

Задача А. Число Фибоначчи

Имя входного файла: *стандартный ввод*
Имя выходного файла: *стандартный вывод*
Ограничение по времени: 2 секунды
Ограничение по памяти: 512 мегабайт

Числа Фибоначчи $F_0, F_1, F_2, \dots, F_n$ определяются следующим образом:
 $F_0 = 0, F_1 = 1$, а для любого $n > 1$ выполнено равенство $F_n = F_{n-1} + F_{n-2}$.

По заданному числу n выведите число Фибоначчи F_n .

Формат входных данных

В первой строке задано целое число n ($0 \leq n \leq 92$).

Формат выходных данных

В первой строке выведите число F_n .

Примеры

<i>стандартный ввод</i>	<i>стандартный вывод</i>
1	1
2	1
3	2
5	5

Задача В. Зайчик

Имя входного файла: *стандартный ввод*
Имя выходного файла: *стандартный вывод*
Ограничение по времени: 2 секунды
Ограничение по памяти: 512 мегабайт

Зайчик прыгает по прямой просеке, для удобства разделённой на n клеток. Клетки пронумерованы по порядку натуральными числами от 1 до n . Некоторые клетки заболочены: если зайчик прыгнет на такую клетку, ему несдобровать. Некоторые другие клетки просеки поросли вкусной зелёной травой: прыгнув на такую клетку, зайчик сможет отдохнуть и подкрепиться.

Зайчик начинает свой путь из клетки с номером 1 и хочет попасть в клетку с номером n , по пути ни разу не провалившись в болото и скушав как можно больше вкусной зелёной травы. Конструктивные особенности зайчика таковы, что из клетки с номером k он может прыгнуть лишь в клетки с номерами $k + 1$, $k + 3$ и $k + 5$.

Выясните, какое максимальное количество клеток с травой сможет посетить зайчик на своём пути.

Формат входных данных

В первой строке задано число n — количество клеток ($2 \leq n \leq 1000$). Вторая строка состоит из n символов; i -й символ соответствует i -й клетке просеки. Символ «w» обозначает болото, символ «.» — зелёную траву, а символ «.» соответствует клетке без каких-либо особенностей. Гарантируется, что первая и последняя клетки не содержат болот и травы.

Формат выходных данных

В первой строке выведите одно число — максимальное количество клеток с травой, которые зайчик сможет посетить на своём пути. Если зайчику не удастся оказаться в клетке с номером n , выведите -1 .

Примеры

<i>стандартный ввод</i>	<i>стандартный вывод</i>
4 ."".	2
5 .w".	0
9 .www.www.	-1

Задача С. Поле

Имя входного файла: *стандартный ввод*
Имя выходного файла: *стандартный вывод*
Ограничение по времени: 2 секунды
Ограничение по памяти: 512 мегабайт

Отряду нужно пересечь прямоугольное поле размера $m \times n$ квадратов, двигаясь из левого верхнего угла в правый нижний и перемещаясь между соседними квадратами только в двух направлениях — вправо и вниз. Поле не очень ровное, но у отряда есть карта, на которой отмечена высота каждого квадрата. Опасность перехода с квадрата высоты h_1 на соседний квадрат высоты h_2 оценивается числом $|h_2 - h_1|$; опасность всех переходов в пути суммируется. Выясните, какова минимальная опасность пути из квадрата $(1, 1)$ в квадрат (m, n) .

Формат входных данных

В первой строке заданы два числа m и n через пробел ($1 \leq m, n \leq 100$). В следующих n строках записано по m чисел в каждой; i -е число j -й из этих строк соответствует высоте квадрата (i, j) . Все высоты — целые числа в диапазоне от 1 до 100, включительно.

Формат выходных данных

Выведите одно число — минимальную опасность пути из квадрата $(1, 1)$ в квадрат (m, n) .

