

Задача А. Взлет

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

В недавно открытом аэропорту города Январьска готовится к взлету первый самолет. Перед заправщиками стоит сложная задача: сколько топлива заправить в самолет.

Заправщики знают массу самолета и всех людей, которые будут на его борту во время взлета. Взлетной массой самолета называют сумму массы самолета, масс всех людей, находящихся на борту и массы топлива в баках. По правилам авиационных перевозок, для того чтобы самолет мог взлететь на каждую тонну его взлетной массы должно приходиться по крайней мере α килограммов топлива. Какое минимальное количество топлива надо залить в топливные баки, чтобы самолету был разрешен взлет?

Формат входного файла

В первой строке входного файла содержатся три целых числа: M — масса самолета, N — количество людей на борту, α — масса топлива, необходимого для взлета одной тонны самолета. ($1 \leq M \leq 10000, 1 \leq N \leq 100, 1 \leq \alpha \leq 1000$). Вторая строка входного файла содержит N целых чисел m_i — массы пассажиров ($5 \leq m_i \leq 100$).

Все массы во входном файле измеряются в килограммах.

Формат выходного файла

В выходной файл выведите единственное число — минимальную массу топлива в килограммах или “Impossible”, если самолет не может взлететь ни при каких условиях.

Ваш ответ должен иметь абсолютную или относительную погрешность не более 10^{-6} . Это означает следующее. Пусть выведенный вами ответ x , а правильный ответ y . Ответ будет засчитан, если число $|x - y| / \max(1, |y|)$ не превышает 10^{-6} .

Примеры

airplane.in	airplane.out
5000 3 200 100 70 80	1312.5
1000 1 1000 10	Impossible

Примечание

Решения, работающие для случаев, когда ответ целый и не превосходит 10^6 , будут оцениваться в 40 баллов.

Пояснения к первому примеру:

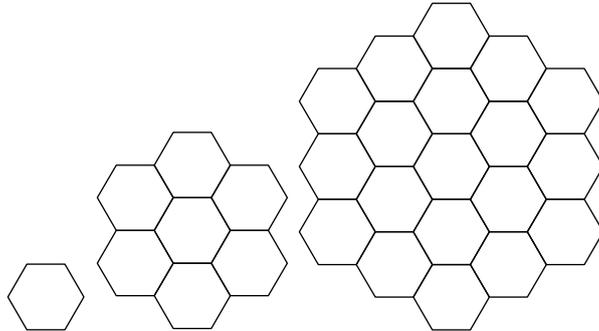
Масса самолета со всеми пассажирами — 5250 килограмм. Если его заправить 1312.5 килограммами топлива, то его масса будет 6562.5 килограмм и на его взлет необходимо ровно $6562.5 * 200 / 1000 = 1312.5$ килограмм топлива.

Задача В. Улей

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

Кевин работает строителем. Но так как Кевин пчела, он строит только улья.

Как известно, улей состоит из одинаковых сот, которые являются правильными шестиугольниками. Среди пчел считается, что улей размера один — это улей, состоящий из единственной соты, улей размера два состоит из улья размера один, окруженного одним кольцом сот, ... улей размера k — это улей размера $k - 1$, окруженный кольцом сот.



Недавно Кевина попросили построить улей размера n . Для планирования строительных работ Кевину необходимо знать точное число сот в улье. Помогите ему.

Напишите программу, которая по размеру улья n будет узнавать число сот в таком улье.

Формат входного файла

Входной файл содержит единственное целое число n ($1 \leq n \leq 10^9$) — размер улья.

Формат выходного файла

В выходной файл выведите единственное число — число сот в улье.

Примеры

<code>beehive.in</code>	<code>beehive.out</code>
1	1
3	19

Примечание

Решения, работающие при $n \leq 15$, будут оцениваться в 30 баллов. Решения, работающие при $n \leq 100000$, будут оцениваться в 60 баллов.

Задача С. Сосна — это дерево

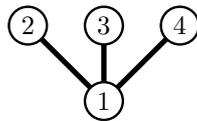
Имя входного файла: `pinetree.in`
 Имя выходного файла: `pinetree.out`
 Ограничение по времени: 2 секунды
 Ограничение по памяти: 256 мегабайт

Будем называть сосной нулевого уровня граф из двух вершин, соединенных ребром. Сосна k -го уровня представляет собой путь, который называется стволом, к некоторым вершинам которого прикреплены сосны уровней не больших $k-1$. Сосны меньших уровней прикреплены таким образом, что одна из крайних вершин их ствола является вершиной ствола новой.

Сосна уровня 0 выглядит следующим образом

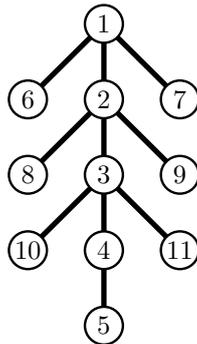


Сосна уровня 1 может выглядеть так:

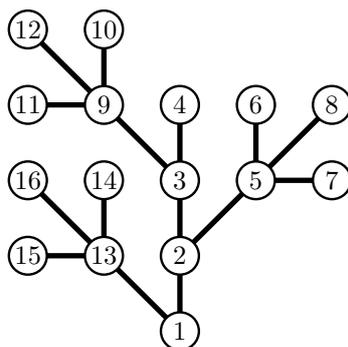


В данном случае одно из ребер графа является стволом, а два других ребра являются соснами уровня ноль.

Или, например, так:



Сосна уровня 2 может выглядеть, например, так:



В данном случае стволом может являться путь $1-2-3-4$ или, например, $12-9-3-2-1-13-16$.

Ваша задача состоит в том, чтобы по графу, представляющему собой дерево, определить, сосной какого минимального уровня является этот граф.

Формат входного файла

В первой строке задано единственное число n — количество вершин дерева ($2 \leq n \leq 100000$). Далее, в каждой из $n - 1$ строк задано по два числа a_i и b_i — номера соединяемых i -м ребром вершин ($1 \leq a_i, b_i \leq n$). Граф, заданный во входном файле, является деревом.

Формат выходного файла

В единственной строке выходного файла выведите единственное целое число — наименьший возможный уровень сосны.

Примеры

pinetree.in	pinetree.out
2 1 2	0
4 1 2 1 3 1 4	1
11 1 2 2 3 3 4 4 5 1 6 1 7 2 8 2 9 3 10 3 11	1
16 1 2 2 3 3 4 2 5 5 6 5 7 5 8 3 9 9 10 9 11 9 12 1 13 13 14 13 15 13 16	2

Примечание

Решения, работающие при $n \leq 15$ и ответе не более 2, будут оцениваться в 30 баллов. Решения, работающие при $n \leq 100$, будут оцениваться в 60 баллов.

Задача D. Маршрутное такси

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

Жители Флатландии обычно передвигаются по городу на таком виде общественного транспорта, как маршрутное такси — он быстрый, удобный и относительно недорогой.

К сожалению, есть одна проблема — производители автобусов для маршрутных такси спроектировали их так, что пассажиры испытывают большие неудобства, пробираясь друг мимо друга к свободным местам в автобусе.

Спиридон — водитель маршрутного такси, который уже давно работает на своем маршруте и знает всех пассажиров, проезжающих на его автобусе изо дня в день. Поэтому, он решил спланировать рассадку пассажиров таким образом, чтобы минимизировать количество прохождений пассажиров друг мимо друга.

Про каждого пассажира известно, когда он входит и выходит из автобуса. Все места расположены в ряд и пронумерованы от 1 до n , начиная от ближайшего места ко входу. Таким образом получается, что когда пассажир, сидящий на i -ом месте входит или выходит, он проходит мимо всех пассажиров, сидящих на местах с номерами меньше i .

Во время поездки пассажиры не перемещаются по автобусу, то есть занимают одно и то же место все время.

Формат входного файла

Первая строка входного файла содержит n ($1 \leq n \leq 100000$) — количество пассажиров автобуса.

Следующие n строк содержат по паре чисел a_i, b_i ($1 \leq a_i < b_i \leq 2n$) — номера остановок, на которых входит и выходит i -ый пассажир.

На каждой остановке входит или выходит не более одного человека.

Формат выходного файла

В первую строку входного файла выведите одно число — минимальное количество прохождений людей друг мимо друга.

Во вторую строку выведите n чисел, i -ое из которых — номер места, на которое должен сесть i -ый пассажир.

Примеры

<code>taxibus.in</code>	<code>taxibus.out</code>
2	0
1 4	2 1
2 3	
5	2
1 8	10 2 4 1 1
3 6	
2 4	
9 10	
5 7	

Примечание

Решения, работающие при $n \leq 10$, будут оцениваться в 30 баллов. Решения, работающие при $n \leq 1000$, будут оцениваться в 60 баллов.