

Задача А. Невероятный Петрович

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

Индивидуальный предприниматель Иван Петрович с самого детства увлекался химией. Именно благодаря своей преданной любви к этому непростому предмету, а также усердному труду, он еще в аспирантуре МГУ создал первый прототип «чудодейственной» сыворотки, которая позволяет увеличить силу любого человека примерно в 30 раз. Причем, как и все гениальное, рецепт сыворотки оказался относительно простым – в его составе всего-то 3 секретных ингредиента.

Названия этих ингредиентов Иван Петрович держит в строгом секрете и для того, чтобы сбить конкурентов с толку, заказывает себе в лабораторию 6 различных компонентов, среди которых есть 3 нужных ему секретных ингредиента и еще 3 ненужных компонента, которые Иван Петрович заказывает для отвода глаз.

Зная количество каждого из заказанных компонентов, а также зная необходимое количество секретных ингредиентов для изготовления одного миллилитра сыворотки, определите возможное минимальное и максимальное количество сыворотки (в миллилитрах), которую может получить Иван Петрович.

Формат входных данных

В первой строке содержится три числа e_1, e_2, e_3 - количество, соответственно, первого, второго и третьего ингредиентов для изготовления одного миллилитра сыворотки ($1 \leq e_i \leq 10^{10}$). Во второй строке содержится шесть чисел $k_1, k_2, k_3, k_4, k_5, k_6$ - количество компонентов, которые заказал Иван Петрович ($1 \leq k_i \leq 10^{10}$).

Формат выходных данных

Выведите два целых неотрицательных числа через пробел: минимальное и максимальное возможное количество сыворотки.

Пример

стандартный ввод	стандартный вывод
2 8 4 10 48 12 50 30 25	1 6

Замечание

Например, для того, чтобы получить 1 мл сыворотки нужно, чтобы секретным ингредиентом 1 оказался первый компонент, секретным ингредиентом 2 - третий компонент, секретным ингредиентом 3 - 5 компонент (это не единственный возможный вариант).

Чтобы получить 6 мл сыворотки, нужно, чтобы секретным ингредиентом 1 оказался пятый компонент, секретным ингредиентом 2 оказался второй компонент, секретным ингредиентом 3 оказался 6 компонент (это не единственный возможный вариант).

Задача В. Железный друг

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

Бизнес Ивана Петровича стремительно рос, и поэтому он решил найти себе хорошего менеджера по продажам. У него есть три возможных кандидата: первый - Петр Александрович, второй - Владимир Сергеевич и третий - Тимур Дмитриевич. Он хочет выбрать только одного из них и для этого решил устроить им месячный испытательный срок.

Кандидатов Иван Петрович набирал по объявлениям, а заработную плату устанавливал по результатам собеседования. Первым к нему пришел Петр Александрович и согласился работать за w_1 рублей в месяц. Вторым пришел Владимир Сергеевич и согласился работать за w_2 рублей в месяц. Третьим пришел Тимур Дмитриевич – он согласился работать за w_3 рублей в месяц.

Работа каждого менеджера осложняется также тем, что рынок сыворотки еще не оформился, а цены не устоялись. Поэтому каждый менеджер сам выбрал цену, по которой продавал один миллилитр сыворотки: Петр Александрович стал продавать сыворотку по цене c_1 , Владимир Сергеевич по цене c_2 , а Тимур Дмитриевич по цене c_3 .

Прошел месяц, и по его результатам Иван Петрович хочет принять решение, кого ему брать на работу: Петра Александровича, Владимира Сергеевича или Тимура Дмитриевича. Для этого он выписал, сколько миллилитров сыворотки продал каждый из них – r_1 , r_2 , r_3 соответственно. Но от этого количества чисел он уже очень устал и хочет, чтобы принять решение ему помог «железный друг» (так Иван Петрович называет свой домашний компьютер). Железный друг никогда не ошибается и точно выберет того (или тех), кто заработал Ивану Петровичу больше всего денег. Кто же это будет?

Формат входных данных

Входные данные состоят из трех строк.