Примеры

<i>стандартный ввод</i>	<i>стандартный вывод</i>
2 2 1 1 1 1	0
4 2 1 2 3 5 3 8 4 7	6
2 3 1 2 2 3 3 1	4

Задача D. Редакционное расстояние

Имя входного файла: *стандартный ввод*
Имя выходного файла: *стандартный вывод*
Ограничение по времени: 2 секунды
Ограничение по памяти: 512 мегабайт

В информатике *редакционным расстоянием* между двумя строками называется минимальное количество добавлений, удалений и замен символов, при помощи которых можно из одной строки получить другую. К примеру, редакционное расстояние между строками «ab» и «ab» равно нулю, так как строки равны между собой безо всяких изменений; расстояние между строками «short» и «ports» равно трём: в слове «short» нужно удалить из начала букву «s», заменить «h» на «p» и добавить в конец букву «s». Редакционное расстояние также называют *расстоянием Левенштейна*.

Найдите редакционное расстояние между двумя заданными строками.

Формат входных данных

В первой строчке задана одна строка, во второй — другая. Длины обеих строк — от 1 до 100 символов, включительно.

Формат выходных данных

Выведите единственное число — редакционное расстояние между двумя заданными строками.

Примеры

<i>стандартный ввод</i>	<i>стандартный вывод</i>
ab ab	0
short ports	3

Задача Е. Наибольшая общая подпоследовательность

Имя входного файла: *стандартный ввод*
Имя выходного файла: *стандартный вывод*
Ограничение по времени: 2 секунды
Ограничение по памяти: 512 мегабайт

Даны две последовательности. Найдите длину их наибольшей общей подпоследовательности (подпоследовательность — это то, что можно получить из данной последовательности вычёркиванием некоторых элементов).

Формат входных данных

В первой строке записано число N — длина первой последовательности ($1 \leq N \leq 1000$).

Во второй строке записаны члены первой последовательности (через пробел) — целые числа, не превосходящие 10 000 по модулю.

В третьей строке записано число M — длина второй последовательности ($1 \leq M \leq 1000$).

В четвёртой строке записаны члены второй последовательности (через пробел) — целые числа, не превосходящие 10 000 по модулю.

Формат выходных данных

Выведите единственное целое число: длину наибольшей общей подпоследовательности, или же число 0, если такой не существует.

Пример

<i>стандартный ввод</i>	<i>стандартный вывод</i>
3 1 2 3 4 2 1 3 5	2

Задача F. Длина подпоследовательности

Имя входного файла: *стандартный ввод*
Имя выходного файла: *стандартный вывод*
Ограничение по времени: 2 секунды
Ограничение по памяти: 512 мегабайт

Вам требуется написать программу, которая по заданной последовательности находит длину её максимальной невозрастающей подпоследовательности (то есть такой последовательности чисел $a_{i_1}, a_{i_2}, \dots, a_{i_k}$ ($i_1 < i_2 < \dots < i_k$), что $a_{i_1} \geq a_{i_2} \geq \dots \geq a_{i_k}$ и не существует последовательности с теми же свойствами длины $k + 1$).

Формат входных данных

В первой строке задано число n — количество элементов последовательности ($1 \leq n \leq 239\,017$). В последующих строках идут сами числа последовательности a_i , отделённые друг от друга произвольным количеством пробелов и переводов строки (все числа не превосходят по модулю $2^{31} - 2$).

Формат выходных данных

Вам необходимо выдать число k — длину максимальной невозрастающей подпоследовательности.

Пример

<i>стандартный ввод</i>	<i>стандартный вывод</i>
5 5 8 10 4 1	3

Задача G. Новая модель телефона

Имя входного файла: *стандартный ввод*
Имя выходного файла: *стандартный вывод*
Ограничение по времени: 2 секунды
Ограничение по памяти: 512 мегабайт

Компания Spusmas разработала новую модель мобильного телефона. Основное достоинство этой модели — ударопрочность: её корпус сделан из особого сплава, и телефон должен выдерживать падение с большой высоты.

