

**Задача А. 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 входного файла**

Задано одно число  $N$  ( $0 < N < 9$ ).

**Формат выходного файла**

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

**Задача В. Следующая перестановка**

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

Найдите следующую перестановку. Лексикографически первая 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**

nextperm.in	nextperm.out
3 3 2 1	1 2 3
2 1 2	2 1

**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

Демиурги Шамбамбукли и Мазукта уже достигли невероятного мастерства в искусстве сотворения миров. Но им, как никому другому, известно, что нет предела совершенству. Неудивительно, что время от времени они собираются в на скорую руку сотворённом кафе за чашечкой кофе, чтобы вместе постигать вершины своего искусства. По своему немалому опыту они знают, что самые интересные и сложные закономерности проще всего обнаружить на самых простых примерах.

В этот раз они решили посмотреть, что выйдет из миров, в основу которых положены самые обычные перестановки. Чтобы не упустить ни один из возможных вариантов, демиурги решили использовать перестановки для сотворения миров по очереди, в лексикографическом порядке.

Шамбамбукли и Мазукта уже успешно создали  $K - 1$  мир, но тут их настиг творческий кризис. Теперь только вы можете помочь им создать очередной мир и вдохновить их этим поступком на завершение эксперимента. По счастливому совпадению, этот мир как раз должен оказаться Хрымбелем, который демиурги в прошлый раз так и не смогли сотворить.

**Формат входного файла**

В первой строке входного файла записано число  $N$  ( $1 \leq N \leq 12$ ) — количество элементов в перестановке, которая должна быть положена в основу Хрымбеля. Во второй строке число  $K$  ( $1 \leq K \leq N!$ ) — номер перестановки.

**Формат выходного файла**

В выходной файл вывести  $N$  чисел через пробел — первооснову Хрымбеля.

**Примеры**

bynumber.in	bynumber.out
3 1	1 2 3

**Задача D. 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 Е. 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

perm.in	perm.out
3 2 1 3	3

### Задача F. Как убить время

Имя входного файла: num2part.in  
Имя выходного файла: num2part.out  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 64 мегабайта

Некто Симон уже давно дорешал все задачи из констестов и теперь убивает время на практике тем, что раскладывает найденные у комповника  $n$  камешков в кучки разного

размера. Изначально он разложил все камешки по одному. В каждую следующую минуту он выкладывает из них следующее лексикографическое разбиение на кучки. Порядок разбиений будем определять следующим образом: рассмотрим все разбиения  $n$  камешком на кучки, 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  $a_{k+1} < b_{k+1}$ .

### Формат входного файла

Во входном файле заданы числа  $n$  и  $r$ .  $1 \leq n \leq 100$ , разбиение с номером  $r$  — существует.

### Формат выходного файла

Выведите  $r$ -ое разбиение  $n$  камешков на кучки, разбиения нумеруются с 0.

### Примеры

num2part.in	num2part.out
4 3	2+2
5 5	2+3