В первой строке три натуральных числа w_1 , w_2 , w_3 - зарплата за месяц Петра Александровича, Владимира Сергеевича и Тимура Дмитриевича соответственно.

Во второй строке три натуральных числа c_1 , c_2 , c_3 - стоимость сыворотки, по которой осуществлял продажи Петр Александрович, Владимир Сергеевич и Тимур Дмитриевич соответственно.

В третьей строке три натуральных числа r_1 , r_2 , r_3 - количество миллилитров сыворотки, которое продали Петр Александрович, Владимир Сергеевич и Тимур Дмитриевич соответственно.

Все числа не превосходят 10^6

Формат выходных данных

Выведите номера тех менеджеров, которые принесли наибольший доход: 1 для Петра Александровича, 2 для Владимира Сергеевича и 3 для Тимура Дмитриевича. Если несколько менеджеров принесли одинаковый и наибольший доход, то выведите их номера по возрастанию, разделяя пробелом.

Если ни один из менеджеров не принес Ивану Петровичу доход, выведите 0.

При расчете дохода в данной задаче себестоимость производства сыворотки не учитывается.

Примеры

стандартный ввод	стандартный вывод
30000 50000 100000 40 70 1000 1000 1000 90	2
20000 40000 30000 100 200 300 150 100 50	0
5000 5000 6000 60 60 70 100 100 100	1 2 3

Замечание

В первом примере Петр Александрович отработал за 30000 рублей и при этом продал 1000 миллилитров сыворотки по цене 40 рублей за миллилитр, т.е. доход от месяца его работы составил 10000 рублей. Владимир Сергеевич работал за 50000 рублей в месяц и при этом продал 1000 миллилитров сыворотки по цене 70 рублей за миллилитр, т.е. доход от месяца его работы составил 20000. Тимур Дмитриевич получил зарплату в размере 100000 рублей, а продал сыворотки только на 90 тысяч - он Ивану Петровичу дохода не принес.

Во втором примере ни один из троих менеджеров не принес Ивану Петровичу доход.

В третьем примере все трое принесли одинаковый доход.

Задача С. Круг

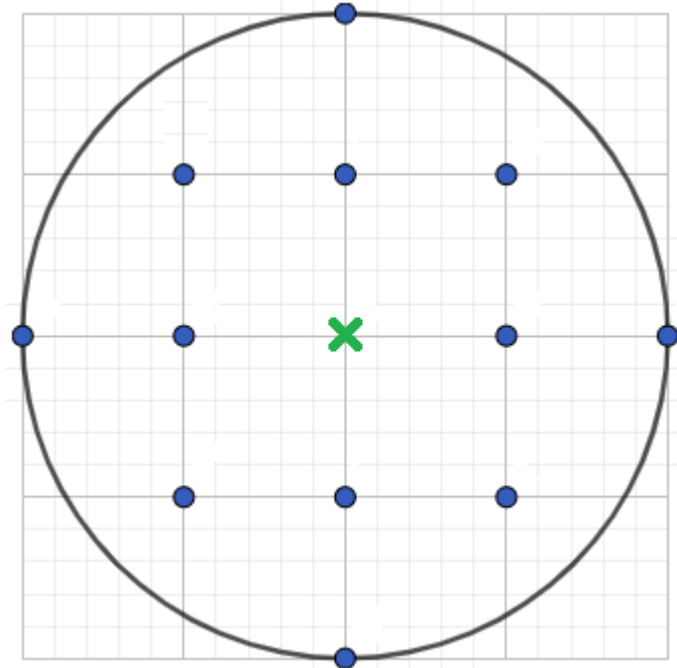
Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1 секунда
Ограничение по памяти:	256 мегабайт

«Любите ли Вы геометрию так, как люблю ее я?»

Эти слова Иван Петрович вспомнил, когда приехал в хозяйственный магазин «Рагнарёк» для выбора системы автоматического полива. Слова эти принадлежали его школьному учителю математики Аристарху Поликарповичу, а вспомнил он их не случайно.

В своем загородном доме Иван Петрович разбил клумбу. К делу подошел как всегда основательно: отмерил ровную квадратную клумбу размером N на N сантиметров. Расстояние между каждым рядом цветов ровно 20 см, при этом ряды начинаются из центра. Точно к центру клумбы подвел шланг с водой, а на саму клумбу на пересечении каждого ряда хочет посадить по одному цветку.

Для того, чтобы цветы не засохли долгим и жарким летом, Иван Петрович собирается установить в центр клумбы систему полива, которая будет распыскивать воду в заданном радиусе. Известно, что если цветок окажется вне радиуса полива, то он погибнет. С другой стороны, чем больше радиус, тем дороже стоит система полива, а Иван Петрович не хочет, чтобы вода лилась мимо клумбы, потому что считает это крайне расточительным.



Уроки Аристарха Поликарповича остались в прошлом вместе со всеми знаниями Ивана Петровича о геометрии. Поэтому Вам необходимо написать программу, которая по заданным размерам клумбы определит необходимый минимальный радиус полива такой, что он не превосходит $N/2$ и при этом будет полито максимальное количество цветов, если каждый цветок, который находится на расстоянии $\leq R$, где R - радиус системы полива - будет полит.

Формат входных данных

На вход программы подается одно число - N размер клумбы в сантиметрах ($40 \leq N \leq 10^6$).

Формат выходных данных

Выведите в двух строках два целых числа: максимальное количество цветов, которое удастся полить, не проливая воду вне границ клумбы, а также минимальный радиус полива (в сантиметрах), который для этого необходим.

Примеры

стандартный ввод	стандартный вывод
80	12 40
90	20 45
140	36 64

Замечание

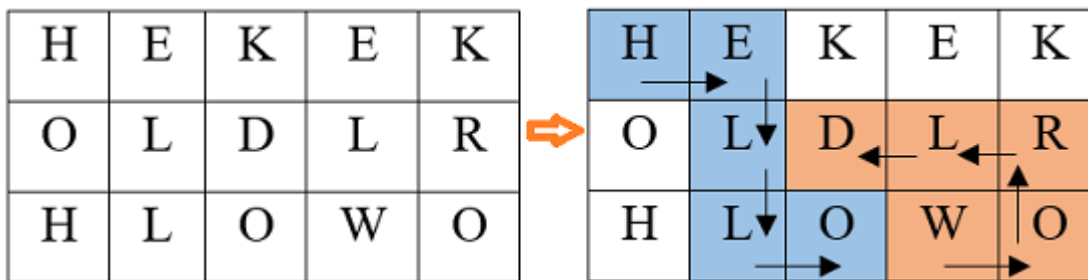
Первый тест изображен на картинке: система полива располагается по центру клумбы (зеленый крестик), размер клумбы 80x80 см. В центре клумбы нет цветка, на пересечение остальных рядов есть цветки (синие круги). Таким образом, получится поливать 12 цветков с радиусом полива 40.

Задача D. Венгерский кроссворд

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

Венгерский кроссворд, также известный как филворд - это разновидность привычного нам с Вами кроссворда. Только в отличие от обычного кроссворда на поле в филворде уже написаны слова, а от игрока требуется их найти и вычеркнуть. Разгадкой филворда является последовательность букв, которая осталась после того, как были вычеркнуты все слова из заданного списка.

Сложность в том, что в отличие от обычного кроссворда буквы одного слова в филворде могут идти в различных направлениях. При этом любые две соседние буквы одного слова должны находиться на соседних клетках в таблице филворда (соседними называют клетки с общей стороной). Кроме того, в отличие от обычного кроссворда, одна буква филворда принадлежит только одному слову из списка.



Ивану Петровичу попался очень большой филворд и теперь он пытается его разгадать. Но вы-то, конечно, сможете ему помочь это сделать быстро?

Формат входных данных

В первой строке два натуральных числа: N и M - количество строк и столбцов филворда ($1 \leq N, M \leq 10$) соответственно.

Далее, начиная со второй строки, следует филворд, состоящий из N строк по M заглавных букв английского алфавита.

В следующей строке идет число K ($1 \leq K \leq 100$) - количество слов, которые необходимо вычеркнуть из филворда, а затем в K строках идут сами слова, составленные из заглавных букв английского алфавита.

Формат выходных данных

Выведите оставшиеся после вычеркивания из филворда всех слов из списка буквы в алфавитном порядке. Гарантируется, что все слова из списка можно вычеркнуть в филворде.

Примеры

стандартный ввод	стандартный вывод
3 5 HEKEK ELDLR PLOWO 2 HELLO WORLD	ЕЕККР
4 4 DLOL FDON SILT AKEI 4 I DONT LIKE DFS	ALLO

Задача Е. Максимальный deck

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

Робот Федор и Робот Джон попали на МКС в рамках миссии по исследованию возможностей роботов на орбите Земли. Беда в том, что роботы эти настолько хороши, что они умеют скучать, когда у них нет работы.

Как-то одним из вечеров, после того, как все тесты были уже сделаны, Робот Федор и Робот Джон так заскучали, что решили придумать новую карточную игру. У каждого из них была с собой колода карт, состоящая из 52 карт: 4 масти (H, S, C, D) по 13 карт (2, 3, 4, 5, 6, 7, 8, 9, 10, J, Q, K, A). Они решили, что каждая карта будет представлять собой какой-то номинал - одно уникальное число, получаемое по следующим правилам: номер масти умножается на 13 и прибавляется номер карты. Например, «H 2» имеет номинал 0, «S 3» номинал 14, «C 10» номинал 34, а «D K» номинал 50.

Робот Федор и Робот Джон сложили две своих колоды вместе и перемешали. Робот Джон называет случайное число N от 1 до 100, после чего выкладывает N карт на стол в одну линию лицевой стороной вверх (т.е. все номиналы карт видны играющим). После этого Федор делает первый ход, который заключается в том, что он берет себе одну из двух возможных карт: либо самую левую, либо самую правую, и кладет ее перед собой - брать карты из середины запрещено. Следующий ход делает Робот Джон - он также берет одну из двух возможных карт и кладет ее перед собой. Игра продолжается до тех пор, пока все карты не будут лежать в двух колодах (англ. deck) - одна колода перед Джоном, одна перед Федором.

Для того, чтобы узнать, кто победил роботы подсчитывают сумму номиналов собственных карт - у кого больше, тот и выиграл, если у обоих одинаковые суммы - это ничья. Напишите программу, которая по начальной позиции определяет, кто выигрывает: Робот Федор, Робот Джон или же между ними будет ничья. Помните, что роботы никогда не ошибаются, т.е. всегда берут карты наилучшим образом для себя.

Формат входных данных

В первой строке содержится число N ($1 \leq N \leq 100$) - количество карт. Дальше в N строках содержится информация о картах, лежащих перед роботами, в порядке от самой левой (первая из карт) до самой правой (последняя из карт). Информация о карте представляет собой записанные через пробел код масти (H, S, C, D) и значение (2, 3, 4, 5, 6, 7, 8, 9, 10, J, Q, K, A).

Формат выходных данных

Выведите одно слово: FYODOR, если выигрывает Робот Федор, DJON, если выигрывает Робот Джон, или DRAW, если результатом игры является ничья.

Примеры

стандартный ввод	стандартный вывод
4 H J C J C J D 6	FYODOR
3 H 3 S Q H 7	DJON

Замечание

В первом тесте перед игроками лежат слева направо 4 карты: червовый валет, трефовый валет,

трефовый валет и бубновая шестерка. Федор на первом ходу берет себе бубновую шестерку и затем, как бы не пошел Джон, он берет себе один из валетов треф. Таким образом, у Федора $43 + 35 = 78$, а у Джона $9 + 35 = 44$. Федор выиграл.

Во втором тесте перед игроками лежат слева направо 3 карты: червовая тройка, пиковая дама, червовая семерка. Федор может взять на первом ходу одну из двух червовых карт (тройку или семерку), но независимо от этого на своем первом ходу Джон возьмет пиковую даму. У Федора будет $1 + 5 = 6$, у Джона будет 23. Джон выиграл.

Задача F. Все на отдых!

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

Индивидуальный предприниматель Иван Петрович, честно заработавший свои миллионы, решил съездить наконец-то на отдых. Отдохнуть он собирается основательно поэтому выбрал себе для посещения N различных городов по всему миру. Заранее он решил, что некоторые из городов он не хочет посещать, пока не посетит какой-то другой город. Другими словами он составил для себя список зависимостей посещения, состоящий из пар $(town_i, town_j)$ такой, что $town_j$ может быть посещен только позже $town_i$ (необязательно сразу после, но позже во время отдыха).

Среди всего списка городов Иван Петрович выбрал для себя **обязательный список** - несколько городов, которые он хочет посетить обязательно. Остальные он может как посетить, так и не посещать. В каждом городе Иван Петрович выбрал гостиницу, в которой будет останавливаться. При этом так как городов много, то в каждой гостинице он будет находиться ровно одни сутки. Стоимость суток проживания в каждой гостинице также известна.

Вам нужно по имеющейся информации определить любой допустимый порядок посещения городов, такой чтобы Иван Петрович побывал в каждом городе из обязательного списка, соблюдая определенный заранее список зависимостей посещения и при этом, чтобы стоимость всей поездки была минимальной.

Гарантируется, что отсутствуют транзитивные и циклические зависимости, а значит необходимый порядок может быть определен всегда. Также гарантируется, что в списке зависимостей отсутствуют одинаковые пары (в которых одновременно совпадают первый и второй город).

Формат входных данных

В первой строке следует натуральное число N - количество городов ($1 \leq N \leq 230$).

Затем в N строках записана информация о городах посещения, состоящая из трех элементов, записанных через пробел: $town_i, price_i, required_i$. $town_i$ - название города состоит из латинских букв и знака «-»; $price_i$ - стоимость проживания в городе - положительное вещественное число с двумя знаками после точки не превосходящее 10^7 ; $required_i$ - это один двоичный бит равный 1, если город является обязательным для посещения и 0 в противном случае. Гарантируется, что хотя бы один город отмечен как обязательный к посещению.

Затем следует число M ($0 \leq M \leq 500$) - целое неотрицательное число с количеством зависимостей. После чего записано еще M строк, в которых содержится информация о зависимостях посещения городов в виде пар: $town_i town_j$.

Формат выходных данных

В первой строке выведите T - количество городов, которое посетит Иван Петрович.

Затем T строках выведите названия этих городов в том порядке, в котором их необходимо посещать. Если возможных списков несколько, то выведите в любом допустимом.

В последней строке выведите сумму стоимости проживания в выбранных городах с двумя знаками после точки.

Примеры

стандартный ввод	стандартный вывод
7 Budapest 70.90 0 Riga 75.41 1 Berlin 95.12 0 Warsaw 77.77 1 Rome 100.25 1 Oslo 90.00 0 Paris 112.12 0	5 Budapest Riga Berlin Warsaw Rome 419.45
6 Budapest Riga Riga Warsaw Berlin Warsaw Berlin Oslo Berlin Paris Oslo Paris	
7 Budapest 70.90 0 Riga 75.41 1 Berlin 95.12 0 Warsaw 77.77 1 Rome 100.25 1 Oslo 90.00 0 Paris 112.12 0	4 Berlin Riga Warsaw Rome 348.55
2 Berlin Riga Berlin Rome	

Замечание

В первом примере список обязательных городов: Riga, Warsaw, Rome. Чтобы посетить Riga, необходимо сначала побывать в Budapest. Чтобы посетить Warsaw нужно сначала отправиться в Berlin. Rome в первом примере не имеет зависимости для посещения, поэтому его можно посетить в любой момент. Сумма стоимости проживания: $70.90 + 75.41 + 95.12 + 77.77 + 100.25 = 419.45$

Во втором примере сначала необходимо посетить Berlin и только потом можно посетить Riga и Rome. Город Warsaw также является обязательным к посещению и может быть посещен в любой момент. Сумма стоимости проживания: $95.12 + 75.41 + 77.77 + 100.25 = 348.55$