Компания Spusmas арендовала n -этажное здание и наняла экспертов, чтобы те при помощи серии экспериментов выяснили, с какой высоты бросать телефон можно, а с какой — нельзя. Один эксперимент заключается в том, чтобы бросить телефон с какого-то этажа и посмотреть, сломается он от этого или нет. Известно, что любой телефон этой модели ломается, если его сбросить с x -го этажа или выше, где x — некоторое целое число от 1 до n , включительно. Задача экспертов заключается в том, чтобы узнать число x и передать его рекламному отделу компании.

Задача осложняется тем, что экспертам предоставлено всего k образцов новой модели телефона. Каждый телефон можно бросать сколько угодно раз, пока он не сломается; после этого использовать его для экспериментов больше не удастся.

Подумав, эксперты решили действовать так, чтобы минимизировать максимально возможное количество экспериментов, которое может потребоваться произвести. Чему равно это количество?

Формат входных данных

В первой строке входного файла записаны через пробел два целых числа n и k — количество этажей в здании и количество образцов новой модели телефона ($1 \leq n \leq 100\,000$, $0 \leq k < n$).

Формат выходных данных

В выходной файл выведите единственное число — минимальное количество экспериментов, которое потребуется совершить, чтобы узнать число x и использовать не более k телефонов. Если решить задачу невозможно, выведите вместо этого -1 .

Примеры

<i>стандартный ввод</i>	<i>стандартный вывод</i>
4 2	2
4 1	3

Пояснения к примерам

В первом примере сначала следует бросить телефон со второго этажа. Если он сломается, то второй бросок следует сделать с первого этажа. В случае поломки станет известно, что $x = 1$. Иначе мы узнаем, что $x = 2$.

Если же при броске со второго этажа телефон не сломался, бросим телефон с третьего этажа. При поломке будет ясно, что $x = 3$. Иначе из условия $1 \leq x \leq 4$ следует, что $x = 4$.

Всего будет сделано два эксперимента. В них будет использовано не более чем два телефона.

Во втором примере следует сначала бросить единственный данный нам телефон с первого этажа, если он не сломается, то со второго, а если опять не сломается, то с третьего. При первой же поломке мы узнаем точное значение x . Если после трёх бросков телефон так и не сломался, то $x = 4$.

Задача Н. Игра в деление

Имя входного файла: *стандартный ввод*
Имя выходного файла: *стандартный вывод*
Ограничение по времени: 2 секунды
Ограничение по памяти: 512 мегабайт

На доске записано целое число $n > 1$. Ивонна и Зара играют в игру, делая ходы по очереди, начинает Ивонна. За каждый ход игрок может либо поделить число на 2 и округлить вниз, либо поделить на 3 и округлить вверх. После этого старое число стирается, а новое записывается на доску.

Выигрывает та, кто первой запишет на доске число 1. Кто выиграет при правильной игре?

Формат входных данных

В первой строке задано целое число n ($1 < n \leq 10^{18}$).

Формат выходных данных

Если при правильной игре выигрывает Ивонна, выведите «Yvonne».

Если при правильной игре выигрывает Зара, выведите «Zara».

Буквы можно выводить как в верхнем, так и в нижнем регистре.

Примеры

<i>стандартный ввод</i>	<i>стандартный вывод</i>
2	Yvonne
4	Zara

Задача I. Мирные множества

Имя входного файла: *стандартный ввод*
Имя выходного файла: *стандартный вывод*
Ограничение по времени: 2 секунды
Ограничение по памяти: 512 мегабайт

Группа математиков проводит бои между натуральными числами. Результаты боя между двумя натуральными числами, вообще говоря, случайны, однако подчиняются следующему правилу: если одно из чисел не менее чем в два раза превосходит другое, то большее число всегда побеждает; в противном случае победить может как одно, так и другое число.

