

Задача A. RPG battle

Имя входного файла:	standard input
Имя выходного файла:	standard output
Ограничение по времени:	2 seconds
Ограничение по памяти:	64 Mebibytes

Не так давно была анонсирована очередная компьютерная RPG (Role Playing Game) под рабочим названием TCM. Вскоре вышла и бета-версия... тестеры нашли в ней определённые ошибки, которые были исправлены. На игровых форумах и в социальных сетях обсуждалось, как же расшифровывается название TCM. Дошло до того, что знатный свиновод S. написал в своём блоге статью «:O) TCM». И вот подошёл день релиза. Вы зашли на один из открывшихся серверов этой игры, выяснили, что название TCM расшифровывается как «Totally Controlled Magic», сгенерировали персонажа... Но перед тем, как начать основную игру, каждый игрок обязан пройти несколько заданий в Гильдии — для того, чтобы понять базовые концепции игры, в частности — принципы выбора оружия. Правила битвы между игроком и монстром традиционно заимствуются из классических настольных RPG.

В настольных RPG сражение происходит следующим образом. У каждого монстра есть уровень защиты (AC, Armor Class) — целое положительное число. Игрок сражается с помощью оружия, которое имеет два параметра: «пробивная сила» и «повреждение». Пробивная сила — целое неотрицательное число, повреждение задаётся в виде $nds+b$, где n и s — целые положительные числа, b — целое неотрицательное число.

При атаке игрок сначала бросает 20-гранную игральную кость, на гранях которой расположены числа от 1 до 20 (вероятность выпадения каждого из чисел одинакова).

Если на верхней грани выпадает число от 2 до 19, то к выпавшему числу прибавляется пробивная сила. В случае, если получившаяся сумма меньше уровня защиты монстра, считается, что игрок не попал. В противном случае для вычисления повреждения, нанесённого монстру, игрок n раз бросает s -гранную игральную кость, на гранях которой расположены числа от 1 до s (вероятность выпадения каждого из чисел одинакова), суммирует выпавшие значения и прибавляет значение b .

Если на верхней грани выпадает 1, то считается, что игрок не попал вне зависимости от уровня защиты монстра.

Если на верхней грани выпадает 20, то считается, что вне зависимости от уровня защиты монстра игрок попал в уязвимую точку монстра и нанёс максимально возможное повреждение (то есть при вычислении повреждения принимается, что на всех n игровых костях он выбросил s).

Вам заданы параметры (пробивная сила и повреждение) двух типов оружия, а также AC монстра, с которым предстоит сражаться. Вычислите, какое оружие в среднем нанесёт наибольшее повреждение этому монстру.

Формат входного файла

В первой строке входного файла записано целое положительное число $AC \leq 1000$ — уровень защиты атакуемого монстра. В следующих двух строках описаны оба типа оружия: в каждой из строк сначала идёт название (не более 20 строчных или прописных латинских букв), затем — «пробивная сила» h ($0 \leq h \leq 100$), затем повреждение в виде $nds+b$ ($1 \leq n \leq 100$, $2 \leq s \leq 100$, $0 \leq b \leq 1000$).

Формат выходного файла

В выходной файл выведите название более эффективного оружия. Если средние значения повреждения, наносимого этими видами оружия, отличаются менее, чем на 10^{-7} , выведите строку `Equal damage`.

Пример

standard input	standard output
14 ShortSword 2 1d7+4 Lance 1 1d9+6	Lance
1000 Katana 80 2d90+20 Firesword 1 5d10+150	Equal damage

Задача В. Maze

Имя входного файла: `standard input`
Имя выходного файла: `standard output`
Ограничение по времени: 2 seconds
Ограничение по памяти: 64 Mebibytes

После обучения в Гильдии вы должны пройти лабиринт, в каждой из комнат которого один из параметров, определяющих способности вашего персонажа, увеличивается на 1. Лабиринт устроен следующим образом: одна из комнат является входом в лабиринт, несколько комнат являются выходом из лабиринта. В каждой из комнат, не являющихся выходом, есть один или два портала, ведущие в другие комнаты. При этом порталы расположены так, что невозможно дважды побывать в одной и той же комнате.

По имеющейся у вас информации о расположении порталов и выходов вычислите, в каком наибольшем количестве комнат можно побывать за одно путешествие по лабиринту от входа до одного из выходов.

Формат входного файла

В первой строке входного файла задано целое число N ($2 \leq N < 1000$) — количество комнат в лабиринте. В последующих N строках идёт описание комнат: в i -й строке задано не более двух целых чисел от 1 до N — номера комнат, в которые ведут порталы из i -й комнаты (комнаты занумерованы с единицы, в первой комнате находится вход), если i -я комната не является одним из выходов. Если i -я комната является выходом, то i -я строка содержит слово EXIT. Порталы расположены таким образом, чтобы ни в одной комнате нельзя было побывать дважды.

Формат выходного файла

В выходной файл выведите единственное целое число — максимальное количество комнат, в которых можно побывать за одно путешествие по лабиринту от входа до одного из выходов.

Пример

standard input	standard output
3 3 2 EXIT EXIT	2
5 5 4 EXIT 2 3 4	5

Задача C. Balance

Имя входного файла: `standard input`
Имя выходного файла: `standard output`
Ограничение по времени: 2 seconds
Ограничение по памяти: 64 Mebibytes

Перед тем, как отправиться на охоту за уровнями и сокровищами, в любой RPG-игре необходимо приобрести снаряжение, а также еду и питьё. При этом, по-видимому, для придания реалистичности, а может быть для того, чтобы сделать отыгрыш более интересным, разработчики игры сделали возможным мелкий обман со стороны продавцов; впрочем, внимательный игрок сможет обернуть его себе на пользу.

В продуктовой лавке, расположенной неподалёку от выхода из лабиринта, вы обратили внимание на конструкцию весов. На весах, представлявших из себя однородный рычаг, закреплённый строго посередине, были на разных расстояниях от точки подвеса размещены одинаковые грузы. При этом конструкция весов допускает прикрепление нескольких грузов к одной точке. К концам рычага подвешены две одинаковые чашки весов. Явный механизм для обвеса!

Вы измерили расстояния от точки подвеса до каждого из грузов и решили выяснить, в каком положении будут находиться весы, когда на их чашках ничего не размещено.

Формат входного файла

В первой строке входного файла записано целое неотрицательное число $N \leq 100$ — количество грузов. В следующей строке заданы N целых чисел, каждое из которых по модулю не превосходит 1000. i -е число задаёт расстояние от i -го груза до точки подвеса: если число положительное, груз находится справа от точки подвеса, если отрицательное — слева, если 0 — непосредственно в точке подвеса.

Формат выходного файла

В выходной файл выведите строку `Left Scale`, если при пустых чашках весов перевешивает левая чашка, `Right Scale`, если перевешивает правая, и `Fair`, если весы остаются в равновесии.

Пример

<code>standard input</code>	<code>standard output</code>
3 -3 0 3	Fair
1 6	Right Scale
4 5 -3 0 -3	Left Scale

Задача D. Teleport

Имя входного файла: `standard input`
Имя выходного файла: `standard output`
Ограничение по времени: 2 seconds
Ограничение по памяти: 64 Mebibytes

Вы вышли из города и зашли в телепорт, который случайным образом переместил вас на один из двух оставшихся континентов игрового мира: оказалось, что первый же встреченный Вами местный житель вместо приветствия произнёс слово на непонятном языке.

Дело в том, что в данной игре для того, чтобы понимать то, что говорят персонажи, живущие на других континентах, необходим специальный навык, которого у вас пока что нет. Поэтому вместо связного текста вам выдаётся некая непонятная фраза, причём для каждого континента действуют свои правила, по которым строятся подобные фразы. Если на северном континенте выдаётся случайный набор заглавных латинских букв, то на южном правила построения набора таковы: при построении фразы длины L берётся случайная перестановка первых K (где K — случайное число от 1 до 26) заглавных латинских букв, при необходимости повторяется несколько раз (например, DCBADCBCBADCBA...), после чего в качестве фразы выдаются первые L символов полученной строки.

Например, фраза CBFDAEC может встретиться на южном континенте (начальная подстрока длины 7 повторяющейся перестановки CBFDAE первых 6 букв латинского алфавита), а фразы BCBC или BACABBACAB не могут.

Ваша задача — по заданной Вам фразе вычислить, на каком континенте Вы находитесь, или определить, что информации недостаточно.

Формат входного файла

Во входном файле задана непустая строка, состоящая из не более, чем 1000 заглавных латинских букв — услышанная вами фраза.

Формат выходного файла

В выходной файл выведите `South`, если по данной фразе можно однозначно определить, что вы находитесь на южном континенте, `North`, если по данной фразе можно однозначно определить, что вы находитесь на северном континенте, `Undefined`, если по данной фразе нельзя определить, на каком из двух континентов вы находитесь.

