

Задача А. Новая надежда

Имя входного файла: `a.in`
Имя выходного файла: `a.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

В далёкой-далёкой галактике силы Сопротивления продолжают войну против императора Палпатина. Повстанцы наконец-то одержали свою первую победу над силами Империи и захватили чертежи новейшей боевой станции — Звезды смерти. Теперь перед повстанцами стоит непростая задача: нужно любой ценой уничтожить станцию, иначе Империя с её помощью укрепит свои позиции в галактике, и дни Сопротивления будут сочтены.

Уничтожение Звезды смерти, как и любая другая амбициозная задача, проходит этап компьютерного моделирования. По текущему плану, для уничтожения станции будет построен мощный ракетный крейсер, который станет флагманом сил Сопротивления и вступит в бой со Звездой смерти, пока остальной флот будет отвлекать внимание противника. Новый корабль будет стрелять ракетами мощностью A единиц и обладать щитами с энергией, достаточной для поглощения B единиц урона. Звезда смерти, в свою очередь, не может использовать в ближнем бою турболазер, способный уничтожать целые планеты, поэтому с неё будет вестись ответный огонь импульсами мощностью C единиц. Щиты Звезды смерти обладают энергией, достаточной для поглощения D единиц урона.

Вам поручено написать программу, моделирующую сражение крейсера мятежников с боевой станцией. Бой происходит поэтапно. На каждом этапе Звезда смерти и флагман одновременно наносят друг по другу удар. Каждый залп уменьшает энергию щита на величину, равную мощности выстрела неприятеля. Как только энергия щитов любого из противников перестаёт быть положительной, он терпит поражение.

Определите, удастся ли кому-либо одолеть своего врага, или же бой завершится уничтожением обеих сторон.

Формат входных данных

В первой и второй строках записаны два целых числа A и B ($1 \leq A, B \leq 1\,000$) — соответственно огневая мощь и энергия щитов флагмана флота Сопротивления.

В третьей и четвертой строках записаны два целых числа C и D ($1 \leq C, D \leq 1\,000$) — соответственно огневая мощь и энергия щитов Звезды смерти.

Формат выходных данных

Если Сопротивление одержит победу, взорвав Звезду смерти на каком-то этапе, и сохранит свой флагман, выведите «WIN».

Если флагман взорвётся, а станция уцелеет, выведите «LOSE».

В случае, если противники уничтожат друг друга на одном и том же шаге боя, выведите «DRAW».

Примеры

a.in	a.out
3 5 1 11	WIN
4 10 5 9	LOSE
1 5 2 3	DRAW

Задача В. ABC

Имя входного файла: `b.in`
Имя выходного файла: `b.out`
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Анализируя результаты командной олимпиады прошлого года, жюри решило помимо лиг А и В организовать ещё и лигу С. На стадии распределения задач по лигам каждая отобранная задача была помечена соответствующим набором букв: А, В, С, АВ, ВС или ABC.

Для подготовки олимпиады требуется составить единый список задач. Жюри хочет, чтобы задачи для одной лиги шли в списке подряд. Помогите жюри упорядочить задачи требуемым образом или определите, что это невозможно.

Формат входных данных

В первой строке ввода находится число N ($3 \leq N \leq 32$) — количество задач, отобранных для олимпиады.

В каждой из следующих N строк содержится описание задачи, начинающееся с одной из пометок А, В, С, АВ, ВС или ABC, записанной заглавными латинскими буквами, за которой через пробел следует название задачи. Название задачи — одно слово, состоящее из строчных или прописных латинских букв. Все названия задач являются различными. Длина каждой строки не превосходит 255 символов.

Формат выходных данных

Выведите описания задач в требуемом порядке в том же формате, что и во входных данных. Все задачи, в пометку которых входит буква А, должны идти подряд, то же касается задач, пометка которых содержит букву В, и задач, пометка которых содержит букву С. Задачи, имеющие одинаковые пометки, могут идти в любом порядке.

В случае, если добиться требуемого невозможно, выведите в единственной строке слово `Impossible`.

Примеры

<code>b.in</code>	<code>b.out</code>
5 C Tetris B DOOM A WOW C LINES AB CS	A WOW AB CS B DOOM C Tetris C LINES
4 A a B b C c ABC Abc	Impossible

Задача С. Соцопрос

Имя входного файла: `c.in`
Имя выходного файла: `c.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Мэр уездного города $\mathfrak{N}$ заботится о жителях и прислушивается к их мнению, поэтому он ежегодно даёт поручение социологической комиссии провести опрос общественного мнения. Участникам опроса предлагается M вопросов, i -й из которых содержит целое положительное количество вариантов ответа a_i . Жители города крайне серьёзно относятся к данному опросу, поэтому заполняют анкету добросовестно: каждый участник опроса выбирает ровно один вариант ответа в каждом вопросе.

При обработке результатов по каждому вопросу для каждого из доступных ответов считается процент людей, отдавших ему предпочтение. К большой радости сотрудников комиссии, обрабатывающих результаты опроса, в этом году все полученные числа выражаются целым количеством процентов.

В целях соблюдения конфиденциальности и анонимности все данные с бумажных анкет были занесены в электронную базу, а листы уничтожены. Но, по закону подлости, в ночь перед презентацией результатов исследования испортился жёсткий диск компьютера, на котором они хранились в единственном экземпляре. Более того, вместе с результатами был утрачен даже список вопросов и вариантов ответов на них! Единственной сохранившейся информацией являются пометки на полях тетради, сделанные во время подсчётов результатов секретаршей Жанной. После обработки каждого варианта она записывала процент людей, выбравших его, в совершенно произвольное место своей тетради ровно один раз.

Долгий и кропотливый процесс восстановления результатов предполагается начать с определения количества вопросов в исходном тестировании. Эта задача поручена вам.

Формат входных данных

В первой строке находится одно целое число K ($1 \leq K \leq 100$) — количество чисел, записанных Жанной на полях тетради, совпадающее с суммарным количеством вариантов ответа на все вопросы.

Во второй строке записаны K неотрицательных целых чисел от 0 до 100, разделённых пробелами, каждое из которых обозначает процент жителей, проголосовавших за какой-то вариант ответа на какой-то вопрос.

Формат выходных данных

Выведите единственное число M — количество вопросов в социологическом опросе. Гарантируется, что существует ответ, удовлетворяющий записям из тетради Жанны. Если определить количество вопросов однозначно не удаётся, выведите любое подходящее значение.

Примеры

<code>c.in</code>	<code>c.out</code>
4 25 50 75 50	2

Замечание

В приведённом примере существует единственный способ получить подходящее разбиение: первое и третье число представляют ответы на один вопрос, а второе и четвёртое — на другой.

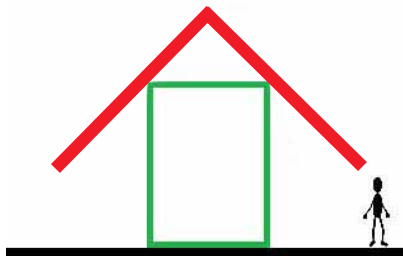
Задача D. Маленькое новоселье

Имя входного файла: d.in
Имя выходного файла: d.out
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Ситуация с землёй в столице одной малоизвестной страны приближается к катастрофической. Уже практически невозможно найти даже небольшой клочок земли для постройки своего жилища.

Тем не менее Фёдору повезло: после прохождения многочисленных инстанций он всё-таки нашел место для нового дома. Более того, щедрое правительство даже подарило ему невероятно красивую крышу для постройки.

Важным обстоятельством является то, что Федя — совершенно плоский, как и весь мир, который его окружает. В этом плоском мире введена стандартная декартова система координат: ось Ox совпадает с землёй, а ось Oy направлена от земли вверх. Крыша представляет собой два отрезка одинаковой длины, исходящих из одной точки (вершины крыши) вниз в разные стороны, причём оба отрезка составляют угол 45° с осью Ox :



Разумеется, крыша находится над землёй, то есть отрезки находятся в верхней полуплоскости. Оба отрезка имеют ненулевую длину. Дом Феде должен представлять собой прямоугольник со сторонами, параллельными осям координат, одна сторона которого находится на земле, а другая сторона упирается в оба отрезка крыши. Дом не должен выходить за пределы крыши по горизонтали. Помогите Феде построить для себя дом наибольшей площади.

