

Задача А. Максимальный поток минимальной стоимости

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

Задан ориентированный граф, каждое ребро которого обладает пропускной способностью и стоимостью. Найдите максимальный поток минимальной стоимости из вершины с номером 1 в вершину с номером n .

Формат входных данных

Первая строка входного файла содержит n и m — количество вершин и количество ребер графа ($2 \leq n \leq 100$, $0 \leq m \leq 1000$). Следующие m строк содержат по четыре целых числа: номера вершин, которые соединяет соответствующее ребро графа, его пропускную способность и его стоимость. Пропускные способности и стоимости неотрицательны и не превосходят 10^5 .

Формат выходных данных

В выходной файл выведите одно число — цену максимального потока минимальной стоимости из вершины с номером 1 в вершину с номером n . Ответ не превышает $2^{63} - 1$.

Пример

стандартный ввод	стандартный вывод
4 5 1 2 1 2 1 3 2 2 3 2 1 1 2 4 2 1 3 4 2 3	12

Задача В. Задача о назначениях

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1.5 секунд
Ограничение по памяти:	256 мегабайт

Дана целочисленная матрица C размера $n \times n$. Требуется выбрать n ячеек так, чтобы в каждой строке и каждом столбце была выбрана ровно одна ячейка, а сумма значений в выбранных ячейках была минимальна.

Формат входных данных

Первая строка входного файла содержит n ($2 \leq n \leq 300$). Каждая из последующих n строк содержит по n чисел: C_{ij} . Все значения во входном файле неотрицательны и не превосходят 10^6 .

Формат выходных данных

В первую строку выходного файла выведите одно число — искомая минимизируемая величина. Далее выведите n строк по два числа в каждой — номер строки и столбца клетки, участвующей в оптимальном назначении.

Пары чисел можно выводить в произвольном порядке.

Пример

стандартный ввод	стандартный вывод
3	3
3 2 1	2 1
1 3 2	3 2
2 1 3	1 3

Замечание

Обратите внимание, что эту задачу можно сдать с помощью алгоритма за $O(n^3)$, без использования Венгерского алгоритма, а используя только максимальный поток минимальной стоимости.

Задача С. Командная олимпиада

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

Совсем скоро начнется первый тур очередной всероссийской командной олимпиады школьников по палеонтологии (ВКОШП). На олимпиаду приехали команды из n школ, от каждой школы приехало ровно по две команды. Команды уже заняли свои места, когда обнаружилось, что некоторые команды из одной школы сидят слишком близко. Перед организаторами олимпиады встала серьезная задача — пересадить участников олимпиады.

Стол, за которыми сидят команды, расставлены в один ряд. Расстояние между рабочими местами соседних команд оказалось равно 10 метрам. Организаторы хотят, чтобы минимальное расстояние между двумя командами из одной школы было как можно больше.

Пересаживая команду, организатором необходимо перенести на новое место все оборудование, приготовленное командой для исследований. Поэтому организаторы хотят пересадить команды так, чтобы сумма расстояний между старыми и новыми рабочими местами команд была как можно меньше.

Например, пусть в соревновании принимают участие по две команды школ 1, 2, 3 и 4. Пусть исходно команды распределены по рабочим местам следующим образом: 1, 3, 2, 2, 1, 4, 4, 3 (для каждой команды указан номер школы, которую она представляет). При таком распределении по рабочим местам команды из школы 2 сидят слишком близко, как и команды из школы 4. Пересадив команды в следующем порядке: 1, 3, 2, 4, 1, 3, 2, 4, жюри может добиться своего: команды из одной школы сидят на местах, расстояние между которыми не меньше 40 м, большего расстояния добиться нельзя. Сумма расстояний между старыми и новыми позициями для данного примера равна $0 + 0 + 0 + 20 + 0 + 20 + 30 + 10 = 80$ м, для исходного распределения команд она минимальна.

Вам задано исходное распределение команд по рабочим местам. Требуется пересадить их так, чтобы минимальное расстояние между командами из одной школы было как можно больше. При этом среди возможных новых размещений команд следует выбрать такое, чтобы сумма расстояний между старыми и новыми местами рабочими команд была минимально возможной.

Формат входных данных

В первой строке входного файла задано число n — количество команд ($1 \leq n \leq 100$). Во второй строке задано исходное распределение команд по рабочим местам — последовательность a_1, a_2, \dots, a_{2n} . Для каждой команды указан номер школы, которую она представляет. Гарантируется, что последовательность состоит из чисел от одного до n и каждое число встречается ровно два раза.

Формат выходных данных

В единственную строку выходного файла выведите, каким образом следует пересадить команды, чтобы минимальное расстояние между командами из одной школы было как можно больше. При этом среди возможных новых размещений команд следует выбрать такое, чтобы сумма расстояний между старыми и новыми рабочими местами команд была минимально возможной. Если оптимальных ответов несколько, можно вывести любой из них.