Пример

standard input	standard output
ACB	Undefined
CC	North
PRIVET	Undefined
PREVED	North

Задача E. Dragon City

Имя входного файла: `standard input`
Имя выходного файла: `standard output`
Ограничение по времени: 2 seconds
Ограничение по памяти: 64 Mebibytes

После очередной битвы вы заметили тяжёлые ворота и открыли их. Сделав шаг внутрь, вы обнаружили, что стоите на большой площади Города Драконов, по всей видимости, покинутого его жителями. В городе имеется некоторое количество площадей, соединённых дорогами с двусторонним движением. На каждой из площадей, кроме Square of Wooden Gates (той, на которой вы сейчас находитесь), драконы поставили золотую статуэтку. Ваша задача — вынести все статуэтки из города. Для того, чтобы вынести статуэтку из города, её необходимо принести на Square of Wooden Gates. Одновременно Вы можете нести не более трёх статуэток. При этом вынести статуэтки необходимо как можно быстрее — ваша израненная в боях интуиция подсказывает, что драконы могут и вернуться.

По заданному расположению площадей и соединяющих их дорог и по заданному для каждой дороги времени, за которое вы можете пройти по ней от одной площади до другой, вычислите наименьшее время, за которое все статуэтки могут быть вынесены. Время, которое тратится на то, чтобы взять статуэтку, а также на то, чтобы передать принесённые на Square of Wooden Gates статуэтки вашим помощникам, находящимся за воротами города, считать нулевым.

Формат входного файла

В первой строке входного файла заданы два целых числа N и M — количество площадей, на которых установлены статуэтки, и количество дорог соответственно ($1 \leq N \leq 20$). В каждой из последующих M строк заданы 3 целых числа x_i , y_i и t_i ($0 \leq x_i, y_i \leq N$, $1 \leq t_i \leq 10^5$) — соответственно номера площадей, которые соединяет дорога, и время, за которое можно пройти по этой дороге. При этом Square of Wooden Gates соответствует номер N , остальным площадям — номера от 0 до $N - 1$, и для каждой пары площадей существует не более одной дороги, непосредственно соединяющей эти площади. Также известно, что между каждыми двумя площадями Города Драконов существует путь по сети дорог.

Формат выходного файла

В выходной файл выведите единственное целое число — минимальное время, которое требуется для того, чтобы вынести все золотые статуэтки из Города Драконов.

Пример

standard input	standard output
2 2 2 1 11 1 0 11	44

Задача F. Lake

Имя входного файла: `standard input`
Имя выходного файла: `standard output`
Ограничение по времени: 2 seconds
Ограничение по памяти: 64 Mebibytes

На обратном пути из Города Драконов при переправе через озеро вас заметили разбойники, также переправлявшиеся в некотором направлении. И вы, и разбойники находитесь на прямоугольных плотках, которые движутся равномерно и прямолинейно, не поворачиваясь ни в какой момент движения. Плоты начали своё движение одновременно и за единицу времени пересекают озеро. В случае, если плоты столкнутся (то есть если представляющие их прямоугольники пересекутся хотя бы по одной точке), вы будете атакованы, в случае, если столкновения не произойдёт, разбойники не заметят сокровищ и пойдут своим путём.

Вам известны размеры плотов и координаты положения верхнего левого угла каждого из плотов в моменты времени 0 и 1 соответственно. Чтобы подготовиться к отражению атаки разбойников, вам надо вычислить первый момент, в который произойдёт столкновение (если таковое произойдёт). Оба плота расположены таким образом, что стороны соответствующих прямоугольников всё время параллельны осям координат. Система координат устроена таким образом, что значения x увеличиваются слева направо, а значения y — сверху вниз.

Формат входного файла

В первой строке входного файла заданы 6 целых чисел: $A_1, B_1, Xst_1, Yst_1, Xfin_1, Yfin_1$ — соответственно длина и высота вашего плота, координаты положения его левого верхнего угла в начале и в конце пути. Во второй строке входного файла заданы 6 целых чисел: $A_2, B_2, Xst_2, Yst_2, Xfin_2, Yfin_2$ — соответственно длина и высота плота разбойников, координаты положения его левого верхнего угла в начале и в конце пути ($0 < A_i, B_i \leq 100, 0 \leq Xst_i, Yst_i, Xfin_i, Yfin_i \leq 10^4$). Координаты x и y в выбранной системе координат возрастают слева направо и сверху вниз соответственно.

Формат выходного файла

В выходной файл выведите единственное число — момент времени, в который плоты столкнутся в первый раз, или -1.0 , если столкновения не произойдёт. Ответ выводить с точностью 10^{-7} .

Пример

standard input	standard output
1 1 0 0 10 10 1 1 10 12 0 2	0.55
1 1 0 2 0 2 1 1 0 0 10000 9999	-1.0