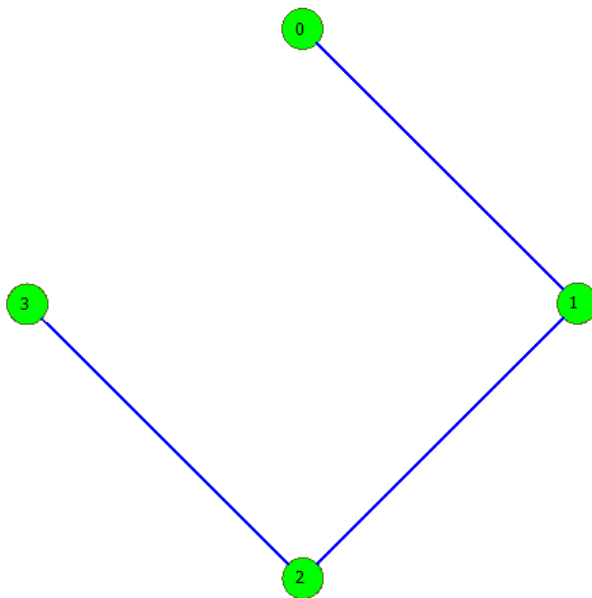


## Программная реализация алгоритмов

- Выбранный алгоритм — нахождение шарниров графа на основе глубинных номеров вершин графа.
- Необходимая теория
  - Шарниром графа называется вершина, при удалении которой количество компонент связности возрастает
  - Компонента связности графа — максимальный (по включению) связный подграф графа. Например, в данном графе одна компонента связности.



- Шарнирами в этом графе являются вершины 1 и 2.
- Описание алгоритма
  - на ввод подаётся неориентированный ненагруженный граф, который преобразовывается программой ГРАФОИД в матрицу смежности
  - из графа удаляется одна вершина, то есть в матрице смежности обнуляются соответствующие строка и столбец
  - происходит обход графа в глубину

- алгоритм:
  - перебираем все исходящие из рассматриваемой вершины рёбра
  - если ребро ведет в вершину, которая не была рассмотрена ранее, то запускаем алгоритм от этой нерассмотренной вершины, а после возвращаемся и продолжаем перебирать рёбра
  - возврат происходит в том случае, если в рассматриваемой вершине не осталось рёбер, которые ведут в нерассмотренную вершину
- каждый обход, кроме самого первого, начинается с нулевой вершины (первый обход начинается с первой), создается вектор целых чисел, в который добавляется порядковый номер вершины в обходе (индекс в векторе соответствует номеру вершины в графе), если вершина не попала в обход, добавляется число  $-1$ .
- если после обхода нашлись вершины, не попавшие в обход (то есть в векторе обхода число под индексом номера вершины равно  $-1$ ), их номера добавляются в вектор шарниров
  - сие действие выполняется для каждой вершины графа
  - в текстовое окно выводятся номера вершин, являющихся шарнирами графа, то есть выводится вектор шарниров, если вектор шарниров пустой, выводится сообщение о том, что шарниров в графе не найдено.
- Инструкция: ввести граф в программе ГРАФОИД, далее запустить исполняемый файл *hinges*, результатом работы программы является сообщение в текстовом окне, где будут указаны вершины, являющиеся шарнирами.

- Оценка вычислительной сложности алгоритма: пусть  $n$  — количество вершин в графе, сложность обхода в глубину  $O(n)$ , сам обход в глубину выполняется для каждой вершины, то есть  $n$  раз, таким образом, общая оценка вычислительной сложности алгоритма —  $O(n^2)$ .

Скриншоты выполненных тестовых примеров:

