

Задача А. Реверс

Имя входного файла: `reverse.in`
Имя выходного файла: `reverse.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 64 мегабайта

Переверните массив чисел. Без циклов. Без списков. Без контейнеров.

Формат входных данных

В первой строке — число N . Далее в N строках указаны N чисел ($1 \leq N \leq 10000$), по одному числу на строке. Все числа по модулю не превышают 10^5 .

Формат выходных данных

Выведите N чисел — перевернутый массив.

Примеры

<code>reverse.in</code>	<code>reverse.out</code>
4 92846 96841 33582 25998	25998 33582 96841 92846

Задача В. Ханойские башни

Имя входного файла: `hanoi.in`
Имя выходного файла: `hanoi.out`
Ограничение по времени: 6 секунды
Ограничение по памяти: 64 мегабайта

Даны три стержня. На первом стержне находятся несколько дисков сверху вниз по возрастанию размера диска. Два другие пустые. Требуется перенести все диски с первого стержня на второй. Переносить диски разрешается только по одному. Не разрешается класть больший диск на меньший.

Формат входных данных

Вводится одно число n ($1 \leq n \leq 19$) — количество дисков на первом стержне.

Формат выходных данных

Выведите по два числа в строке — номера стержней, откуда и куда переносится диск. Решение должно быть оптимальным по количеству действий.

Примеры

<code>hanoi.in</code>	<code>hanoi.out</code>
3	1 2 1 3 2 3 1 2 3 1 3 2 1 2

Замечание

В программе нельзя пользоваться циклами.

Задача С. Степень двойки

Имя входного файла: `two.in`
Имя выходного файла: `two.out`
Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта

У инопланетян с планеты Уран по два пальца на руке, поэтому им очень нравятся степени двойки. Начинаящий дипломат Василий хочет привезти им N подарков. Чтобы всё прошло идеально, Василий хочет убедиться, что количество подарков соответствует их предпочтениям. Помогите ему в этом!

Формат входных данных

В первой строке дано число N ($1 \leq N \leq 10^9$).

Формат выходных данных

Нужно в единственной строке выходного файла вывести строку "YES", если число является степенью двойки и "NO" в противном случае.

Примеры

<code>two.in</code>	<code>two.out</code>
1	YES
16	YES
24	NO

Замечание

При решении данной задачи **нельзя** использовать циклы (`for`, `while`), а также операцию возведения в степень.

Задача D. Возведение в степень

Имя входного файла: `power.in`
Имя выходного файла: `power.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 64 мегабайта

Формат входных данных

Во входном файле даны три натуральных числа A, B, M ($1 \leq A, B \leq 10^9, 2 \leq M \leq 10^9$), записанные на одной строке через пробел.

Формат выходных данных

В выходной файл выведите одно число, равное $A^B \bmod M$ (`mod` означает взятие остатка при делении).

Примеры

<code>power.in</code>	<code>power.out</code>
2 3 100	8

Замечание

При решении данной задачи нельзя использовать встроенную операцию возведения в степень.

Задача Е. Старый компьютер

Имя входного файла: `oldcomputer.in`
Имя выходного файла: `oldcomputer.out`
Ограничение по времени: 5 секунд
Ограничение по памяти: 64 мегабайта

У Темы очень старый компьютер с очень странной операционной системой. Она настолько странная, что окна на экране не могут пересекаться, но могут быть вложены.

Будем говорить, что окно A вложено в окно B , если все клеточки (пиксели) окна B находятся внутри окна A .

Также будем говорить, что глубина вложенности окон на экране равна X , если существует последовательность окон $a_1, a_2, a_3 \dots a_x$, такая что a_1 вложено в a_2 , a_2 вложено в a_3 , и так далее, а в конце a_{x-1} вложено в a_x , ну и a_x никуда не вложено. И, конечно, не существует более длинной последовательности, которая удовлетворяет таким же требованиям.

Однажды сестра Темы открыла на его компьютере очень много игр в оконном режиме. Все бы было хорошо, но, когда Тема захотел поиграть в свою новую любимую игру, он вспомнил, что, если максимальная вложенность окон больше K и он запустит игру, то компьютер взорвется. Теперь он хочет узнать, какова же текущая глубина вложенности, чтобы закрыть лишние окна.

Формат входных данных

Вам задана карта экрана. Карта представляет из себя табличку $N \times N$ ($N \leq 500$), и про каждый пиксель известны два числа: a и b . Если этот пиксель является верхней левой точкой какого-то окна, то a это высота этого окна, а b это его ширина. a и b заданы в пикселях. Если же пиксель не является левой верхней клеткой, какого-то окна, то $a = b = -1$

В первой строке входного файла содержится число N . В следующих N строках содержится N пар записанных через слэш чисел a_1/b_1 $a_2/b_2 \dots a_n/b_n$

Формат выходных данных

Выведите одно число — глубину вложенности окон на экране.

Примеры

<code>oldcomputer.in</code>	<code>oldcomputer.out</code>
3 2/2 -1/-1 -1/-1 -1/-1 1/1 -1/-1 -1/-1 -1/-1 -1/-1	2