

Задача А. Двоичные строки заданной длины

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

По данному числу N выведите все строки длины N из нулей и единиц в лексикографическом порядке.

Формат входных данных

Задано единственное число N . (натуральное, $1 \leq N \leq 10$)

Формат выходных данных

Необходимо вывести все строки длины N из нулей и единиц в лексикографическом порядке, по одной на строке.

Пример

стандартный ввод	стандартный вывод
3	000 001 010 011 100 101 110 111

Задача В. Все строки длины n из k различных символов

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

По данным числам N и K выведите все строки длины N из символов $0 \dots K - 1$ в лексикографическом порядке.

Формат входных данных

Заданы 2 числа: N и K ($1 \leq N, K \leq 10; N + K < 15$)

Формат выходных данных

Необходимо вывести все строки длины N из символов $0 \dots K - 1$ в лексикографическом порядке.

Пример

стандартный ввод	стандартный вывод
2 3	00 01 02 10 11 12 20 21 22

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

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

По данным числам sN 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.

Пример

стандартный ввод	стандартный вывод
4 3	0111 1011 1101 1110

Задача D. Все перестановки заданной длины

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

По данному числу N выведите все перестановки чисел от 1 до N в лексикографическом порядке.

Формат входных данных

Задано 1 число: N ($0 < N < 10$).

Формат выходных данных

Необходимо вывести все перестановки чисел от 1 до N в лексикографическом порядке. Перестановки выводятся по одной в строке, числа в перестановке выводятся без пробелов.

Пример

стандартный ввод	стандартный вывод
3	123 132 213 231 312 321

Задача E. Все убывающие последовательности длины k из чисел $1\dots n$

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

По данным числам N и K выведите все убывающие последовательности длины K из чисел $1\dots N$ в лексикографическом порядке.

Формат входных данных

Заданы 2 числа: N и K . ($1 \leq N, K \leq 100$)

Формат выходных данных

Необходимо вывести все убывающие последовательности длины K из чисел $1\dots N$ в лексикографическом порядке. Последовательности выводятся по одной в строке, числа внутри последовательностей разделяются пробелами

Пример

стандартный ввод	стандартный вывод
5 2	2 1 3 1 3 2 4 1 4 2 4 3 5 1 5 2 5 3 5 4

Задача F. Разбиения на слагаемые

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 2 секунды
Ограничение по памяти: 64 мегабайта

Перечислите все разбиения целого положительного числа N на целые положительные слагаемые. Разбиения должны обладать следующими свойствами:

- Слагаемые в разбиениях идут в невозрастающем порядке.
- Разбиения перечисляются в лексикографическом порядке.

Формат входных данных

Во входном файле находится единственное число N ($1 \leq N \leq 40$).

Формат выходных данных

В выходной файл выведите искомые разбиения по одному на строку.

Пример

стандартный ввод	стандартный вывод
4	1 1 1 1 2 1 1 2 2 3 1 4

Задача G. Разбиение на неубывающие слагаемые, обратный порядок

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Дано натуральное число N . Рассмотрим его разбиение на различные натуральные слагаемые. Два разбиения, отличающихся только порядком слагаемых, будем считать за одно, поэтому можно считать, что слагаемые в разбиении упорядочены по неубыванию.

Формат входных данных

Задано единственное число N . ($1 \leq N \leq 40$)

Формат выходных данных

Необходимо вывести все разбиения числа N на различные натуральные слагаемые. Слагаемые выводите по неубыванию.

Пример

стандартный ввод	стандартный вывод
5	5 2 3 1 4 1 2 2 1 1 3 1 1 1 2 1 1 1 1 1

Задача Н. Разложение на множители

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Дано натуральное число N . Требуется вывести все разбиения числа N на натуральные делители не меньше M и большие 1.

Формат входных данных

Даны два числа N и M ($2 \leq M \leq N \leq 1000$).

Формат выходных данных

В выходной файл требуется вывести все разбиения на множители в лексикографическом порядке. Разбиения должны сравниваться не как строки, а как последовательности чисел. Внутри разбиения множители должны идти в неубывающем порядке.

Примеры

стандартный ввод	стандартный вывод
4 3	4
18 3	3*6 18

Задача I. Мирные ферзи

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Дано число N . Определите, сколькими способами можно расставить на доске $N \times N$ N ферзей, не бьющих друг друга.

Формат входных данных

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

Формат выходных данных

Необходимо вывести количество способов, которыми можно расставить на доске $N \times N$ N ферзей, не бьющих друг друга.

Пример

стандартный ввод	стандартный вывод
5	10

Задача J. Генерация правильных скобочных последовательностей - 2

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

По данному числу n выведите все правильные скобочные последовательности из круглых и квадратных скобок длины n . Придерживайтесь следующего порядка скобок: " $()$ " (см. тест из условия)

Формат входных данных

Одно целое число n ($0 \leq n \leq 16$).

