планируется построить плавучий мост из брёвен, оставшихся от строительства бани бизнесмена, поселившегося неподалёку.

Все оставшиеся брёвна имеют одинаковую толщину. При этом есть x брёвен длины a и y брёвен длины b .

Построенный мост должен состоять из l рядов, каждый из которых составлен из одного или нескольких брёвен. Пилить брёвна нельзя, так как последняя пила утонула при разливе Дубровки.

Главный инженер хочет построить мост максимальной возможной ширины. Ширина моста определяется по минимальной ширине ряда брёвен в нём.

Например, если нужно построить мост из семи рядов, и при этом есть шесть брёвен длины 3 и десять брёвен длины 2, то можно построить мост ширины 5.

Формат входных данных

Ввод состоит из одного или нескольких тестовых случаев. Каждый тестовый случай состоит из пяти целых положительных чисел x , a , y , b и l . Каждое число не превосходит 500. Общее количество брёвен в каждом тестовом случае не меньше l .

Обозначим $d = \max(x, a, y, b, l)$. Гарантируется, что сумма d по всем тестам не превосходит 5000.

Формат выходных данных

Примеры

bridge.in	bridge.out
6 3 10 2 7	5
10 7 20 9 25	9
106 126 135 28 137	112

Задача J. Конфеты и Камни [20 секунд, 45 мегабайт]

Мальчик Геральд и его тренер Миша играют в интересную игру. В начале игры имеются куча из n конфет и куча из m камней. Геральд и Миша ходят по очереди, первым ходит Миша. Миша на своем ходу проверяет, сколько на данный момент Геральд съел конфет и камней. Пусть Геральд съел a конфет и b камней. Тогда Миша начисляет Геральду $f(a, b)$ призовых очков. Геральд же на своем ходу съедает либо одну конфету из кучи с конфетами, либо один камень из кучи с камнями. Когда Миша обнаруживает, что Геральд съел все, кроме одной конфеты и одного камня, он последний раз начисляет очки и игра заканчивается. Опустошать ни ту, ни другую кучку Геральд не имеет права. Расскажите Геральду, как ему играть, чтобы получить наибольшее количество очков: требуется найти один из возможных оптимальных вариантов игры для Геральда.

Формат входных данных

В первой строке содержатся три целых числа n, m, p ($1 \leq n, m \leq 20000$, $1 \leq p \leq 10^9$). Во второй строке находятся n целых чисел x_0, x_1, \dots, x_{n-1} ($0 \leq x_i \leq 20000$). В третьей строке находятся m целых чисел y_0, y_1, \dots, y_{m-1} ($0 \leq y_i \leq 20000$). Величина $f(a, b)$ вычисляется, как остаток от деления суммы $x_a + y_b$ на число p .

Формат выходных данных

В первой строке выведите единственное число: максимальное количество призовых очков, которое может заработать Геральд. Во второй строке выведите строку из $n + m - 2$ символов, каждый из которых — это «C» или «S», i -ый символ должен быть «C», если на своем i -ом ходу Геральд должен съесть конфету, и «S», если камень.

Примеры

стандартный ввод	стандартный вывод
2 2 10 0 0 0 1	2 SC
3 3 10 0 2 0 0 0 2	10 CSSC
3 3 2 0 1 1 1 1 0	4 SCSC

Замечание

В первом тесте, если на первом ходу Геральд съест камень, то после него он получит одно очко, а если конфету — то ноль. Перед первым своим ходом Геральд получит в любом случае 0 очков, а после второго — в любом случае 1. Таким образом, максимальное количество очков, которое может получить Геральд равно 2, и для этого надо съесть сначала камень, потом конфету.