

Задача А. 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 входного файла

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 выходного файла

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

Система оценки

Тесты состоят из четырех групп. Во всех группах $0 \leq d_i, m_i, a_i \leq 10^7$.

0. Тесты 1–2, из условия, оцениваются в 0 баллов.
1. В тестах этой группы $N = 10$. Эта группа оценивается в 20 баллов, баллы начисляются только при прохождении всех тестов группы.
2. В тестах этой группы $10 \leq N \leq 40$. Эта группа также оценивается в 20 баллов, они начисляются только при прохождении всех тестов группы.
3. В тестах этой группы $10 \leq N \leq 10\,000$. Эта группа также оценивается в 20 баллов, они начисляются только при прохождении всех тестов группы.
4. Offline-группа, $10 \leq N \leq 500\,000$. Баллы за тесты этой группы начисляются только при прохождении всех тестов всех предыдущих групп. Некоторые тесты этой группы объединяются в подгруппы, тесты за каждую подгруппу ставятся только при прохождении всех тестов подгруппы.

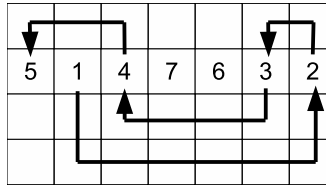
Задача В. Микроконтроллеры

Имя входного файла: `stripe.in`
Имя выходного файла: `stripe.out`
Ограничение по времени: 1 секунда
Ограничение по памяти: 128 мегабайт

Михаил придумал решение задачи аппаратного кодирования видео с помощью последовательно соединенных микроконтроллеров. Каждый микроконтроллер выполняет определенную часть задачи, после чего передает данные следующему микроконтроллеру (получается некий конвейер из микроконтроллеров). В устройстве используется N микроконтроллеров, которые должны быть соединены последовательно: первый со вторым, второй с третьим и т. д. По задумке, микроконтроллеры располагались на плате в одну горизонтальную линию.

Михаил заказал платы с микроконтроллерами на фабрике, однако получилось так, что микроконтроллеры вместо того, чтобы стоять последовательно, оказались в хаотичном порядке! Поскольку заказ был довольно дорогим, Михаил решил максимально использовать имеющуюся плату, т. е. последовательно соединить дорожками наибольшее количество микроконтроллеров в цепочку вида $1 - 2 - \dots - m$. Оставшуюся часть придется заказать заново.

Плата, на которой расположены микроконтроллеры, будет односторонней (все дорожки расположены на одной плоскости и, естественно, не могут пересекаться). Если в микроконтроллер дорожка со входным сигналом входит сверху, то дорожка с выходным сигналом должна выходить из него обязательно снизу, и наоборот. Микроконтроллеры расположены вплотную друг к другу (проложить между ними дорожку нельзя). Обойти линию микроконтроллеров дорожкой слева или справа также нельзя (только сверху или снизу). Сверху и снизу от линии микроконтроллеров плата имеет достаточные размеры и позволяет проложить любое число дорожек. Помогите Михаилу определить, какое максимальное количество последовательных микроконтроллеров удастся соединить, начиная с первого.



Формат входного файла

В первой строке входного файла задано число N — длина полоски из микроконтроллеров.

Во второй строке задана перестановка из N чисел — порядок расположения микроконтроллеров.

Формат выходного файла

Выведите единственное число — максимальное количество микроконтроллеров, которые удастся соединить последовательно, начиная с первого.

Пример

<code>stripe.in</code>	<code>stripe.out</code>
7	5
5 1 4 7 6 3 2	

Система оценки

Тесты состоят из трех групп.

0. Тест 1, из условия, оценивается в 0 баллов.
1. В тестах этой группы количество микроконтроллеров N не превосходит 5 000. Эта группа оценивается в 30 баллов, баллы начисляются только при прохождении всех тестов группы.
2. В тестах этой группы количество микроконтроллеров N не превосходит 100 000. Эта группа оценивается в 40 баллов, баллы начисляются только при прохождении всех тестов этой и предыдущей группы.
3. Offline-группа. В тестах этой группы количество микроконтроллеров N не превосходит 1 000 000. Эта группа оценивается в 30 баллов, баллы начисляются только при прохождении всех тестов из всех групп.

