Задача SMT в теории конечных частично упорядоченных множеств.

Задача выполнимости формул в теориях (satisfiability modulo theories, SMT) — это задача проверки выполнимости формул с учётом лежащих в их основе теорий. Примерами таких теорий для SMT-формул являются теории целых и вещественных чисел, теории списков, массивов, битовых векторов и т.п.

Теория упорядоченных множеств — это раздел математики, который исследует интуитивное понятие порядка с помощью бинарных отношений. Она обеспечивает формальную основу для описания таких утверждений, как «это меньше, чем это» или «это предшествует тому». Эта теория особенно полезна для описания и нахождения кратких решений проблем теории графов, анализа программ и запросов к базам данных.

Задача.

Дан частичный порядок. Необходимо установить различные его свойства. Важно, что **задача должна быть решена через использование API (любого существующего) SMT-решателя**, а не реализацией стандартных графовых алгоритмов.

Суть задачи состоит в разработке способа эффективного сведения теории частичных порядков к теориям, реализованным в существующих SMT-решателях (например, Z3). К примеру, можно использовать теории линейной целочисленной арифметики и неинтерпретированных функций.

Ввод

Первая строка содержит два числа: число n ($0 < n <= 10^2$), число элементов частично упорядоченного множества, и число k ($0 <= k <= 10^4$). Следующие k строк содержат по паре чисел a и b ($0 <= a$, $b < n$), которые соответствуют номерам элементов частично упорядоченного множества. Множество таких пар образует некоторое отношение на этом множестве.

Вывод

Необходимо построить рефлексивно-транзитивное замыкание этого отношения и проверить, удовлетворяет ли оно свойству антисимметричности, то есть является ли оно частичным порядком. Если отношение не является частичным порядком, то нужно вывести номера элементов, которые нарушают отношение антисимметричности. Если отношение является частичным порядком, то для него нужно проверить следующие свойства.

1. Проверить существование наибольшего элемента. Вывести номер элемента, если такой есть, иначе вывести *greatest not exist*.

2. Проверить существование наименьшего элемента. Вывести номер элемента, если такой есть, иначе вывести *least not exist*.
3. Вывести множество максимальных элементов.
4. Вывести множество минимальных элементов.
5. Проверить, является ли отношение частичного порядка линейным порядком. Вывести 1, если является, иначе 0.
6. Вывести построчно пары элементов, которые находятся в отношении транзитивного сокращения отношения строгого частичного порядка.

Пример ввода 1	Пример вывода 1
6 8 0 1 0 2 0 4 3 5 1 3 2 3 3 4 1 5	greatest not exist 0 4 5 0 0 0 1 0 2 2 3 1 3 3 4 3 5

Пример ввода 2	Пример вывода 2
5 4 0 1 2 3 3 4 4 2	2 3 4

Пример ввода 3	Пример вывода 3
6 6 0 2 1 3 2 4 3 5 2 1 5 4	4 0 4 0 1 0 2 2 1 1 3 3 5 5 4