Пример

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

Задача D. Расшифровка

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

Недавно на уроке во время контрольной Мария Ивановна перехватила записку от Саши к Оле. Мария Ивановна очень хочет знать, что в записке, но, к сожалению, записка зашифрована. Мария Ивановна знает, что её ученики для шифровки заменяют каждую букву исходного сообщения на какую-то другую. Замена происходит таким образом, что одинаковые буквы всегда заменяются одной и той же буквой, а разные — разными.

Мария Ивановна подозревает, что записка — это ответы к контрольному тесту (ведь её длина случайно оказалась равной длине строки с правильными ответами). Однако она знает, что ответы Саши не обязательно полностью правильны. На каждый вопрос возможен один из K вариантов ответа. Естественно, Мария Ивановна знает правильные ответы.

Мария Ивановна решила расшифровать записку таким способом, чтобы максимизировать количество правильных ответов Саши. Однако, она очень занята, поэтому попросила Вас помочь ей в этом пустяковом деле.

Формат входных данных

В первой строке задана длина каждой из строк N ($1 \leq N \leq 2\,000\,000$) и K — количество возможных ответов на каждый вопрос ($1 \leq K \leq 52$). Ответы нумеруются в порядке abcde...xyzABCDE...XYZ. То есть, при $K = 6$ возможные ответы выглядят как abcdef, а при $K = 30$ — abcde...xyzABCD.

Во второй строке задана зашифрованная записка — строка, состоящая из строчных и заглавных латинских букв.

В третьей строке заданы правильные ответы — строка той же длины, что и первая, состоящая из строчных и заглавных латинских букв.

Формат выходных данных

В первой строке выведите единственное число — максимально возможное количество правильных ответов у Саши.

Во второй строке выведите расшифровку — строчку длины K , где по порядку для каждой буквы из шифра учеников указано, какому ответу она соответствует.

Если несколько расшифровок дают правильный ответ, выведите любую.

Примеры

стандартный ввод	стандартный вывод
10 2 aaabbbbaaab bbbbabbbbb	7 ba
10 2 aaaaaaabbb bbbbaaabbb	6 ab
9 4 dacbdacbd acbdacbda	9 cdba

Задача E. План эвакуации

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	2 секунды
Ограничение по памяти:	64 мегабайта

В городе есть муниципальные здания и бомбоубежища, которые были специально построены для эвакуации служащих в случае ядерной войны. Каждое бомбоубежище имеет ограниченную вместительность по количеству людей, которые могут в нем находиться. В идеале все работники из одного муниципального здания должны были бы бежать к ближайшему бомбоубежищу. Однако, в таком случае, некоторые бомбоубежища могли бы переполниться, в то время как остальные остались бы наполовину пустыми.

Чтобы разрешить эту проблему Городской Совет разработал специальный план эвакуации. Вместо того, чтобы каждому служащему индивидуально приписать, в какое бомбоубежище он должен бежать, для каждого муниципального здания определили, сколько служащих из него в какое бомбоубежище должны бежать. Задача индивидуального распределения была переложена на внутреннее управление муниципальных зданий.

План эвакуации учитывает количество служащих в каждом здании — каждый служащий должен быть учтен в плане и в каждое бомбоубежище может быть направлено количество служащих, не превосходящее вместимости бомбоубежища.

Городской Совет заявляет, что их план эвакуации оптимален в том смысле, что суммарное время эвакуации всех служащих города минимально.

Мэр города, находящийся в постоянной конфронтации с Городским Советом, не слишком то верит этому заявлению. Поэтому он нанял Вас в качестве независимого эксперта для проверки плана эвакуации. Ваша задача состоит в том, чтобы либо убедиться в оптимальности плана Городского Совета, либо доказать обратное, представив в качестве доказательства другой план эвакуации с меньшим суммарным временем для эвакуации всех служащих.

Карта города может быть представлена в виде квадратной сетки. Расположение муниципальных зданий и бомбоубежищ задается парой целых чисел, а время эвакуации из муниципального здания с координатами (X_i, Y_i) в бомбоубежище с координатами (P_j, Q_j) составляет $D_{ij} = |X_i - P_j| + |Y_i - Q_j| + 1$ минут.

Формат входных данных

Входной файл содержит описание карты города и плана эвакуации, предложенного Городским Советом. Первая строка входного файла содержит два целых числа N ($1 \leq N \leq 100$) и M ($1 \leq M \leq 100$), разделенных пробелом. N — число муниципальных зданий в городе (все они занумерованы числами от 1 до N), M — число бомбоубежищ (все они занумерованы числами от 1 до M).

Последующие N строк содержат описания муниципальных зданий. Каждая строка содержит целые числа X_i , Y_i и B_i , разделенные пробелами, где X_i , Y_i ($-1000 \leq X_i, Y_i \leq 1000$) — координаты здания, а B_i ($1 \leq B_i \leq 1000$) — число служащих в здании.

Описание бомбоубежищ содержится в последующих M строках. Каждая строка содержит целые числа P_j , Q_j и C_j , разделенные пробелами, где P_j , Q_j ($-1000 \leq P_j, Q_j \leq 1000$) — координаты бомбоубежища, а C_j ($1 \leq C_j \leq 1000$) — вместимость бомбоубежища.

