

Задача А. Предыдущая перестановка

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

Найдите предыдущую в лексикографическом порядке перестановку. Перестановка вида $N, N - 1, \dots, 3, 2, 1$ является предыдущей для $1, 2, 3, \dots, N - 1, N$

Формат входных данных

В первой строке входного файла записано число N ($1 \leq N \leq 10^5$) количество элементов в перестановке. Во второй строке записана перестановка.

Формат выходных данных

В выходной файл вывести N чисел — искомую перестановку.

Примеры

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

Задача В. 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 ($1 \leq K \leq N \leq 50$). Во второй строке содержится K целых чисел от 1 до N в возрастающем порядке — подмножество из K элементов.

Формат выходных данных

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

Примеры

<code>nextcomb.in</code>	<code>nextcomb.out</code>
6 4 1 4 5 6	2 3 4 5
6 2 5 6	0

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

Выведите перестановку по её номеру.

Формат входных данных

В первой строке входного файла записано число N ($1 \leq N \leq 12$) — количество элементов в перестановке. Во второй строке записано число K ($0 \leq K < N!$) — номер перестановки в нумерации с нуля.

Формат выходных данных

В выходной файл выведите N чисел через пробел — искомую перестановку.

Примеры

<code>bynumber.in</code>	<code>bynumber.out</code>
3 0	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 входных данных

Во входном файле сначала записано число N ($1 \leq N \leq 12$). 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	3
2 1 3	

Задача Е. Карточки

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

На день рождения Пете подарили набор карточек с буквами. Теперь Петя с большим интересом составляет из них разные слова. И вот, однажды, составив очередное слово, Петя заинтересовался вопросом: «А сколько различных слов можно составить из тех же карточек, что и данное?». Помогите ему ответить на этот вопрос.

Формат входных данных

Вводится слово, составленное Петей — строка из маленьких латинских букв не длиннее 15 символов.

Формат выходных данных

Выведите одно целое число — искомое количество слов.

Примеры

<code>cards.in</code>	<code>cards.out</code>
solo	12

Задача F. Монетки

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

В Волшебной стране используются монетки достоинством A_1, A_2, \dots, A_M . Волшебный человечек пришел в магазин и обнаружил, что у него есть ровно по две монетки каждого достоинства. Ему нужно заплатить сумму N . Напишите программу, определяющую, сможет ли он расплатиться без сдачи.

Формат входных данных

Сначала вводится целое число N ($1 \leq N \leq 10^9$), затем — целое число M ($1 \leq M \leq 10$) и далее M попарно различных целых чисел A_1, A_2, \dots, A_M ($1 \leq A_i \leq 10^9$).

Формат выходных данных

Выведите сначала K — количество монет, которое придется отдать Волшебному человечку, если он сможет заплатить указанную сумму без сдачи. Далее выведите K чисел, задающих достоинства монет. Если решений несколько, выведите вариант, в котором Волшебный человек отдаст наименьшее возможное количество монет. Если таких вариантов несколько, выведите любой из них.

Если без сдачи не обойтись, то выведите одно число 0. Если же у Волшебного человечка не хватит денег, чтобы заплатить указанную сумму, выведите одно число -1 (минус один).

Примеры

<code>coins.in</code>	<code>coins.out</code>
5 2 1 2	3 1 2 2
7 2 1 2	-1
5 2 3 4	0