

Задача А. 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 В. Перестановка по номеру

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

Демииурги 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, демииурги 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 демииурги в прошлый раз так и не смогли сотворить.

Формат входных данных

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 выходных данных

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

Примеры

bynumber.in	bynumber.out
3	1 2 3
1	

Задача С. Номер по правильной скобочной последовательности

Имя входного файла: brackets2num.in
Имя выходного файла: brackets2num.out
Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта

Дана правильная скобочная последовательность, содержащая N открывающих скобок. Найдите её номер в лексикографическом порядке среди всех правильных скобочных последовательностей с таким же количеством открывающихся скобок, учитывая, что «(» < «)». Последовательности нумеруются с 0.

Формат входных данных

Во входном файле задана строка, являющаяся правильной скобочной последовательностью, $1 \leq N \leq 20$.

Формат выходных данных

Выведите единственное число — номер последовательности.

Примеры

brackets2num.in	brackets2num.out
((())())	3

Задача D. Светофоры

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

В подземелье M тоннелей и N перекрестков, каждый тоннель соединяет какие-то два перекрестка. Мышиный король решил поставить по светофору в каждом тоннеле перед каждым перекрестком. Напишите программу, которая посчитает, сколько светофоров должно быть установлено на каждом из перекрестков. Перекрестки пронумерованы числами от 1 до N .

Формат входных данных

Во входном файле записано два числа N и M ($0 < N \leq 100$, $0 \leq M \leq \frac{N(N-1)}{2}$). В следующих M строках записаны по два числа i и j ($1 \leq i, j \leq N$), которые означают, что перекрестки i и j соединены тоннелем. Гарантируется, что никакой тоннель не соединяет перекресток сам с собой, и не существует двух различных тоннелей, соединяющих одну и ту же пару вершин.

Формат выходных данных

В выходной файл вывести N чисел: k -е число означает количество светофоров на k -м перекрестке.

Примеры

lights.in	lights.out
7 10	3 3 2 2 5 2 3
5 1	
3 2	
7 1	
5 2	
7 4	
6 5	
6 4	
7 5	
2 1	
5 3	

Задача Е. От матрицы смежности к спискам смежности

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

Простой ориентированный граф задан матрицей смежности. Выведите его представление в виде списков смежности.

Формат входных данных

В первой строке файла находится число N — количество вершин графа ($1 \leq N \leq 100$). Во второй строке и далее — матрица смежности. Гарантируется, что граф не содержит петель.

Формат выходных данных

Выведите N строк — списки смежности графа. В i -й строке сначала выведите количество исходящих из i -й вершины рёбер, а затем — номера вершин, в которые эти рёбра идут, упорядоченные по возрастанию.

Примеры

mtoal.in	mtoal.out
5	1 3
0 0 1 0 0	2 1 3
1 0 1 0 0	1 5
0 0 0 0 1	2 1 2
1 1 0 0 0	2 1 2
1 1 0 0 0	

Задача F. Проверка на неориентированность

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

По матрице $N \times N$ из нулей и единиц определите, может ли данная матрица быть матрицей смежности простого неориентированного графа.

Формат входных данных

В первой строке число N ($1 \leq N \leq 100$), далее матрица — N строк по N чисел, каждое из которых равно 0 или 1.

Формат выходных данных

Выведите YES, если приведенная матрица может быть матрицей смежности простого неориентированного графа, иначе выведите NO.

Примеры

check.in	check.out
3 0 1 1 1 0 1 1 1 0	YES
3 0 1 0 1 0 1 1 1 0	NO
3 0 1 0 1 1 1 0 1 0	NO

Задача G. Полустепени вершин

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

Ориентированный граф задан матрицей смежности. Найдите полустепени захода и полустепени исхода всех вершин графа (т. е. количество входящих в нее и исходящих из нее ребер соответственно для каждой вершины).

Формат входных данных

N — число вершин в графе ($1 \leq N \leq 100$), затем матрица смежности: N строк по N чисел, каждое из которых равно 0 или 1.

Формат выходных данных

Выведите N пар чисел: для каждой вершины сначала полустепень захода и затем полустепень исхода.

Примеры

<code>half-degree.in</code>	<code>half-degree.out</code>
4	2 2
0 1 0 1	3 3
1 0 1 1	2 1
0 1 0 0	3 4
1 1 1 1	

Задача H. Истоки и стоки

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

Вершина ориентированного графа называется истоком, если в нее не входит ни одно ребро, и стоком, если из нее не выходит ни одного ребра.

Ориентированный граф задан матрицей смежности. Найдите все его вершины-истоки и все вершины-стоки.

Формат входных данных

N — число вершин в графе ($1 \leq N \leq 100$), затем матрица смежности — N строк по N чисел, каждое из которых равно 0 или 1.

Формат выходных данных

В первой строке выведите K — число истоков в графе, затем номера вершин, являющихся истоками в порядке возрастания. Во второй строке выведите информацию о стоках в том же формате.

Примеры

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

Задача I. Полный граф

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

Неориентированный граф называется полным, если любая пара его различных вершин соединена хотя бы одним ребром. Для заданного списком ребер графа проверьте, является ли он полным.

Формат входных данных

Программе на вход даются числа N и M , где N — число вершин ($1 \leq N \leq 100$) и M — число ребер ($1 \leq M \leq 10000$), а затем M пар чисел — ребра графа.

Формат выходных данных

Выведите YES, если граф является полным, и NO в противном случае.

Примеры

<code>complete.in</code>	<code>complete.out</code>
3 3	YES
1 2	
1 3	
2 3	