

Задача А. Перестановка по номеру (1 балл)

Имя входного файла: `num2perm.in`Имя выходного файла: `num2perm.out`

Во входном файле задано числа n и k . Выведите в выходной файл k -ю в лексикографическом порядке перестановку чисел от 1 до n . Перестановки занумерованы от 0 до $n! - 1$. $1 \leq n \leq 18$, $0 \leq k \leq n! - 1$.

Пример

<code>num2perm.in</code>	<code>num2perm.out</code>
3 4	3 1 2

Задача В. Номер по перестановке (1 балл)

Имя входного файла: `perm2num.in`Имя выходного файла: `perm2num.out`

Во входном файле задано число n и затем перестановка чисел от 1 до n . Выведите в выходной файл номер заданной перестановки в лексикографическом порядке всех перестановок чисел от 1 до n . Перестановки занумерованы, начиная с 0. $1 \leq n \leq 18$.

Пример

<code>perm2num.in</code>	<code>perm2num.out</code>
3 1 3 2	1

Задача С. Сочетание по номеру (1 балл)

Имя входного файла: `num2choose.in`

Имя выходного файла: `num2choose.out`

Во входном файле заданы числа n , k и m . Выведите в выходной файл m -е в лексикографическом порядке сочетание по k из чисел от 1 до n . Сочетания занумерованы, начиная с 0. $1 \leq k \leq n \leq 30$, $0 \leq m \leq \binom{n}{k} - 1$.

Пример

<code>num2choose.in</code>	<code>num2choose.out</code>
4 2 3	2 3

Задача D. Номер по сочетанию (1 балл)

Имя входного файла: `choose2num.in`

Имя выходного файла: `choose2num.out`

Во входном файле заданы числа n , k и затем сочетание, состоящее из k чисел от 1 до n . Выведите в выходной файл номер этого сочетания в лексикографическом порядке всех сочетаний из n чисел по k ($1 \leq k \leq n \leq 30$). Сочетания нумеруются, начиная с 0.

Пример

<code>choose2num.in</code>	<code>choose2num.out</code>
4 2 2 3	3

Задача Е. Правильная скобочная последовательность по номеру (1 балл)

Имя входного файла: `num2brackets.in`

Имя выходного файла: `num2brackets.out`

Во входном файле заданы числа n и k . Выведите в выходной файл k -ю в лексикографическом порядке правильную скобочную последовательность среди всех правильных скобочных последовательностей с n открывающимися скобками, упорядоченных в лексикографическом порядке, «(» < «)». Последовательности занумерованы, начиная с 0. $1 \leq n \leq 20$. Искомая последовательность существует.

Пример

<code>num2brackets.in</code>	<code>num2brackets.out</code>
4 3	((()))()

Задача F. Номер по правильной скобочной последовательности (1 балл)

Имя входного файла: `brackets2num.in`

Имя выходного файла: `brackets2num.out`

Во входном файле задана правильная скобочная последовательность. Выведите в выходной ее номер в лексикографическом порядке среди всех правильных скобочных последовательностей с таким же количеством открывающихся скобок, «(» < «)». Последовательности занумерованы, начиная с 0. Количество открывающихся скобок в последовательности — от 1 до 20.

Пример

<code>brackets2num.in</code>	<code>brackets2num.out</code>
<code>((()))()</code>	3

Задача G. Правильная скобочная последовательность с двумя типами скобок по номеру (1 балл)

Имя входного файла: num2brackets2.in

Имя выходного файла: num2brackets2.out

Во входном файле заданы числа n и k . Выведите в выходной файл k -ю в лексикографическом порядке правильную скобочную последовательность среди всех правильных скобочных последовательностей с двумя типами скобок с n открывающимися скобками, упорядоченных в лексикографическом порядке, «(» < «)» < «[» < «]». Последовательности занумерованы, начиная с 0. $1 \leq n \leq 20$. Искомая последовательность существует.

Пример

num2brackets2.in	num2brackets2.out
4 100	([]) ([]

Задача Н. Номер по правильной скобочной последовательности с двумя типами скобок (1 балл)

Имя входного файла: `brackets2num2.in`

Имя выходного файла: `brackets2num2.out`

Во входном файле задана правильная скобочная последовательность с двумя типами скобок. Выведите в выходной ее номер в лексикографическом порядке среди всех правильных скобочных последовательностей с таким же количеством открывающихся скобок, «(» < «)» < «[» < «]». Последовательности занумерованы, начиная с 0. Количество открывающихся скобок в последовательности — от 1 до 20.

Пример

<code>brackets2num2.in</code>	<code>brackets2num2.out</code>
<code>([]) ([]</code>	100

Задача I. Разбиение на слагаемые по номеру (1 балл)

Имя входного файла: num2part.in
Имя выходного файла: num2part.out

Рассмотрим все разбиения числа n на слагаемые, в каждом разбиении упорядочим их в порядке не убывания. Будем считать, что разбиение $a_1+a_2+\dots+a_n$ лексикографически меньше $b_1+b_2+\dots+b_m$, если для некоторого $k \forall j \leq k : a_j = b_j$ и либо $k = n$, либо $a_{k+1} < b_{k+1}$. Во входном файле заданы числа n и r . $1 \leq n \leq 100$, разбиение с номером r — существует. Выведите r -ое разбиение числа n на слагаемые, разбиения нумеруются с 0.

Пример

num2part.in	num2part.out
4 3	2+2

Задача J. Номер по разбиению на слагаемые (1 балл)Имя входного файла: `part2num.in`Имя выходного файла: `part2num.out`

Рассмотрим все разбиения числа n на слагаемые, в каждом разбиении упорядочим их в порядке не убывания. Будем считать, что разбиение $a_1+a_2+\dots+a_n$ лексикографически меньше $b_1+b_2+\dots+b_m$, если для некоторого $k \forall j \leq k : a_j = b_j$ и либо $k = n$, либо $a_{k+1} < b_{k+1}$. Во входном файле задано разбиение на слагаемые. Выведите номер этого разбиения, среди всех разбиений упорядоченных лексикографически. Разбиения нумеруются с 0. Гарантируется, что в разбиении слагаемые упорядочены в порядке не убывания, и $1 \leq n \leq 100$.

Пример

<code>part2num.in</code>	<code>part2num.out</code>
2+2	3