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 лабораторной работе №7

по курсу «Анализ Алгоритмов»

на тему: «Графовые модели»

Студент ИУ7-55Б (Группа) _____ (Подпись, дата) Половинкин И. Д. (И. О. Фамилия)

Преподаватель _____ (Подпись, дата) Волкова Л. Л. (И. О. Фамилия)

2025 г.

СОДЕРЖАНИЕ

ВВЕДЕНИЕ

Цель данной лабораторной работы: выполнить построение 4 графов (информационный граф, информационная история, граф управления, операционная история).

Для её достижения были поставлены следующие задачи:

— дать определения графовых моделей;

— выделить законченный фрагмент кода на 15+ значащих строк кода;

— выполнить построение 4 графов;

— сделать вывод о применимости графовых моделей к задаче анализа программного кода.

1 Код, выбранного алгоритма

В листинге 1.1 представлен код, выбранного алгоритма.

Листинг 1.1 – Алгоритм обработки файла в мультипоточном режиме

```
double multi_threads(
    std::vector<std::string> urls, // 1
    int col_threads,               // 2
    int tests)                     // 3
    {

    std::thread threads[128]; // 4

    clock_t begin = clock(); // 5
    for (int test = 0; test < tests; ++test) { // 6

        int index = 0; // 7
        while (index < urls.size()) { // 8

            int col_worked_threads = 0; // 9
            for (int i = 0; i < col_threads; ++i) { // 10

                if (index >= urls.size()) // 11
                    break; // 12

                threads[i] = std::thread(thread_func, &urls,
                   index); // 13
                index++; // 14
                col_worked_threads++; // 15
            }

            for (int i = 0; i < col_worked_threads; ++i) // 16
                threads[i].join(); // 17
        }
    }

    clock_t end = clock(); // 18

    return double(end - begin) / CLOCKS_PER_SEC / tests; // 19
}
```

2 Информационный граф

Информационный граф —модель, в которой вершины: операторы, дуги — информационные отношения. Информационное отношение —это отношение по передаче данных. Информационный граф представлен на рисунке 2.1.