Формат входных данных

В первой строке содержатся два целых числа x_c и y_c — координаты вершины крыши ($|x_c| \leq 10^4$, $1 \leq y_c \leq 10^4$). Во второй строке содержатся два целых числа x_1 и y_1 — координаты одного из концов крыши ($|x_1| \leq 10^4$, $x_1 \neq x_c$, $0 \leq y_1 < y_c$). В следующей строке в таком же формате заданы координаты x_2 и y_2 другого конца крыши.

Гарантируется, что отрезки, образующие крышу, направлены в разные стороны относительно верхней точки крыши, т.е. либо $x_1 < x_c < x_2$, либо $x_2 < x_c < x_1$, оба отрезка имеют одинаковую длину и составляют угол 45° с осью Ox .

Формат выходных данных

Выведите единственное число — искомую максимальную площадь дома. Ваш ответ будет считаться правильным, если его абсолютная или относительная ошибка не будет превосходить 10^{-6} .*

Примеры

d.in	d.out
10 10 5 5 15 5	50.0

* А именно, пусть ваш ответ равен A , а ответ жюри — B . Проверяющая программа будет считать ваш ответ правильным, если $\frac{|A-B|}{\max(1, B)} \leq 10^{-6}$.

Задача Е. Здоровый сон

Имя входного файла: `e.in`
Имя выходного файла: `e.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Глеб очень любит спать, поэтому обращает внимание на всевозможные мелочи, сопутствующие этому великолепному занятию. Только представьте: он всегда помнит, какое время показывали часы в момент, когда он засыпал и когда просыпался в последний раз!

Старые электронные часы в комнате Глеба показывают количество часов от 0 до 11 и количество минут от 0 до 59, а индикатор, отвечающий за время суток, давно сломался. Глеб помнит, что перед тем, как он в прошлый раз заснул, часы показывали ровно t_{start} часов (то есть t_{start} часов 0 минут). Когда же он проснулся, на часах было ровно t_{finish} часов. Глеб абсолютно уверен, что спал не менее одного часа, но хотел бы знать точнее. Помогите ему определить минимальное количество часов, которое он мог проспать.

Формат входных данных

В первой строке содержится целое число t_{start} ($0 \leq t_{start} \leq 11$) — время на часах, когда Глеб лёг спать. Во второй строке содержится целое число t_{finish} ($0 \leq t_{finish} \leq 11$) — время на часах, когда Глеб проснулся.

Формат выходных данных

В единственной строке выведите одно целое число — минимальное количество часов, которое мог проспать Глеб.

Примеры

<code>e.in</code>	<code>e.out</code>
7 11	4
10 1	3

Задача F. Средиземье в опасности

Имя входного файла: `f.in`
Имя выходного файла: `f.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Тёмные силы под руководством Саурана заполонили Средиземье, и только Арагорн, сын Араторна, наследник Исилдура и истинный правитель Гондора может найти силы противостоять Тёмному владыке Мордора. Впрочем, мы поможем ему в этом в другой раз, сейчас же давайте оценим, как далеко может зайти Тёмный Властелин.

Карта Средиземья представляет собой клетчатый прямоугольник из N строк по M клеток, каждая из которых может либо полностью принадлежать Сауруну, либо полностью не принадлежать. Если в каком-либо квадрате размером 2×2 три клетки уже захвачены тёмными силами, то они способны захватить и четвёртую клетку.

Изначально полчищам Саурана уже подвластны некоторые клетки на карте Средиземья. Оцените, сколько клеток может оказаться под его контролем в худшем случае.

Формат входных данных

Первая строка входного файла содержит два целых числа N и M ($1 \leq N, M \leq 40$) — количество строк и столбцов на карте Средиземья. Следующие N строк по M символов описывают игровое поле в порядке следования сверху вниз, слева направо. Символ `'.'` соответствует клетке карты, свободной от власти Саурана, а `'#'` — клетке, захваченной Саураном. Строки нумеруются от 1 до N , столбцы — от 1 до M .

