

## Задание по курсу “Теория графов и ее приложения” v.2025

### Часть 2. Командный проект

Задание основано на исследовании структуры социальных графов и связанных прикладных задач (вычисления расстояний).

Задание состоит из нескольких частей:

1. Анализ структуры сети (5 баллов);
2. Реализация базового алгоритма для вычисления (оценки) расстояний между вершинами (10 баллов);
3. Реализация модификации базового алгоритма (15 баллов).

**Необходимое условие** выполнения зачёта – выполнения 1 и 2 части задания.

#### Данные.

Датасеты будут размещены на Y.disk [https://disk.yandex.ru/d/jKL\\_5xzgot6JlAQ](https://disk.yandex.ru/d/jKL_5xzgot6JlAQ).

#### 1. Анализ структуры сети

A. Для каждой из сетей определить следующие характеристики:

1. Число вершин, число рёбер, плотность (отношение числа рёбер к максимально возможному числу рёбер), число компонент слабой связности, долю вершин в максимальной по мощности компоненте слабой связности. Для ориентированных графов определить число компонент сильной связности и долю вершин графа в наибольшей компоненте сильной связности компоненте.
2. Для наибольшей компоненты слабой связности оценить значения диаметра сети, 90 процентиля расстояния (геодезического) между вершинами графа. Оценку провести на основании:
  - a. Двойного прохода BFS (the double sweep): Для случайно выбранного узла  $r$  найти максимально удаленный узел  $a$ , а затем найти узел  $b$ , максимально удаленный от  $a$ . За диаметр принять эксцентриситет вершины  $ecc(a) = d(a, b)$ .
  - b. вычисления расстояний между 500 (1000) случайно выбранными вершинами из наибольшей компоненты слабой связности;
  - c. вычисления расстояний по подграфу "снежный ком" (snowball sample), построенного по следующему принципу: выбирается небольшое начальное множество смежных вершин (2 или 3), затем в граф добавляются все их соседи, затем соседи соседей и т.д., пока число вершин в подграфе не станет равным (примерно) заданному значению (например, 500 или 1000).
3. Число треугольников (полных подграфов на 3 вершинах), средний и глобальный кластерные коэффициенты.
4. Для наибольшей компоненты слабой связности вычислить средний кластерный коэффициент сети  $\bar{C}l = \frac{1}{|V|} \sum_{u \in G} Cl_u$ , где
$$Cl_u = \begin{cases} \frac{2L_u}{|\Gamma(u)| \cdot |\Gamma(u)-1|}, & |\Gamma(u)| \geq 2, \\ 0, & \text{иначе.} \end{cases}$$
где  $\Gamma(u)$  – множество соседей вершины,  $|\Gamma(u)| = k_u$  – степень вершины,  $L_u$  --- число ребер между соседями.
5. Найти минимальную, максимальную, среднюю степень узла в графе. Построить функцию вероятности для степеней вершин сети в обычных и log-log шкалах.

- В. Для каждой из сетей исследовать, как меняется доля вершин в наибольшей компоненте слабой связности, если из сети удаляется*
- 1. случайным образом  $x\%$  узлов;*
  - 2. удаляется  $x\%$  узлов наибольшей степени.*
- С. Сравнить результаты, полученные для каждой сети, описать полученные сходства/различия.*

Графы считать неориентированными с единичными весами рёбер, кроме поиска компонент сильной связности для орграфов.

## **2. Вычисление расстояний между вершинами сети**

Вычисление кратчайших расстояний между вершинами – один из фундаментальных алгоритмов теории графов – широко используется при вычислении различных метрик вершин социальных сетей (например, метрик центральности по близости, по посредничеству и т.д.). Однако, из-за размера таких сетей точное вычисление расстояний зачастую невозможно, и возникает необходимость в приближенных алгоритмах.

Реализовать один из вариантов алгоритма вычисления (оценки) расстояний в сетях, изложенный в [1].