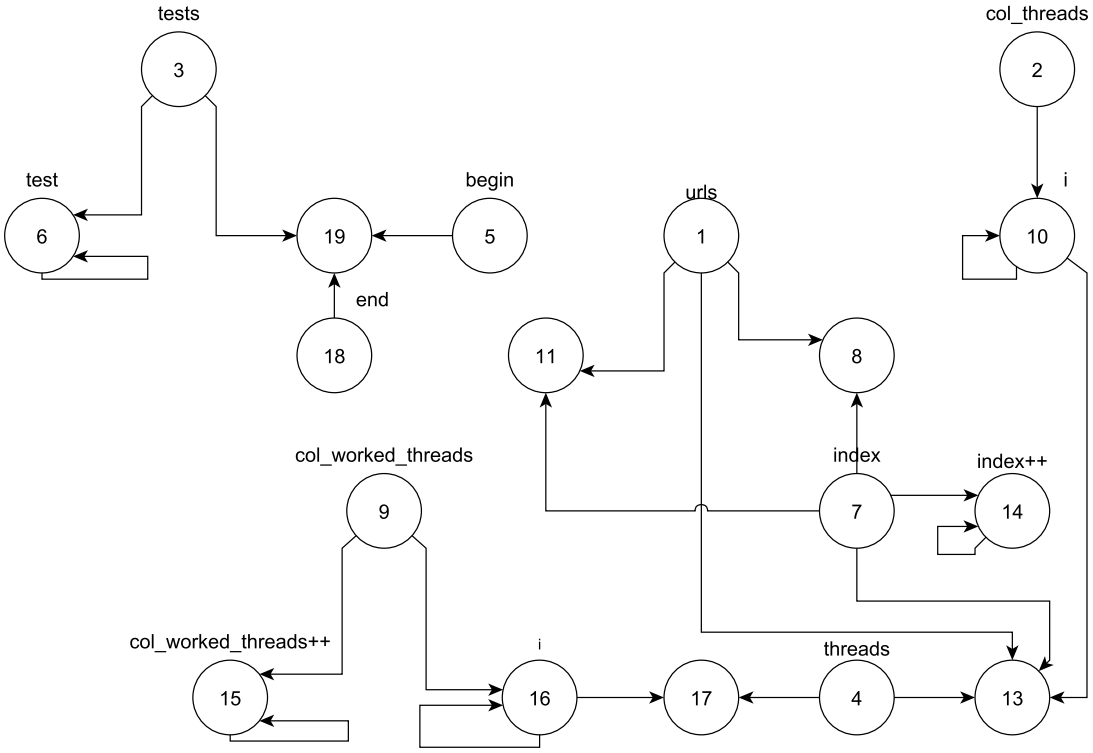

Рисунок 2.1 – Информационный граф

3 Граф управления

Граф управления —модель, в которой вершины: команды, дуги — операционные отношения. Операционные отношение —это отношение по передаче управления. Граф управления представлен на рисунке 3.1.

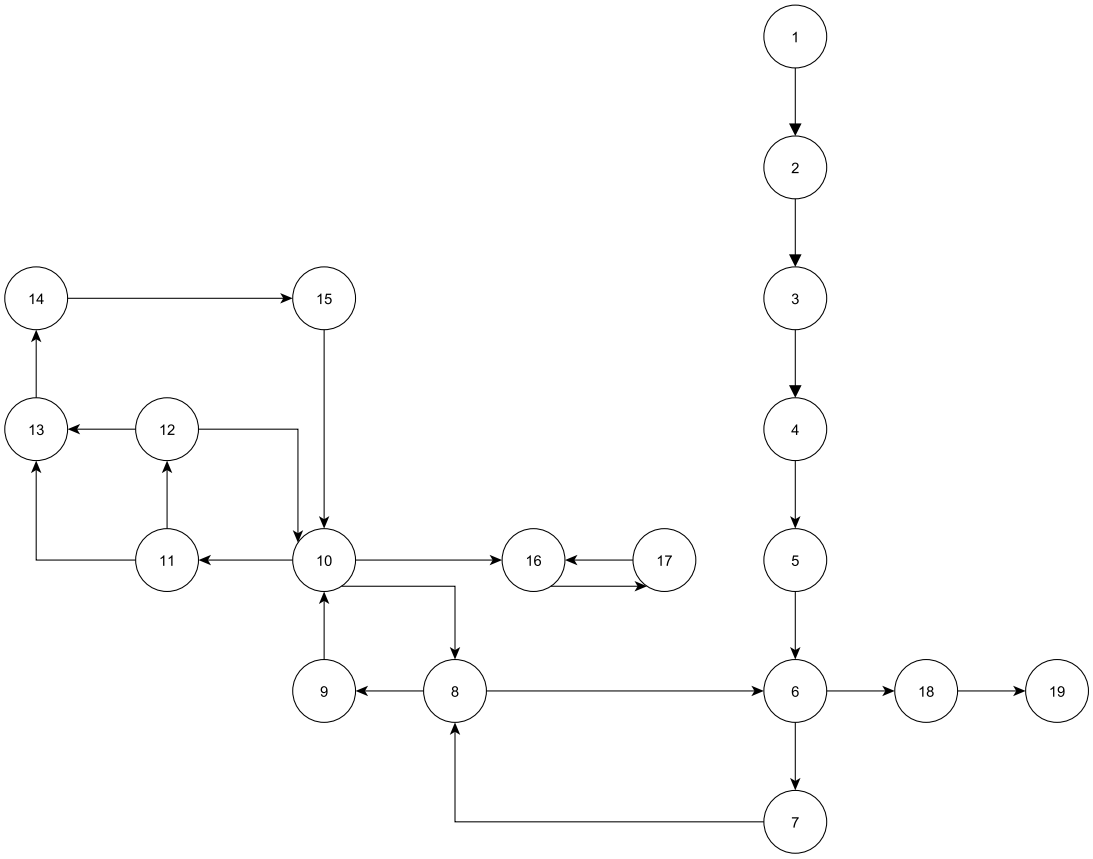

Рисунок 3.1 – Граф управления

4 Информационная история

Информационная история —модель, в которой вершины: срабатывания операторов, дуги —информационные отношения. Информационная история представлена на рисунке 4.1.