Формат выходных данных

В выходной файл выведите одно число — максимальное количество клеток, которые могут контролировать тёмные силы после всех завоеваний.

Примеры

<code>f.in</code>	<code>f.out</code>
2 2 ## #.	4
3 4 #... #... ###.	9
3 5 ...## #.... #.#..	5

Задача G. Турбулентность

Имя входного файла: `g.in`
Имя выходного файла: `g.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Миша решил развлечься в свободное от решения задач время и сейчас проходит квест «Турбулентность». Задание квеста заключается в том, что Мише нужно добыть статуэтку золотого кота. Миша уже добрался до сейфа, в котором предположительно находится статуэтка, и ему осталось лишь подобрать код от замка.

Сообщник Миши оставил ему послание со словом, состоящим из строчных букв латинского алфавита. По опыту предыдущих квестов Миша догадался, что код — это слово минимальной длины, которое не является подстрокой переданного сообщником слова и состоит только из строчных букв латинского алфавита.

Также Миша решил, что перебирать все такие слова будет слишком долго, поэтому в качестве кода от сейфа среди всех подходящих слов минимальной длины он хочет попробовать лексикографически минимальное. Помогите Мише определить, какое слово ему нужно ввести.

Напомним, что *подстрокой* называется некоторая непрерывная часть строки. К примеру, подстроками строки `abacaba` являются `aba`, `bacab`, `a` и многие другие.²

Напомним также, что *лексикографическое сравнение* строк соответствует упорядочиванию слов в словаре, а именно, сначала идут все слова, начинающиеся на букву `a`, затем слова, начинающиеся на букву `b`, и так до конца алфавита. Если у каких-то двух слов первые буквы совпадают, то они сравниваются по второй букве. В случае повторного равенства следует сравнить третью букву и так далее. К примеру, слово `game` лексикографически меньше слова `gate`, так как первые две буквы у них совпадают, а третья идет по алфавиту раньше.³

Формат входных данных

В первой строке содержится единственное целое число N — длина переданного сообщником слова ($1 \leq N \leq 100$).

Во второй строке содержится переданное сообщником слово.

Формат выходных данных

Выведите строку, которую Миша хочет попробовать в качестве кода от сейфа.

Примеры

<code>g.in</code>	<code>g.out</code>
9 pascalabc	d
28 aabcdefghijklmnopqrstuvwxyz	ac

²Более формально, подстрокой строки $s = s_0s_1 \dots s_{n-1}$ с позиции l до позиции r называется строка $s_ls_{l+1} \dots s_r$.

³Строка $s = s_0s_1 \dots s_{n-1}$ лексикографически меньше строки $t = t_0t_1 \dots t_{m-1}$, если найдётся такое i , что $\forall j < i$ $s_j = t_j$ и $s_i < t_i$, а также в случае, если $\forall i < n$ $s_i = t_i$ и при этом $n < m$.

Задача Н. Язык PJ

Имя входного файла: `h.in`
Имя выходного файла: `h.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Мальчик Вартан настолько любит языки программирования, что выучил уже целых девять и обзавёлся самоучителем по ещё одному языку! Впрочем, просто учить языки ему уже немного надоело, и он решил придумать свой. Название для языка было придумано всего за пару минут — *PJ* (аббревиатура от любимых команд Вартана), казалось бы, полдела сделано.

После пяти лет упорной работы Вартан реализовал две команды, работающие с целыми неотрицательными числами:

1. `print value` — число *value* выводится на экран, после чего исполнитель переходит к следующей строке.
2. `jump num count` — если выполнено условие $count > 0$, то управление передаётся в строку с номером *num*, а *count* уменьшается на 1. В текущей реализации PJ не предусмотрено использование в качестве *num* числа большего, чем номер строки, в которой записана данная команда. Если же $count = 0$, то исполнитель просто переходит к следующей строке. Обратите внимание, что в следующий раз, когда интерпретатору встретится данная строка, он будет работать уже с новым значением *count*, то есть можно считать, что каждой команде `jump` соответствует своя глобальная переменная.