Задача С. Уголки

Имя входного файла: `corners.in`
Имя выходного файла: `corners.out`
Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта

После почти повсеместного закрытия казино у нас в стране, предприимчивые дельцы стали создавать клубы по игре в различные азартные, но пока еще не запрещенные игры. Одной из таких игр стала игра «Уголки».

Игроку выдается карточка, на которой нарисована таблица, состоящая из M строк и N столбцов, каждым элементом которой является целое число. Строки и столбцы пронумерованы, начиная с единицы, клетка с номером $(1, 1)$ находится в левом верхнем углу.

Игрок может сделать несколько ходов по следующим правилам. Первым ходом он может указать на любой элемент таблицы (r_1, c_1) , и все числа в прямоугольнике $(1, 1)-(r_1, c_1)$ меняют свой знак на противоположный. Здесь первое число обозначает номер строки, а второе — номер столбца. Следующие ходы игрока аналогичны, то есть выбирается прямоугольник $(1, 1)-(r_i, c_i)$ и все числа опять меняют знак, однако каждый следующий прямоугольник должен полностью содержать предыдущий выбранный, но не совпадать с ним. Количество ходов при этом естественным образом ограничено размерами таблицы. При желании ходы можно и не выполнять совсем.

Например, если таблица состоит из 10 строк и 9 столбцов, то возможна в том числе такая последовательность ходов: $(2, 2)-(4, 4)-(4, 6)-(5, 6)$.

После того как все желаемые ходы выполнены, подсчитывается результат игры — сумма значений элементов преобразованной таблицы. Если сумма положительная, то игрок получает от заведения соответствующую сумму денег, а если отрицательная — платит ее.

Помогите игроку добиться наилучшего результата: выиграть как можно больше или хотя бы проиграть как можно меньше.

Формат входного файла

В первой строке входного файла заданы два натуральных числа M и N ($M \leq 1000$, $N \leq 1000$). Каждая из следующих M строк содержит N целых чисел, по модулю не превышающих 1000, — элементы таблицы.

Формат выходного файла

В первой строке выходного файла выведите максимальную сумму, которую можно получить с помощью оптимальной стратегии игры по описанным правилам. Во второй строке выведите целое число K — количество действий в алгоритме, дающем наилучший результат. В следующих K строках выведите последовательность из K пар чисел (r_i, c_i) , описывающих сам алгоритм.

Если дающих наилучший результат алгоритмов несколько, выведите любой.

Примеры

<code>corners.in</code>	<code>corners.out</code>
2 3 3 6 4 1 -2 -7	21 2 1 3 2 3
4 4 2 -1 -5 -1 1 -3 -3 -1 0 -5 1 1 -1 -1 0 2	22 2 2 1 4 3

Система оценки

Тесты состоят из трех групп.

0. Тесты 1–2, из условия, оцениваются в 0 баллов.
1. В тестах этой группы $M = 1$, $1 \leq N \leq 1000$. Эта группа оценивается в 30 баллов, баллы начисляются только при прохождении всех тестов группы.
2. В тестах этой группы $1 \leq M \leq 100$, $1 \leq N \leq 100$. Эта группа также оценивается в 30 баллов, они начисляются только при прохождении всех тестов группы.
3. Offline-группа. Баллы за тесты этой группы начисляются только при прохождении всех тестов 1-й и 2-й групп. Каждый тест оценивается независимо от других.

Задача D. Банкротство

Имя входного файла: `bankrupt.in`
Имя выходного файла: `bankrupt.out`
Ограничение по времени: 1 секунда
Ограничение по памяти: 128 мегабайт

Недавно у градообразующего предприятия города Флатвилля сменилось руководство. Новый генеральный директор не очень-то разбирается в техническом производстве, поэтому заключает все договоры подряд. После этого (чтобы спасти положение) в дело вступает директор по расписаниям, который решает, какие из этих заказов будут выполнены, а за какие придется платить неустойку.