Каждая команда должна реализовать базовый алгоритм для оценки расстояний (Landmarks-Basic), а также одну из модификаций алгоритма на основе построения кратчайшего дерева путей (shortest path tree (SPT)) и способа оценки расстояний между парой вершин Landmarks-LCA, Landmarks-SC, Landmarks-BFS.

Исследовать работу алгоритмов: точность вычисления расстояний, скорость работы (алгоритма в целом и отдельных этапов/операций) и т.п.

Влияет ли на точность работы алгоритма число и способ выбора вершин-ориентиров (landmarks) (случайный, с наибольшими степенями, с наилучшим покрытием сети).

Алгоритм для реализации для каждой команды будет указан в таблице.

1. Tretyakov, K., Armas-Cervantes, A., García-Bañuelos, L., Vilo, J. & Dumas, M. Fast fully dynamic landmark-based estimation of shortest path distances in very large graphs. Proc 20th Acm Int Conf Information Knowl Management - Cikm '11 1785–1794 (2011) doi:10.1145/2063576.2063834.
2. Potamias, M., Bonchi, F., Castillo, C. & Gionis, A. Fast shortest path distance estimation in large networks. Proceeding 18th Acm Conf Information Knowl Management - Cikm '09 867–876 (2009) doi:10.1145/1645953.1646063.

В качестве решения будет засчитано следующее:

- Результаты работы алгоритмов на графах малой размерности (тестовые примеры).
- Отчет-презентация, содержащий достаточно информации по использованным инструментам и алгоритмам для решения задачи, а также результаты исследования и выводам.
- Продумать интерфейс так, чтобы можно было демонстрировать работу алгоритма (вывод результатов, вычисление расстояний между некоторой парой вершин, вычисление локального кластерного коэффициента и т.п., симуляция исследований из п. В.
- Результаты исследования (сравнений) работ алгоритмов для п. 2, аналогичные изложенным в статье [1]. Время работы алгоритмов должно быть сравнимым для графов схожей размерности.

## Дополнительные условия

- При выполнении задания запрещено пользоваться готовыми решениями.
- Готовые решения могут использоваться для импорта/экспорта данных, построения графиков.

## Общие требования к выполнению заданий

1. Допускается работа в группах не более 3-х человек.
  - a. После 10.05.2025 студенты, не отметившиеся в файле в составе той или иной команды, считаются работающими самостоятельно.
  - b. В одну команду могут входить студенты из разных учебных групп.
  - c. Все особые ситуации (например, распад команды из-за непреодолимых разногласий) обсуждаются отдельно с преподавателем до даты защиты проекта (зачёта). Если вы не уверены, является ли ваша ситуация особой, всё равно напишите преподавателям, например, в чате в Telegram.
2. По результатам выполнения проекта каждая команда должна подготовить отчет-презентацию и сделать доклад.
3. Каждая команда должна опубликовать свое решение в репозитории в ветку с номером команды Team\_nn.

Ссылка на GitHub: [https://github.com/answerIII/Graph\\_Theory\\_LeetCode25](https://github.com/answerIII/Graph_Theory_LeetCode25)

Номер команды указывается согласно порядковому номеру в файле <https://clck.ru/3GyqXv>.
4. При выставлении итоговых баллов учитывается качество и количество проделанной работы (использованные структуры данных, наличие/отсутствие сравнений работы алгоритмов с готовыми решениями, количество обработанных датасетов и т.п.), история коммитов, ответы на вопросы по проекту.
5. При выполнении работы студенты следуют Кодексу универсанта, в частности соблюдают нормы научной этики, уважают права интеллектуальной собственности и не используют недобросовестных методов при прохождении аттестации (п. 4, 6 и 7).
6. Преподаватели оставляют за собой право выдать дополнительное задание, если возникли сомнения в самостоятельности выполнения студентом (группой) задания.

Сроки выполнения задания

**25.05.2025**

Воронкова Ева Боруховна

e.voronkova@spbu.ru

Вольф Дмитрий Александрович

answer.iii@mail.ru