

Задача А. Подмножества

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

Представьте себе, что вы преподаватель. У вас есть N задач, и вам надо составить из них констест. Но пока вы не попробуете все варианты, вы не поймете, какой — лучший. Чтобы ничего не забыть, вы решили составить список из всех возможных наборов задач. Естественно, в констесте должна быть хотя бы одна задача.

Формат входных данных

Дано единственное число N ($1 \leq N \leq 10$).

Формат выходных данных

Выведите искомые подмножества множества $\{1, 2, \dots, N\}$. В каждой строке выведите одно подмножество. Первое число в строке — количество задач в варианте, остальные — номера задач в варианте. Подмножества можно выводить в любом порядке.

Примеры

<code>subsequences.in</code>	<code>subsequences.out</code>
1	1 1
2	1 1 1 2 2 1 2

Задача В. Все двоичные строки длины n, содержащие ровно k единиц

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

По данным числам N и K выведите все строки из нулей и единиц длины N , содержащие ровно K единиц, в лексикографическом порядке.

Формат входных данных

Заданы 2 числа: N и K ($0 \leq K \leq N$, $0 \leq N \leq 100$).

Формат выходных данных

Необходимо вывести все строки из нулей и единиц длины N , содержащие ровно K единиц, в лексикографическом порядке. Гарантируется, что размер ответа не превышает 10 мегабайт.

Примеры

<code>combnk.in</code>	<code>combnk.out</code>
4 2	0011 0101 0110 1001 1010 1100

Задача С. Сортировка

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

Дан массив целых чисел. Ваша задача — отсортировать его в порядке неубывания.

Формат входных данных

В первой строке входного файла содержится число N ($1 \leq N \leq 100\,000$) — количество элементов в массиве. Во второй строке находятся N целых чисел, по модулю не превосходящих 10^9 .

Формат выходных данных

В выходной файл надо вывести этот же массив в порядке неубывания, между любыми двумя числами должен стоять ровно один пробел.

Примеры

<code>sort.in</code>	<code>sort.out</code>
10 1 8 2 1 4 7 3 2 3 6	1 1 2 2 3 3 4 6 7 8

Замечание

При решении задачи нельзя использовать встроенные функции `sort` и `sorted`.

Задача D. Инвентаризация

Имя входного файла: `sort.in`
Имя выходного файла: `sort.out`
Ограничение по времени: 8 секунды
Ограничение по памяти: 64 мегабайта

В связи с модернизацией производства на заводе зубных щеток в Тау Кита было решено переписать список роботов, обслуживающих завод. Каждый робот имеет 2 номера: основной и вспомогательный. Новый список должен удовлетворять следующим правилам:

1. Если один робот в новом списке находится раньше другого, то основной номер первого меньше или равен основному номеру второго.

2. Если основные номера роботов равны, то они расположены в таком же порядке, как и в исходном списке.

Тау Китяне обратились к Вам с просьбой переписать список. Помогите модернизации организаций!

Формат входных данных

В первой строке входного файла содержится число N ($1 \leq N \leq 100\,000$) — количество роботов на заводе. На каждой следующей строке находятся 2 числа — основной и вспомогательный номера очередного робота. Оба номера неотрицательны и не превосходят 10^9 .

Формат выходных данных

Выведите N строчек, i -ая содержит 2 числа — основной и вспомогательный номер i -го робота в новом списке.

Примеры

sort.in	sort.out
10	1 8
1 8	1 11
8 9	2 10
2 10	2 23
1 11	3 11
4 2	3 3
7 2	4 2
3 11	6 7
2 23	7 2
3 3	8 9
6 7	