В последующих N строках содержится описание плана эвакуации. Каждая строка представляет собой описание плана эвакуации для отдельного здания. План эвакуации из i -го здания состоит из M целых чисел E_{ij} , разделенных пробелами. E_{ij} ($0 \leq E_{ij} \leq 10\,000$) — количество служащих, которые должны эвакуироваться из i -го здания в j -е бомбоубежище.

Гарантируется, что план, заданный во входном файле, корректен.

Формат выходных данных

Если план эвакуации Городского Совета оптимален, то выведите одно слово OPTIMAL. В противном случае выведите на первой строке слово SUBOPTIMAL, а в последующих N строках выведите

Ваш план эвакуации (более оптимальный) в том же формате, что и во входном файле. Ваш план не обязан быть оптимальным, но должен быть лучше плана Городского Совета.

Примеры

стандартный ввод	стандартный вывод
3 4 -3 3 5 -2 -2 6 2 2 5 -1 1 3 1 1 4 -2 -2 7 0 -1 3 3 1 1 0 0 0 6 0 0 3 0 2	SUBOPTIMAL 3 0 1 1 0 0 6 0 0 4 0 1
3 4 -3 3 5 -2 -2 6 2 2 5 -1 1 3 1 1 4 -2 -2 7 0 -1 3 3 0 1 1 0 0 6 0 0 4 0 1	OPTIMAL

Задача F. Автоматное программирование

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	5 секунд
Ограничение по памяти:	256 мегабайт

В один замечательный день в компанию «X» завезли k автоматов. И не простых автоматов, а автоматов-программистов! Это был последний неудачный шаг перед переходом на андроидов-программистов, но это уже совсем другая история.

В компании сейчас n задач, для каждой из которых известно время начала ее выполнения s_i , длительность ее выполнения t_i и прибыль компании от ее завершения c_i . Любой автомат может выполнять любую задачу, ровно одну в один момент времени. Если автомат начал выполнять задачу, то он занят все моменты времени с s_i по $s_i + t_i - 1$ включительно и не может переключиться на другую задачу.

Вам требуется выбрать набор задач, которые можно выполнить с помощью этих k автоматов и который принесет максимальную суммарную прибыль.

Формат входных данных

В первой строке записаны два целых числа n и k ($1 \leq n \leq 1000$, $1 \leq k \leq 50$) — количество задач и количество автоматов, соответственно.

В следующих n строках через пробелы записаны тройки целых чисел s_i, t_i, c_i ($1 \leq s_i, t_i \leq 10^9$, $1 \leq c_i \leq 10^6$), s_i — время начала выполнения i -го задания, t_i — длительность i -го задания, а c_i — прибыль от его выполнения.

Формат выходных данных

Выведите n целых чисел x_1, x_2, \dots, x_n . Число x_i должно быть равно 1, если задачу i следует выполнить, и 0 в противном случае.

Если оптимальных решений несколько, то выведите любое из них.

Примеры

стандартный ввод	стандартный вывод
3 1 2 7 5 1 3 3 4 1 3	0 1 1
5 2 1 5 4 1 4 5 1 3 2 4 1 2 5 6 1	1 1 0 0 1

Задача G. Высокий маразм

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1.5 секунд
Ограничение по памяти: 256 мегабайт

Город Нью-Васюки представляет собой таблицу N на N кварталов. Таким образом, в нем существует $(N + 1)(N + 1)$ перекрестков и $2N(N + 1)$ двусторонних дорог. Каждый перекресток имеет высоту. Известно, что левый верхний перекресток имеет высоту 0, а правый нижний – высоту 1. Для каждой дороги известно, сколько человек идет в каждом направлении по этой дороге. При этом, если дорога ведет от перекрестка i к перекрестку j , и разность высот $h = h_j - h_i$, то неудобство перемещения для каждого человека равно $\max(h, 0)$.

Для всех клеток, кроме двух угловых вы вправе выбирать любую высоту. Найдите распределение, при котором суммарное неудобство будет минимальным.

Формат входных данных

В первой строке входного файла содержится число N – размер таблицы, $1 \leq N \leq 500$. Далее следует $4N(N+1)$ чисел – искомое число людей для каждой дороги в естественном порядке. Сначала следует $N(N + 1)$ чисел для каждой дороги с запада на восток, затем столько же чисел с севера на юг, потом столько же чисел с востока на запад, а затем столько же чисел с юга на север. Для каждого направления дороги перечислены с севера на юг, а затем – с запада на восток. Обратите внимание на пример (там все числа специально различны).

Все количества являются целыми неотрицательными числами не превосходящими миллиона.

Формат выходных данных

Выведите одно число – итоговое минимальное суммарное неудобство.

Пример

стандартный ввод	стандартный вывод
1	3
1	
2	
3	
4	
5	
6	
7	
8	

Замечание

Высоты могут быть произвольными вещественными числами! Картинка к примеру из условия:

