

Задача А. Количество перестановок

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

Дано целое число N . Требуется подсчитать и вывести количество различных перестановок чисел от 1 до N .

Формат входного файла

Во входном файле записано одно число N ($1 \leq N \leq 12$).

Формат выходного файла

В выходной файл нужно вывести количество различных перестановок чисел от 1 до N .

Примеры

<code>numperm.in</code>	<code>numperm.out</code>
3	6

Задача В. Следующая перестановка

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

Найдите следующую перестановку. Тождественная перестановка является следующей для обратной.

Формат входного файла

В первой строке входного файла записано число N ($1 \leq N \leq 100\,000$) — количество элементов в перестановке. Во второй строке записана перестановка из N чисел.

Формат выходного файла

В выходной файл вывести N чисел — искомую перестановку.

Примеры

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

Задача С. Перестановка по номеру

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

Найдите перестановку по её номеру в лексикографическом порядке.

Формат входного файла

В первой строке входного файла записано число N ($1 \leq N \leq 12$) — количество элементов в перестановке. Во второй строке число K ($1 \leq K \leq N!$) — номер перестановки.

Формат выходного файла

В выходной файл вывести N чисел — искомую перестановку.

Примеры

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

Задача D. 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 входного файла

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

Формат выходного файла

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

Примеры

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

Задача Е. 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 входного файла

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

Формат выходного файла

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

Задача F. Разбиения на слагаемые

Имя входного файла: `partition.in`
Имя выходного файла: `partition.out`
Ограничение по времени: 1 секунда
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 входного файла

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

Формат выходного файла

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