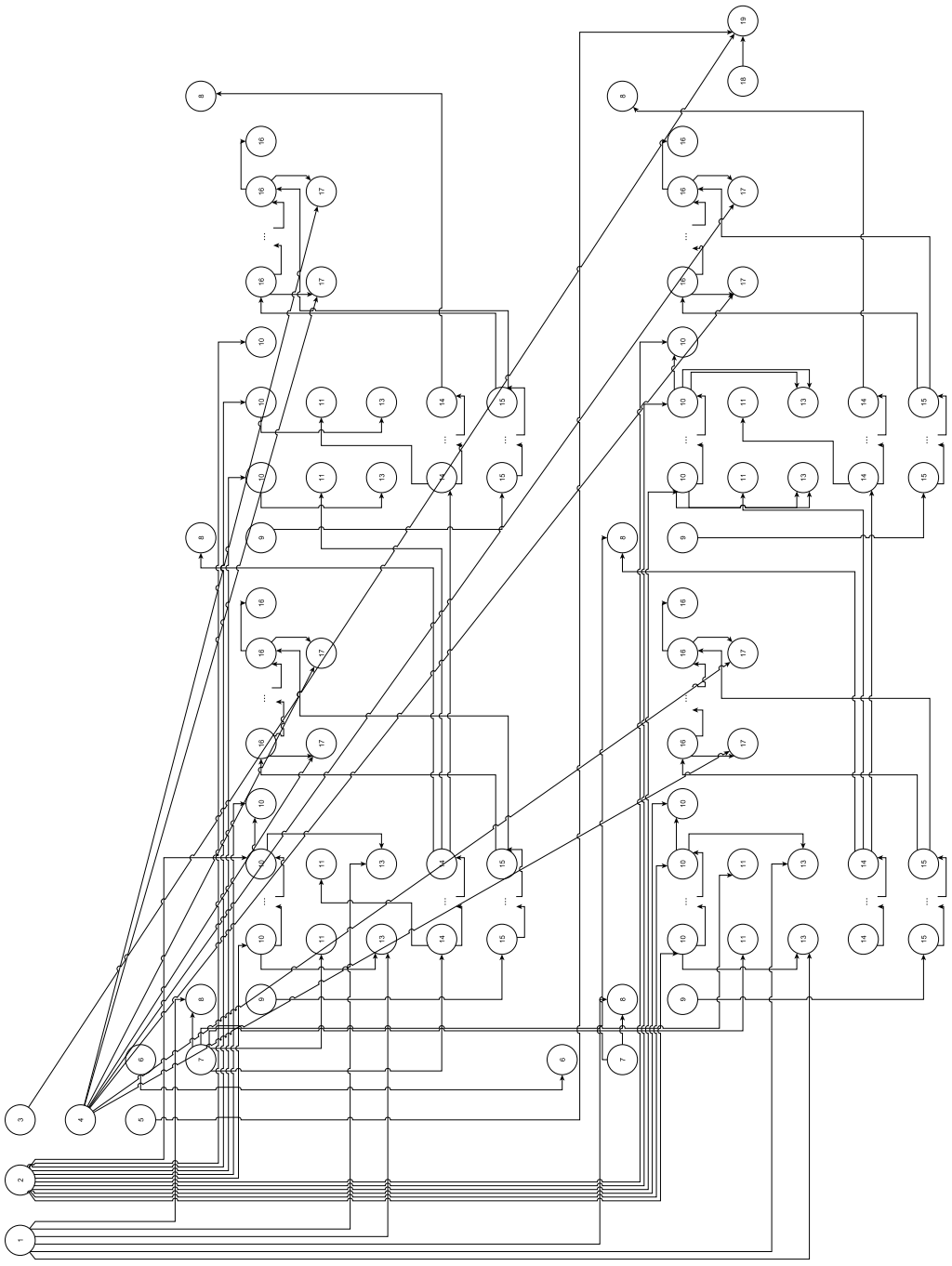

Рисунок 4.1 – Информационная история

5 Операционная история

Операционная история —модель, в которой вершины: срабатывания операторов, дуги —операционные отношения. Операционная история представлена на рисунке 5.1.