Изначально исполнитель находится на первой строке, работа программы начинается с выполнения записанной в ней команды. Выполнение программы заканчивается после того, как исполнитель попытается перейти дальше последней строки. Нетрудно догадаться, что на данном языке невозможно написать программу, не завершающуюся за конечное число итераций.

По имеющейся программе на языке PJ определите сумму всех чисел, которые будут выведены в результате её исполнения.

Формат входных данных

На ввод подаётся корректная программа на языке PJ, состоящая как минимум из одной, но не более чем из 10^5 команд. Количество строк в файле совпадает с количеством команд, каждая команда занимает отдельную строку. Команда, записанная в строке с номером *i* (при нумерации от единицы), удовлетворяет одному из двух форматов:

1. `print value`, где $0 \leq value \leq 10^4$.
2. `jump num count`, где $1 \leq num \leq i$, $0 \leq count \leq 10^4$.

Формат выходных данных

Выведите сумму всех чисел, которые будут выведены в процессе выполнения данной программы.

Примеры

h.in	h.out
<code>print 1</code> <code>print 2</code> <code>jump 2 2</code>	7
<code>print 3</code> <code>jump 1 1</code> <code>print 1</code> <code>print 1</code> <code>jump 1 2</code>	18

Задача I. У любой магии есть своя цена

Имя входного файла: i.in
Имя выходного файла: i.out
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Городок Сторибрук проклят Злой Королевой, и Эмме просто необходимо разрушить её чары. Для того чтобы спасти жителей маленького городка штата Мэн, ей необходимо преобразовать имеющееся заклинание, которое по странному стечению обстоятельств является математическим выражением, в эквивалент более сильного заклинания. Изначально у Эммы имеется выражение $+x_1 + x_2 + \dots + x_n$. Для его преобразования доступны две операции:

1. Заменить в текущем выражении любой знак '+' на '-'.
2. Добавить в выражение пару из открывающейся и закрывающейся скобок. Открывающаяся скобка обязательно ставится сразу после знака '+' или '-', а закрывающаяся обязательно ставится сразу после переменной (и, разумеется, после соответствующей ей открывающейся).

Требуется получить выражение, *эквивалентное*

$$\diamond_1 x_1 \diamond_2 x_2 \diamond_3 \dots \diamond_n x_n,$$

где каждое $\diamond_i$ — это знак '+' или '-'.

Выражения $f(x_1, x_2, \dots, x_n)$ и $g(x_1, x_2, \dots, x_n)$ называются *эквивалентными*, если $f(x_1, x_2, \dots, x_n) = g(x_1, x_2, \dots, x_n) \forall (x_1, x_2, \dots, x_n)$, то есть они равны для любых возможных значений используемых переменных. С точки зрения предлагаемой задачи данное определение означает, что после раскрытия всех скобок каждая переменная встречается в обоих выражениях с одним и тем же знаком.

Эмма использует A единиц магии на выполнение каждой операции первого типа и B единиц магии на выполнение каждой операции второго типа. Помогите Эмме определить, какое минимальное количество единиц магии ей придётся использовать, чтобы преобразовать исходное выражение в эквивалентное требуемому.

Формат входных данных

В первой строке входных данных записано единственное целое число N ($1 \leq N \leq 10^6$) — количество переменных в выражениях. Во второй строке записана строка длины N , состоящая только из символов '+' и '-', i -й символ строки соответствует знаку $\diamond_i$. В третьей строке записаны два целых числа A и B ($0 \leq A, B \leq 10^9$), задающие стоимости операций.

Формат выходных данных

Выведите одно число — искомое минимальное количество единиц магии, необходимое для получения эквивалентного выражения из исходного.

Примеры

i.in	i.out
3 --- 1 0	1
7 ---++-- 2 1	6

Замечание

В первом примере Эмме выгодно сначала заменить первый '+' на '-', затратив одну единицу магии, а потом бесплатно поставить скобки. В результате у нее получится

$$-(x_1 + x_2 + x_3) = -x_1 - x_2 - x_3.$$