

Задача А. Функция printlist

Имя входного файла: `stdin`
Имя выходного файла: `stdout`
Ограничение по времени: 2 секунды
Ограничение по памяти: 64 мегабайта

Напишите функцию `printlist`, которая получает на вход 4 аргумента: список `s` и строки `before`, `between`, `after`. Эта функция должна выводить на экран список в следующем формате: в начале списка строка `before`, затем элементы списка, разделенные строкой `between`, в конце строка `after`.

Вставьте вашу функцию в начало приведенной ниже программы и сдайте ее на проверку:

```
s = input().split()
printlist(s, 'x=', '*', '.')
printlist(s, ' ', ' ', '')
printlist(s, ' ', ' ', '')
printlist(s, '"', '"', '"', '"')
```

Не забывайте, что список может состоять из 0 или 1 элемента!

Формат входных данных

Вам не надо об этом задумываться.

Формат выходных данных

И об этом тоже.

Примеры

stdin	stdout
abcd e fg	x=abcd*e*fg. abcd e fg abcdefg "abcd","e","fg"

Задача В. Перепись населения

Имя входного файла: `stdin`
Имя выходного файла: `stdout`
Ограничение по времени: 2 секунды
Ограничение по памяти: 64 мегабайта

В главном корпусе живет N человек, возраст каждого варьируется от 0 до 1000 лет. Как-то раз с утра преподаватели решили выяснить, людей какого возраста в главном корпусе больше всего. Разумеется, выполнять эту работу придется вам. Найдите, какой возраст встречается чаще всего, и выведите его.

Формат входных данных

В первой строчке дано целое число N — количество обитателей главного корпуса ($1 \leq N \leq 10^5$). Во второй строчке даны N чисел — их возраста. Каждый возраст — целое число от 0 до 1000.

Формат выходных данных

Выведите самый часто встречающийся возраст (если таких несколько, выведите возраст самых младших).

Примеры

<code>stdin</code>	<code>stdout</code>
2 1 2	1
5 31 861 202 273 672	31
7 16 15 14 17 15 16 16	16

Задача С. Мегавзлом

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

У Егора очень старый телефон. Слава отправил Егору кодовую фразу от Серёжиного банковского счёта, чтоб его ограбить, но телефон Егора настолько стар, что не умеет принимать длинные сообщения. Он разбивает их на несколько маленьких случайной длины, и приходят они ему в случайном порядке.

Когда Егор получил все эти сообщения, он по-настоящему расстроился. Он проклинал телефон, кидал его об стену. К несчастью, телефон оказался обидчивым и перемешал ещё и все буквы в каждом сообщении, хоть и склеил при этом все сообщения в одно. Егор знал, что Серёжа любит такие строки, что в них все буквы идут по убыванию (в обратном алфавитном порядке). Помогите Егору и Славе и найдите кодовую фразу от Серёжиного банковского счёта по единственному оставшемуся сообщению в телефоне.

Формат входных данных

Единственная строка входных данных содержит строку s — сообщение в телефоне Егора ($1 \leq |s| \leq 10^5$). Гарантируется, что s содержит только маленькие латинские буквы.

Формат выходных данных

Выведите кодовую фразу от Серёжиного банковского счёта.

Примеры

<code>writeabc.in</code>	<code>writeabc.out</code>
<code>qwerty</code>	<code>ywtrqe</code>
<code>onehundredseventynine</code>	<code>yvutsronnnnniheeeedd</code>

Замечание

В этой задаче нельзя пользоваться встроенной сортировкой.

Задача D. Частотный анализ

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

Вам дан текст. Мы не спрашиваем вас, что хотел сказать автор; в течение отведенного вам времени выпишите все слова из текста в порядке убывания их частоты.

Формат входных данных

Во входном файле содержится исходный текст. Текст состоит не более чем из 100 000 слов, разделённых пробелами и переводами строк. Все слова состоят из строчных латинских букв. Соседние слова разделены ровно одним пробельным символом. Длина любого слова не превышает 20 символов.

Формат выходных данных

Выведите все слова, встречающиеся в тексте, по одному на каждую строку. Слова должны быть отсортированы по убыванию их количества в тексте, а при равенстве — по алфавиту.

Примеры

frequency.in	frequency.out
hi hi what is your name my name is bond james bond my name is damme van damme claudio van damme jean claudio van damme	damme is name van bond claudio hi my james jean what your
oh you touch my tralala mmm my ding ding dong	ding my dong mmm oh touch tralala you

Задача Е. Полуполный граф

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

Ориентированный граф называется полуполным, если между любой парой его различных вершин есть хотя бы одно ребро. Для заданного списком рёбер графа проверьте, является ли он полуполным. В данном графе нет кратных рёбер.

Формат входных данных

Сначала вводятся целые числа n ($1 \leq n \leq 100$) — количество вершин в графе, и m ($1 \leq m \leq n(n-1)$) — количество ребер. Затем следует m пар целых чисел — ребра графа.

Формат выходных данных

Выведите «YES», если граф является полуполным, и «NO» в противном случае.

Примеры

<code>half-complete.in</code>	<code>half-complete.out</code>
3 4 1 2 2 1 1 3 2 3	YES
3 3 1 2 2 1 2 3	NO

Задача F. Делители

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

Вася любит простые числа, а особенно он любит простые числа, на которые делится его самое любимое число K .

Формат входных данных

На вход подается единственное целое положительное число K , не превосходящее $3 * 10^9$.

Формат выходных данных

В первой строке выведите количество чисел, которые Вася особенно любит. Во второй строке выведите в порядке возрастания все простые числа, которые Вася особенно любит.

Примеры

<code>onedivisor.in</code>	<code>onedivisor.out</code>
5	1 5
12	2 2 3
30	3 2 3 5

Задача G. Поворот матрицы

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

Дана матрица (в виде двумерного массива). Вам надо её повернуть на 90 градусов по часовой стрелке.

Формат входных данных

Сначала заданы числа N и M - ширина и высота матрицы соответственно. В последующих M строчках заданы строки матрицы.

Формат выходных данных

Выведите повернутую матрицу.

Примеры

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

Задача Н. Это бомба!

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

Мише приснился страшный сон — как будто ему очень быстро необходимо разминировать бомбу! На большом дисплее бомбы он видит одно число, рядом есть клавиатура с цифрами (явно для экстренной отмены взрыва), а где-то внизу уже идёт обратный отсчёт. По удачному стечению обстоятельств у него с собой есть инструкция для разминирования, в которой ему нужно срочно разобраться. В такой тяжёлой ситуации мозги у Миши совсем отказали и ему явно нужна помощь. Инструкция:

Посчитайте числовой код для числа на дисплее соответственно приведённым правилам и введите его. Применяется первое подошедшее правило:

1. Если число ≤ 2 , то числовой код равен 1.
2. Если число заканчивается на 7, то нужно отнять от него 5. Посчитайте числовой код для нового числа и прибавьте 1.
3. Если число делится на 4 без остатка, его числовой код равен сумме кодов для числа, делённого на 4 и числа, делённого на 2.
4. Во всех остальных случаях к числу нужно прибавить 1. Посчитайте числовой код для нового числа и прибавьте 2.

Какой числовой код нужно ввести Мише?

Формат входных данных

В единственной строке содержится одно число от 1 до 10^8 , которое отображается на дисплее.

Формат выходных данных

Выведите в ответ одно число, которое Мише нужно срочно ввести.

Примеры

	<code>bomb.in</code>	<code>bomb.out</code>
1	1	1
10	10	12