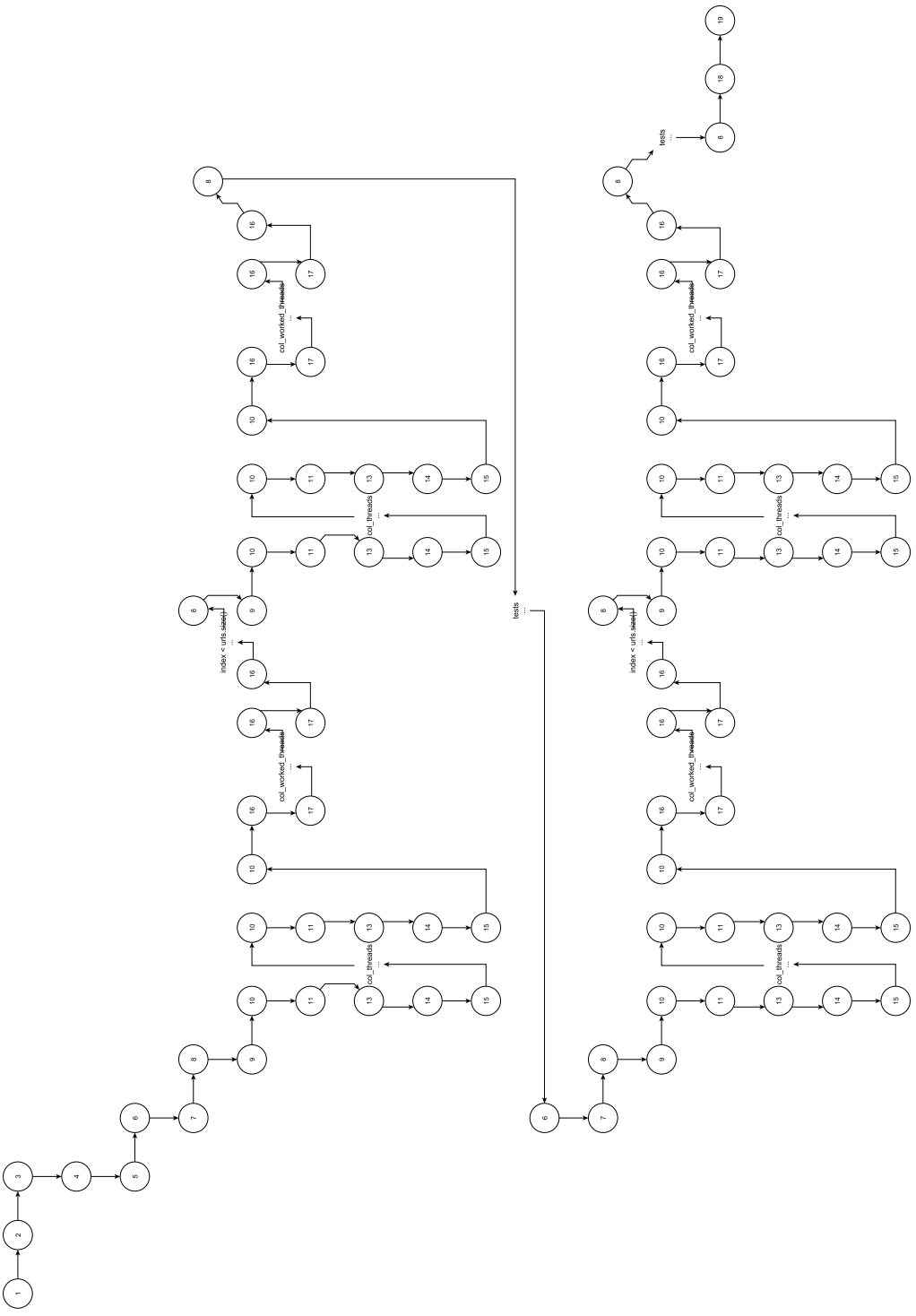

Рисунок 5.1 – Операционная история

ЗАКЛЮЧЕНИЕ

Графовые модели предоставляют мощные инструменты для анализа программного кода, которые помогают структурировать информацию, выявлять взаимосвязи и зависимости, а также находить проблемы и уязвимости. Их применимость охватывает множество аспектов, включая статический и динамический анализ, оптимизацию производительности, упрощение структуры кода и понимание изменений состояния. С учетом всё более сложных и больших кодовых баз, графовые модели становятся незаменимыми в области программной инженерии и анализа кода.

Цель данной лабораторной работы была достигнута.

Также были решены следующие задачи:

— даны определения графовых моделей;

— выделен законченный фрагмент кода на 15+ значащих строк кода;

— выполнено построение 4 графов;

— сделан вывод о применимости графовых моделей к задаче анализа программного кода.