Каждый заказ обладает тремя параметрами: d_i — максимальный номер дня, к которому этот заказ должен быть выполнен; a_i — доход, который получает завод, если выполняет заказ в срок; b_i — неустойка, которую выплачивает завод в случае, если заказ не был выполнен к дню d_i включительно. Нумерация дней ведется со следующего после заключения договоров дня. Известно, что на выполнение одного заказа завод тратит непрерывный отрезок из ровно K дней, причем в один момент времени завод может быть занят только одним заказом.

В итоге перед директором по расписаниям стоит непростая задача: выбрать какие заказы выполнить (и в какое время), а за какие проще сразу заплатить неустойку.

Конкурирующая фирма хочет данное предприятие разорить. У директора конкурирующей фирмы есть возможность перехватить любое число заказов. Все остальные заказы достанутся предприятию (даже если они явно ему не выгодны). Требуется помочь директору конкурирующей фирмы выбрать такое подмножество имеющихся потенциальных заказов для перехвата, чтобы при оптимальной работе директора по расписаниям (то есть при выборе им наилучшей стратегии выполнения оставшихся заказов) доход предприятия был минимальным (он может быть и отрицательным).

Формат входного файла

В первой строке входного файла записано одно число G — номер группы тестов.

Во второй строке задано два числа N и K .

Следующие N строк содержат описание заказа: d_i, a_i, b_i . Числа K, a_i, b_i целые положительные и не превосходят 1000. Числа d_i целые положительные и не превосходят 100000.

Формат выходного файла

В первой строке выведите значение искомого дохода (убытка) предприятия. Следующая строка должна содержать ровно N нулей и единиц, разделенных пробелами. Если на i -й позиции стоит 0, это означает, что этот заказ должна перехватить конкурирующая фирма; если же стоит 1, то заказ достанется предприятию.

Примеры

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

Система оценки

Тесты состоят из восьми групп.

0. Тесты 1–2, из условия, оцениваются в 0 баллов.

1. В тестах этой группы $1 \leq N \leq 100\,000, K = 1, d_i = 1, a_i = b_i$ для всех i . Эта группа оценивается в 10 баллов, баллы начисляются только при прохождении всех тестов группы.
2. В тестах этой группы $1 \leq N \leq 100\,000, K = 1, d_i = 1$ для всех i . Эта группа оценивается в 10 баллов, баллы начисляются только при прохождении всех тестов группы.
3. В тестах этой группы $1 \leq N \leq 100\,000, a_i = b_i, d_i = d_j$ для всех i, j . То есть все заказы имеют одинаковый срок выполнения, и для всех заказов прибыль равна неустойке. Эта группа оценивается в 10 баллов, баллы начисляются только при прохождении всех тестов группы.
4. В тестах этой группы $1 \leq N \leq 1000, d_i = d_j$ для всех i, j . Эта группа оценивается в 10 баллов, баллы начисляются только при прохождении всех тестов группы.
5. В тестах этой группы $1 \leq N \leq 100\,000, d_i = d_j$ для всех i, j . Эта группа оценивается в 10 баллов, баллы начисляются только при прохождении всех тестов группы.
6. В тестах этой группы $1 \leq N \leq 15$. Эта группа оценивается в 10 баллов, баллы начисляются только при прохождении всех тестов группы.
7. Offline-группа. В тестах этой группы $1 \leq N \leq 100, a_i = b_i$. На этой группе будут тестироваться только решения, прошедшие хотя бы одну из групп 1 – 6. Группа состоит из 20 тестов, за каждый тест участник получает либо 0, либо 1 балл. Участник получает за тест 1 балл, если его ответ корректен и никакое другое решение (другого участника или жюри) не находит ответ лучше.
8. Offline-группа. В тестах этой группы $1 \leq N \leq 100$. На этой группе будут тестироваться только решения, прошедшие хотя бы одну из групп 1 – 6. Группа состоит из 20 тестов, за каждый тест участник получает либо 0, либо 1 балл. Участник получает за тест 1 балл, если его ответ корректен и никакое другое решение (другого участника или жюри) не находит ответ лучше.