

Задача А. 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 В. 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 С. 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 человек 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 человеку, 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 человека 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

Задача D. 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 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.

Примеры

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

Задача Е. Предыдущая правильная скобочная последовательность

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

По правильной скобочной последовательности вывести последовательность, предшествующую ей в лексикографическом порядке.

Формат входных данных

На вход подается одна строка, состоящая из $2N$ ($N \leq 10000$) символов - исходная правильная скобочная последовательность.

Формат выходных данных

Вывести в единственной строке искомую последовательность. Если исходная последовательность - первая в лексикографическом порядке, то следует вывести последнюю в лексикографическом порядке правильную скобочную последовательность такой же длины.

Примеры

prev.in	prev.out
()()	(())
()(())	(())()

Задача F. Конфеты Кирилла

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

У Кирилла было k конфет, и он захотел их раздать ученикам своей параллели. Однако

заметил, что конфет у него меньше чем учеников в параллели. Кирилл сел на скамейку и задумался. Просидев полчаса и доев последнюю конфету он подумал — интересно, а сколько было способов раздать все k конфет n ученикам параллели С, если конфеты нельзя делить, а каждому школьнику можно дать не более одной конфеты.

Формат входных данных

В единственной строке записаны числа n, k ($1 \leq k \leq n \leq 64$).

Формат выходных данных

Выведите единственное число — ответ на задачу.

Примеры

combination.in	combination.out
5 3	10