



Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное
учреждение высшего образования
«Московский государственный технический университет имени Н.
Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н. Э. Баумана)

ФАКУЛЬТЕТ «Информатика и системы управления»

КАФЕДРА «Программное обеспечение ЭВМ и информационные технологии»

Отчет по практикуму №2 по курсу «Архитектура ЭВМ»

Тема Обработка и визуализация графов

Студент Калашков П. А.

Группа ИУ7-56Б

Оценка (баллы) _____

Преподаватель Ибрагимов С. В. В.

Москва — 2022 г.

Задание 1

Практикум 2 выполнялся по варианту 4.

Задание: Выполнить визуализацию неориентированного графа, представленного в формате tsv. Каждая строка файла представляет собой описание ребра, состоящее из трех чисел (Вершина,Вершина,Вес) или двух чисел (Вершина,Вершина). Во втором случае вес ребра принимается равным 1.

Для выполнения данного задания были изменены файлы с кодом примера.

В листинге 1 представлено создание и передача данных из буфера.

Листинг 1 – Создание и передача данных из буфера

```
1 unsigned int* host2gpc_ext_buffer[LNH_GROUPS_COUNT][
    LNH_MAX_CORES_IN_GROUP];
2 unsigned int edge_count = get_edge_count(SRC_FILE);
3 unsigned int messages_count = 0;
4 unsigned int u, v, w;
5
6 __foreach_core(group, core)
7 {
8     host2gpc_ext_buffer[group][core] = (unsigned int*)lnh_inst.gpc[
        group][core]—>external_memory_create_buffer(2 * 3 * sizeof(
            int) * edge_count); //2*3*sizeof(int)*edge_count);
9
10    std::ifstream fin(SRC_FILE);
11    for (int edge = 0; edge < edge_count; ++edge)
12    {
13        fin >> u >> v >> w;
14        EDGE(u, v, w);
15        EDGE(v, u, w);
16        messages_count += 2;
17    }
18    fin.close();
19
20
21    lnh_inst.gpc[group][core]—>external_memory_sync_to_device(0, 3 *
        sizeof(unsigned int)*messages_count);
22 }
```

Для получения количества рёбер используется функция *get_edge_count*, код которой представлен в листинге 2.

Листинг 2 – Создание и передача данных из буфера

```
1 int get_edge_count(std::string filename)
2 {
3     std::ifstream fin(filename);
4     printf("%d\n", fin.is_open());
5     int a1, a2, a3;
6     int count = 0;
7     while (fin >> a1 >> a2 >> a3)
8     ++count;
9     fin.close();
10    return count;
11 }
```

После заполнения графа данными, с помощью сервера boken визуализируется граф (рис. 1).

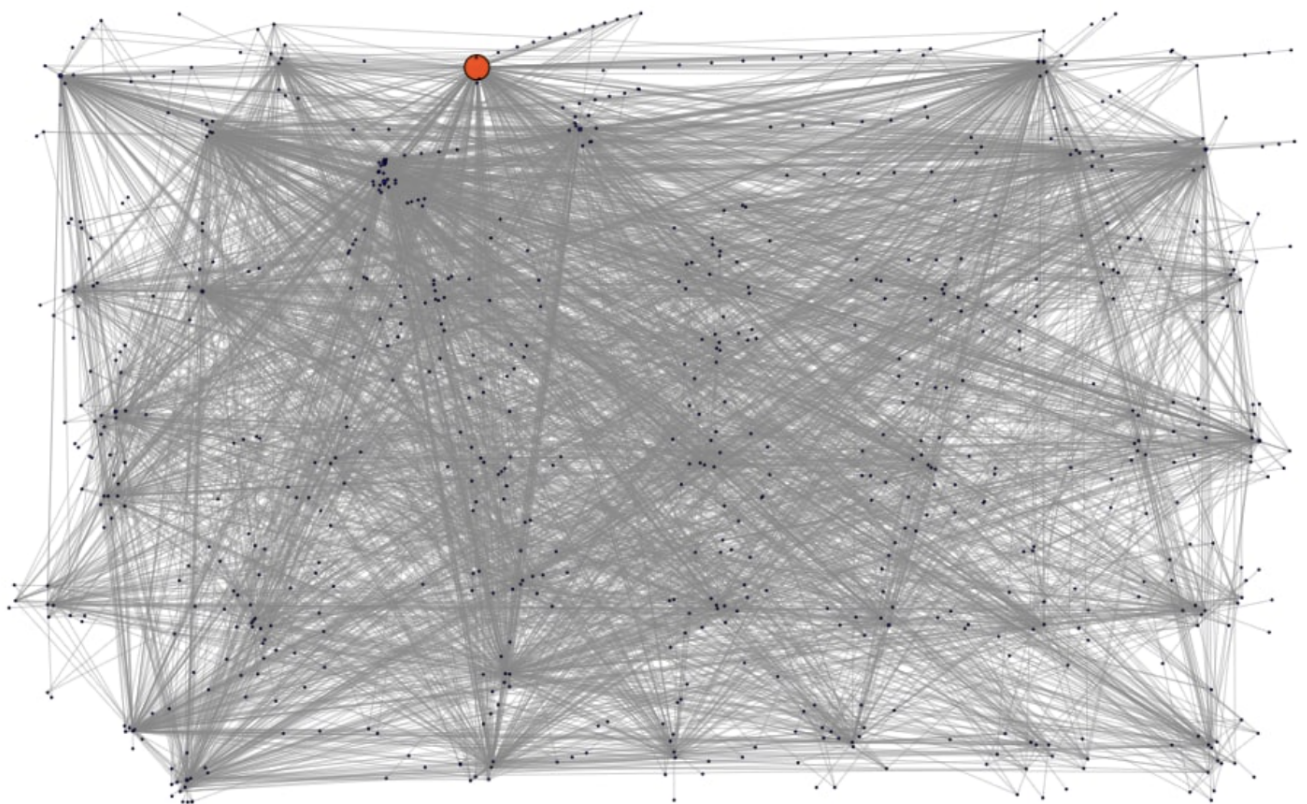


Рисунок 1 – Визуализированный граф по варианту