Задача A. Без двух единиц подряд

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

По данному натуральному числу n выведите все двоичные последовательности длины n, не содержащие двух единиц подряд, в лексикографическом порядке.

Формат входных данных

Одно натуральное число n ($1 \leqslant n \leqslant 20$).

Формат выходных данных

Каждая последовательность должна выводиться в отдельной строке, вывод должен завершаться символом новой строки. Числа, входящие в последовательность, должны быть разделены одним пробелом.

Примеры

notwo.in	notwo.out
4	0 0 0 0 0 0 0 1 0 0 1 0 0 1 0 0 0 1 0 1 1 0 0 0 1 0 0 1 1 0 1 0

Задача B. Номер по перестановке

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

Дана перестановка из N чисел от 1 до N. Требуется найти её номер в лексикографическом порядке.

Формат входных данных

Во входном файле сначала записано число N ($1 \leqslant N \leqslant 12$). В следующей строке записана сама перестановка — N чисел, разделённых пробелами.

Формат выходных данных

В выходной файл нужно вывести единственное число — номер перестановки в лексикографическом порядке.

Примеры

perm.in	perm.out
3 2 1 3	3

Задача C. Все перестановки заданной длины

Имя входного файла:	`perm.in`
Имя выходного файла:	`perm.out`
Ограничение по времени:	3 секунды
Ограничение по памяти:	64 мегабайта

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

Формат входных данных

Задано одно число N ($0 < N < 9$).

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.

Примеры

perm.in	perm.out
3	123 132 213 231 312 321

Замечание

В задаче нельзя использовать функцию std::next_permutation.

Задача D. Разбиения на слагаемые

Имя входного файла:	`partition.in`
Имя выходного файла:	`partition.out`
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 ($1 \leqslant N \leqslant 40$).

Формат выходных данных

В выходной файл выведите искомые разбиения по одному на строку.

Примеры

partition.in	partition.out
4	1 1 1 1 2 1 1 2 2 3 1 4

Задача E. Следующее сочетание

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

Дано множество целых чисел от 1 до N. Рассмотрим подмножество этого множества, состоящее из K элементов, в возрастающем порядке.

Выведите следующее в лексикографическом порядке подмножество из K элементов.

Формат входных данных

В первой строке входного файла содержатся целые положительные числа N и K ($1 \leqslant K \leqslant N \leqslant 50$). Во второй строке содержится K целых чисел от 1 до N в возрастающем порядке — подмножество из K элементов.

Формат выходных данных

Выведите следующее в лексикографическом порядке после данного подмножество из K элементов. Если следующего подмножества нет, выведите 0.

Примеры

nextcomb.in	nextcomb.out
6 4 1 4 5 6	2 3 4 5
6 2 5 6	0