Задача A. Числа-перестановки

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

Пусть задано основание позиционной системы счисления $k \geq 2$. Назовем число (в котором, возможно есть ведущие нули) *числом-перестановкой*, если оно содержит ровно k цифр, и все цифры в нем различны.

Рассмотрим последовательность чисел a_1, a_2, $\ldots$, задаваемую следующим образом. Сначала запишем в нее все числа-перестановки для $k = 2$ в порядке возрастания, затем для $k = 3$, $k = 4$ и т.д. Например, $a_1 = 01_2 = 1_{10}$, $a_2 = 10_2 = 2_{10}$, $a_3 = 012_3 = 5_{10}$. Ваша задача — по числу n найти a_n.

Формат входного файла

Входной файл содержит целое число n ($1 \leq n \leq 10^9$).

Формат выходного файла

В выходной файл искомое число в десятичной системе счисления.

Примеры

numperm.in	numperm.out
1	1
3	5

Задача B. Многоугольник и отрезок

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

На плоскости задан многоугольник и отрезок. Найдите суммарную длину частей отрезка, которые лежат внутри или на границе многоугольника.

Формат входного файла

Первая строка входного файла содержит целое число n ($3 \le n \le 1000$) — число вершин многоугольника. Каждая из последующих n строк содержит по два целых числа x_i и y_i — координаты соответствующей вершины многоугольника. Вершины заданы в порядке обхода против часовой стрелки, многоугольник не содержит самопересечений и самокасаний.

Последняя строка входного файла содержит четыре целых числа: x_1, y_1, x_2, y_2 — координаты концов заданного отрезка. Все координаты во входном файле не превосходят 10^5 по абсолютному значению.

Формат выходного файла

В выходной файл выведите ответ на задачу не менее чем с 5 точными знаками после десятичной точки.

Примеры

polyseg.in	polyseg.out
4 1 0 0 1 -1 0 0 -1 -2 0 2 0	2.0

Задача C. Разбиения на множества

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

Разбиением множества X на k подмножеств называется набор множеств $A_1, A_2, \ldots, A_k$, где $A_i \cap A_j = \varnothing$ для $i \neq j$ и $A_1 \cup A_2 \cup \ldots \cup A_k = X$. Например, пара множеств $\{1, 4\}, \{2, 3\}$ является разбиением множества $\{1, 2, 3, 4\}$ на два подмножества.

Два разбиения считаются различными, если они отличаются хотя бы одним множеством. Разбиения, отличающиеся только порядком множеств, считаются одинаковыми.

Так, разбиения $\{1, 4\}, \{2, 3\}$ и $\{1, 3\}, \{2, 4\}$ являются различными, а разбиения $\{1, 4\}, \{2, 3\}$ и $\{2, 3\}, \{1, 4\}$ — нет.

Требуется перечислить все разбиения множества $\{1, 2, \ldots, n\}$ на k подмножеств.

Формат входного файла

Во входном файле заданы числа n и k ($1 \leq k \leq n \leq 10$).

Формат выходного файла

Выведите в выходной файл все разбиения множества $\{1, 2, \ldots, n\}$ на k подмножеств. Разбиения можно выводить в любом порядке. Внутри разбиения множества можно выводить в любом порядке. Внутри множества числа надо выводить в возрастающем порядке. Следуйте формату из примера.

Пример

partitions.in	partitions.out
4 2	1 2 3 4 2 1 3 4 3 1 2 4 4 1 2 3 1 2 3 4 1 3 2 4 2 3 1 4

Задача D. Электрические сети

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

Компания «Антарктические энергосети» владеет энергетической инфраструктурой Соединенных Штатов Антарктики. Энергетическая система Антарктики представляет собой n узлов (генерирующих или потребляющих энергию) и m соединяющих их высоковольтных линий. Каждая высоковольтная линия является направленной, она может передавать энергию от своего начала к своему концу.

Когда система исходно планировалась, предполагалось, что каждые два узла будут соединены высоковольтной линией в одном из направлений, таким образом планировалось провести $n(n-1)/2$ высоковольтных линии. Однако позже некоторые линии были удалены из проекта, всего было удалено k линий (таким образом, $m = n(n-1)/2 - k$). В результате этого изменения система стала удовлетворять следующему свойству: в системе отсутствуют направленные треугольники из высоковольтных линий (иначе говоря, ни для каких узлов u, v и w линии uv, vw и wu одновременно в системе не присутствуют).

Теперь, с целью экономии ресурсов, планируется демонтировать еще некоторые линии электропередачи. В результате демонтажа система должна соответствовать следующему требованию: никакие высоковольтные линии не образуют направленный цикл. При этом министр энергетической безопасности требует, чтобы было демонтировано не более k линий электропередачи. Вас наняли, чтобы составить план реорганизации. Выберите, какие линии электропередачи следует демонтировать. Заметьте, что вам не надо минимизировать это количество, надо только, чтобы оно не превышало k.

Формат входного файла

Первая строка входного файла содержит два целых числа: n и m ($1 \le n \le 100$, $0 \le m \le n(n-1)/2$). Следующие m строк описывают высоковольтные линии электропередачи, каждая линия описывается двумя числами — номерами начального и конечного узла (узлы пронумерованы от 1 до n). В текущей энергетической системе отсутствуют направленные треугольники.

Формат выходного файла

Первая строка выходного файла должна содержать число l — количество линий электропередачи, которые следует демонтировать. Это количество не должно превышать $n(n-1)/2 - m$. Вторая строка должна содержать l целых чисел — номера линий, которые следует удалить. После того, как удалены перечисленные линии, система не должна содержать направленных циклов. Линии пронумерованы от 1 до m в порядке, в котором они заданы во входном файле.

Если решения не существует, выведите -1 на первой строке выходного файла.

Пример

electricity.in	electricity.out
4 4 1 2 2 3 3 4 4 1	1 1