Формат выходных данных

Выведите все правильные скобочные последовательности из круглых и квадратных скобок длины n в лексикографическом порядке. Каждая последовательность должна выводиться в новой строке.

Пример

стандартный ввод	стандартный вывод
4	$()()$ $([])$ $()()$ $()[]$ $[()]$ $[[]]$ $[]()$ $[] []$

Задача К. Монетки

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Ваня пришел в магазин, чтобы купить призы за олимпиаду. У Вани есть n номиналов монет. i -й номинал стоит a_i рублей, у Вани есть по 2 монеты каждого номинала. Приз стоит k рублей. Поскольку в этом магазине сдачи нет, а переплачивать Ваня не собирается, то ему нужно набрать монет на ровно k рублей. Ваня очень занятой, поэтому поручил эту задачу вам!

Формат входных данных

В первой строке находятся 2 целых числа n и k ($1 \leq n \leq 10$, $1 \leq k \leq 50$).
Во второй строке находятся n целых чисел, i -е из которых равно a_i ($1 \leq a_i \leq 50$).

Формат выходных данных

Если Ваня может набрать монет ровно на k рублей, то в первой строке выведите «Yes» (без кавычек).

Во второй строке выведите количество монет, которые Ваня будет использовать.

В третьей строке выведите список стоимостей монет, которыми расплатится Ваня. Их сумма должна быть равна k .

Если Ваня не может расплатиться, то в первой строке выведите «No» (без кавычек).

Примеры

стандартный ввод	стандартный вывод
2 5 2 1	Yes 3 2 2 1
3 1 2 3 4	No

Задача L. Теорема Лагранжа

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Теорема Лагранжа утверждает, что любое натуральное число можно представить в виде суммы четырех точных квадратов. По данному числу n найдите такое представление: напечатайте 4 целых неотрицательных числа, квадраты которых дают в сумме данное число.

Формат входных данных

Программа получает на вход одно натуральное число $n < 10000$.

Формат выходных данных

Программа должна вывести 4 целых неотрицательных числа, квадраты которых дают в сумме данное число.

Пример

стандартный ввод	стандартный вывод
7	1 1 1 2

Задача М. Супер префиксная сумма

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 10 секунд
Ограничение по памяти: 1024 мегабайта

Вам дан k мерный гиперпараллелепипед a размером $n_1 \times n_2 \times \dots \times n_k$ и q запросов вида $l_i^1, l_i^2, \dots, l_i^k, r_i^1, r_i^2, \dots, r_i^k$. Для каждого запроса найдите сумму на подгиперпараллелепипеде $[(l_i^1, l_i^2, \dots, l_i^k), (r_i^1, r_i^2, \dots, r_i^k)]$ гиперпараллелепипеда a .

Формат входных данных

В первой строке вводится одно число k — размерность входного гиперпараллелепипеда.

Во второй строке вводятся k чисел n_i — размеры гиперпараллелепипеда.

В третьей строке вводится $n_1 \times n_2 \times \dots \times n_k$ чисел — значения элементов гиперпараллелепипеда в лексикографическом порядке вектора индексов, то есть в порядке $a_{1,1,\dots,1}, a_{1,1,\dots,2}, \dots, a_{1,1,\dots,n_k}, a_{1,\dots,2,1}, \dots, a_{1,\dots,2,n_k}, \dots, a_{1,\dots,n_{k-1},n_k}, \dots, a_{n_1,n_2,\dots,n_k}$.

В четвертой строке вводится число q — количество запросов поиска суммы.

В следующих q строках вводится по $2 \cdot n$ чисел $l_i^1, l_i^2, \dots, l_i^k, r_i^1, r_i^2, \dots, r_i^k$ — описания запросов.

Также выполняются неравенства:

- $1 \leq k \leq 10, 1 \leq n_i \leq 10^5$
- $1 \leq n_1 \times n_2 \times \dots \times n_k \leq 10^6$
- $1 \leq n_1 \times n_2 \times \dots \times n_k \times 2^k \leq 10^8$
- $-10^9 \leq a_{i_1,i_2,\dots,i_k} \leq 10^9, 1 \leq l_i^j \leq r_i^j \leq n_j$.

Формат выходных данных

Выведите q строк, в каждой ответ на очередной запрос.

Примеры

стандартный ввод	стандартный вывод
2	-1
3 4	3
1 -5 3 0 4 5 -9 1 -3 6 1 2	6
6	2
1 1 2 3	-1
2 2 3 3	-5
1 1 3 4	
1 1 3 1	
1 1 1 4	
2 3 3 4	
3	38
3 2 3	21
2 4 3 0 0 1 1 5 4 9 0 3 0 5 1 0 3 9	15
3	
1 1 2 3 2 3	
1 1 1 2 2 2	
2 1 2 3 1 3	