

# Лабораторная работа № 9 по курсу дискретного анализа: Графы

## Условие

Задан неориентированный граф, состоящий из  $n$  вершин и  $m$  ребер. Вершины пронумерованы целыми числами от 1 до  $n$ . Необходимо вывести все компоненты связности данного графа. Каждую компоненту связности нужно выводить в отдельной строке, в виде списка номеров вершин через пробел. Строки при выводе должны быть отсортированы по минимальному номеру вершины в компоненте, числа в одной строке также должны быть отсортированы.

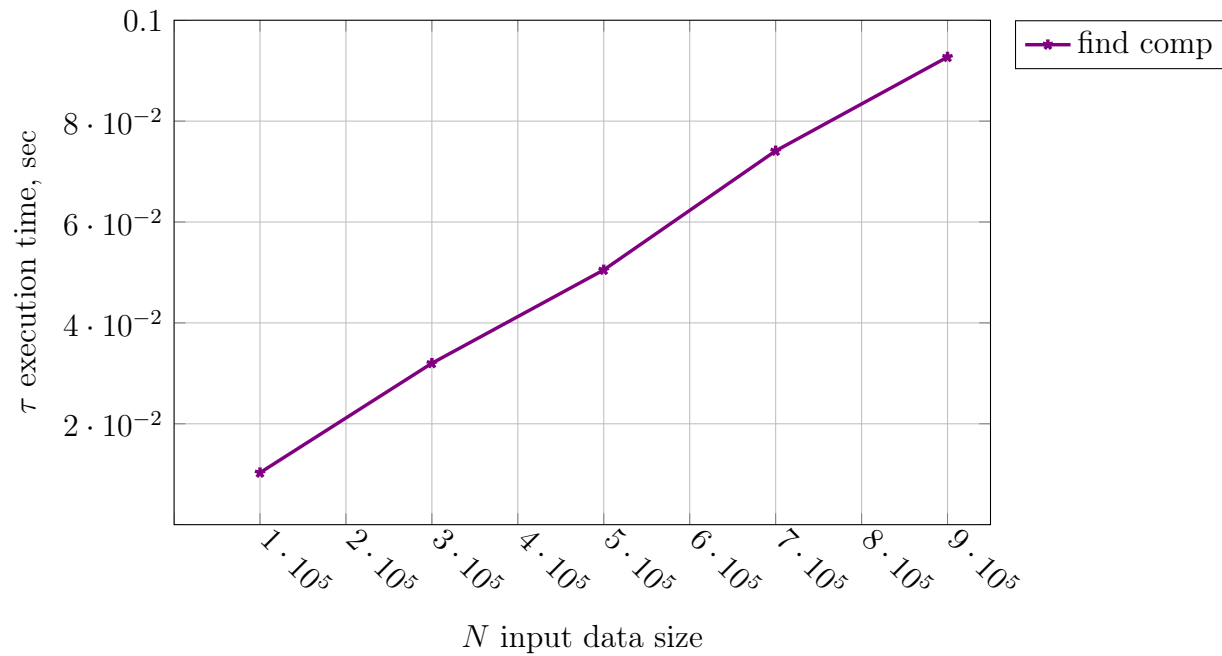
## Метод решения

Компонента связности — набор вершин графа, между любой парой которых существует путь. В таком случае, если запустить обход из любой вершины, мы обойдём всю компоненту связности, а во все остальные вершины мы не попадём. Отлично, тогда пройдем по всем вершинам, если вершина лежит в какой-либо компоненте уже, то игнорируем её, если нет, то запускаем обход графа из неё и все вершины, которые посетим заносим в новую компоненту связности. Ответ сортируем как требует условие!

## Описание программы

Нам подойдёт любой обход графа, ну хорошо напишем bfs за  $O(|E| + |V|)$ , чтобы рекурсивный dfs не тратил моё место на стеке. Сам граф будем хранить в виде списка смежности, хотя можно было бы и матрицу инцидентности, но она потребовала бы больше памяти, и не дала никаких преимуществ.

## Тест производительности



## Выводы

В ходе выполнения лабораторной работы была решена задача по поиску компонент связности графа. Делается это крайне тривиально и не требует особых знаний алгоритмов. Умение написать обход нам дали ещё в школе в 7 классе, когда мы изучали excel таблицы. Сложность временную и асимптотическую не сложно оценить, так как мы всего-навсего выполняем обход графа. Как это видно из графика в предыдущем пункте временная сложность  $O(|V| + |E|)$ , где  $V$  — количество вершин, а  $E$  — рёбер графа.