Бой называется *неинтересным*, если его результат предопределён. Множество натуральных чисел называется *мирным*, если бой любой пары различных чисел из этого множества неинтересен. *Силой* множества называется сумма чисел в нём. Сколько существует мирных множеств натуральных чисел силы n ?

Формат входных данных

В первой строке задано число n ($1 \leq n \leq 2000$).

Формат выходных данных

В первой строке выведите одно число — количество мирных множеств натуральных чисел силы n .

Примеры

<i>стандартный ввод</i>	<i>стандартный вывод</i>
2	1
5	2

Задача J. Требушет Джон

Имя входного файла: *стандартный ввод*
Имя выходного файла: *стандартный вывод*
Ограничение по времени: 8 секунд
Ограничение по памяти: 512 мегабайт

Требушет Джон — основная ударная сила Флатландского королевства. В свободное от бросания девятиногипятикилограммовых валунов время Джон любит наблюдать за плоскими лучниками, играющими в блинчики на плоском озере (блинчики — игра, заключающаяся в бросании плоских камней в воду таким образом, чтобы камень несколько раз отскочил от поверхности воды, прежде чем утонет).

Королевство, которому служит Джон, состоит из n одинаковых провинций, каждая из которых имеет размер d . Так, первой провинции принадлежат точки с x -координатами $(0..d]$, второй — точки с x -координатами $(d..2d]$, и так далее. Более того, каждой провинции соответствует свой коэффициент благополучия w_i . Сам Джон располагается в начале координат.

Будучи настоящим мастером своего дела, Джон может выполнить любой бросок, каким бы сложным тот ни казался на первый взгляд, поэтому развлеченные простые солдаты кажутся ему донельзя занудными, что и побудило его придумать свой аналог блинчиков.

Бросок Джона состоит из прыжков камня. Бросок начинается с того, что Джон выбирает стартовую длину прыжка $k > 0$, после чего забрасывает камень из начала координат в точку k . Для каждого следующего прыжка рассматривается длина предыдущего прыжка p , Джон выбирает какое-то целое число $u > 1$, такое, что p делится нацело на u , и камень прыгает вперёд на расстояние, равное $\frac{p}{u}$. Если такого u не существует, игра заканчивается.

Например, при стартовой длине 6 существует несколько возможных траекторий броска:

- $0 \rightarrow 6 \rightarrow 9 \rightarrow 10$ (при длинах 6, 3, 1 соответственно),
- $0 \rightarrow 6 \rightarrow 8 \rightarrow 9$ (при длинах 6, 2, 1 соответственно),
- $0 \rightarrow 6 \rightarrow 7$ (при длинах 6, 1 соответственно).

Изначально у Джона 0 очков. Каждый раз, когда камень касается земли, Джону начисляется количество очков, равное коэффициенту благополучия той провинции, в которой камень приземлился. При этом камень не может отскочить от земли одной и той же провинции более одного раза: в таком случае очки начисляются только за первое касание, а дальше игра завершается. Также игра завершается в том случае, если камень отскочит от земли,

находящейся за пределами королевства. Заметьте, что, несмотря на то, что до броска камень формально находится в начале координат, он не касается земли в этой точке.

Джон просит вас помочь ему и определить, какое наибольшее количество очков он может набрать, бросив камень наилучшим образом.

Формат входных данных

В первой строке заданы два целых положительных числа — число провинций n и длина одной провинции на карте Флатландского королевства d ($1 \leq n \cdot d \leq 2 \cdot 10^5$). Вторая строка содержит n целых чисел — коэффициенты благополучия ($0 \leq w_i \leq 10^9$).

Формат выходных данных

Выведите одно число — максимальное число очков, которое Джон может набрать в своей игре.

Примеры

<i>стандартный ввод</i>	<i>стандартный вывод</i>
3 5 5 6 6	12
5 20000 1 2 3 4 5	12

Пояснение к примеру

В первом примере выгодно пойти путём $0 \rightarrow 8 \rightarrow 12$ или $0 \rightarrow 9 \rightarrow 12$.