



**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»

Лабораторная работа №7 по дисциплине «Анализ алгоритмов»

Тема Графовые модели

Студент Тузов Даниил Александрович

Группа ИУ7-52Б

Преподаватель Волкова Лилия Леонидовна

Москва, 2024 г.

СОДЕРЖАНИЕ

ВВЕДЕНИЕ	3
1 Исходный код, выбранного алгоритма	4
2 Информационный граф	5
3 Граф управления	6
4 Информационная история	7
5 Операционная история	8
ЗАКЛЮЧЕНИЕ	9

ВВЕДЕНИЕ

В 7 лабораторной работе рассматриваются графовые модели.

Целью работы является описание 4 графовых моделей на примере выбранного фрагмента кода. Для достижения поставленной цели необходимо решить следующие задачи:

- выбрать фрагмент кода, для которого будут строиться графовые модели;
- построить информационный граф;
- построить информационную историю;
- построить граф управления;
- построить операционную историю.

1 Исходный код, выбранного алгоритма

Были поставлены следующие требования к фрагменту кода:

- 15 или более значащих строк кода (не пробелы, не комментарии, не фигурные скобки);
- в коде есть хотя бы 2 цикла, среди которых один вложен в другой.

По этой причине был выбран код основной функции муравьиного алгоритма. Код представлен в листинге 1.1.

Листинг 1.1 — Муравьиный алгоритм

```
1 def ant_alg(mat, alpha, ro, t_max):
2     n, q = len(mat), calc_q(mat)
3     pheromone = init_pheromone(n)
4     attract = init_attract(mat)
5     min_path_len = -1
6     best_path = list()
7     for t in range(t_max):
8         memory = init_memory(n)
9         for ant in range(n):
10             while len(memory[ant]) != n:
11                 p = calc_p(pheromone, attract, memory[ant], n, alpha)
12                 memory[ant].append(calc_next(p))
13                 path_len = calc_length(mat, memory[ant])
14                 if min_path_len == -1 or min_path_len > path_len:
15                     min_path_len = path_len
16                     best_path = memory[ant]
17             pheromone = update_pheromone(mat, memory, pheromone, q, ro)
18     for i in range(len(best_path)):
19         best_path[i] += 1
20     return best_path, min_path_len
```

2 Информационный граф

Информационный граф (ИГ) — это такая модель программы, в которой вершины — это команды, операторы или строки кода, а дуги — информационное отношение.

Информационное отношение — это отношение по передаче данных.

Информационный граф для муравьиного алгоритма представлен на рисунке 2.1.

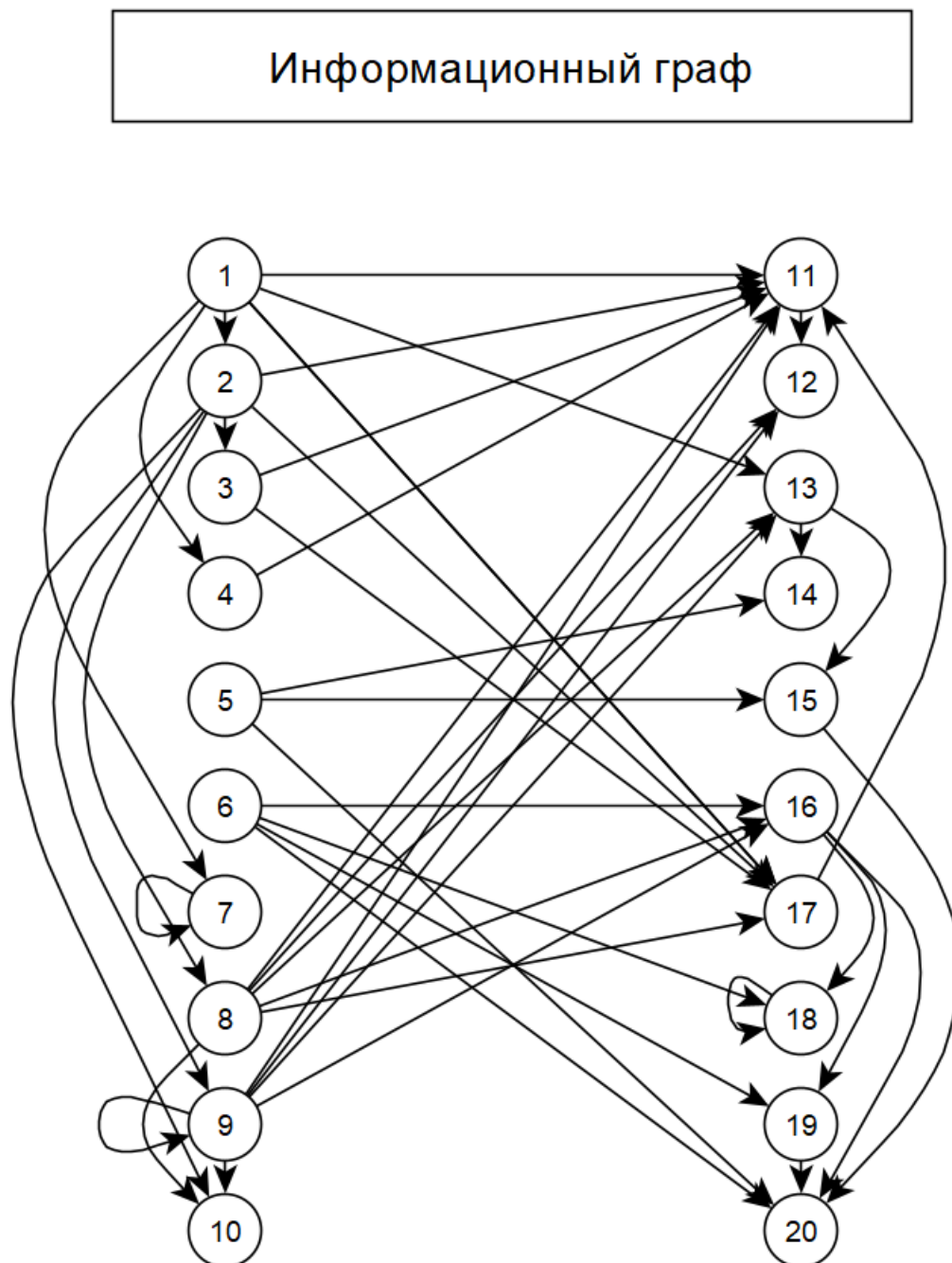


Рисунок 2.1 — Информационный граф

3 Граф управления

Граф управления (ГУ) — это такая модель программы, в которой вершины — это команды, операторы или строки кода, а дуги — операционное отношение.

Операционное отношение — это отношение по передаче управления.

Граф управления для муравьиного алгоритма представлен на рисунке 3.1.

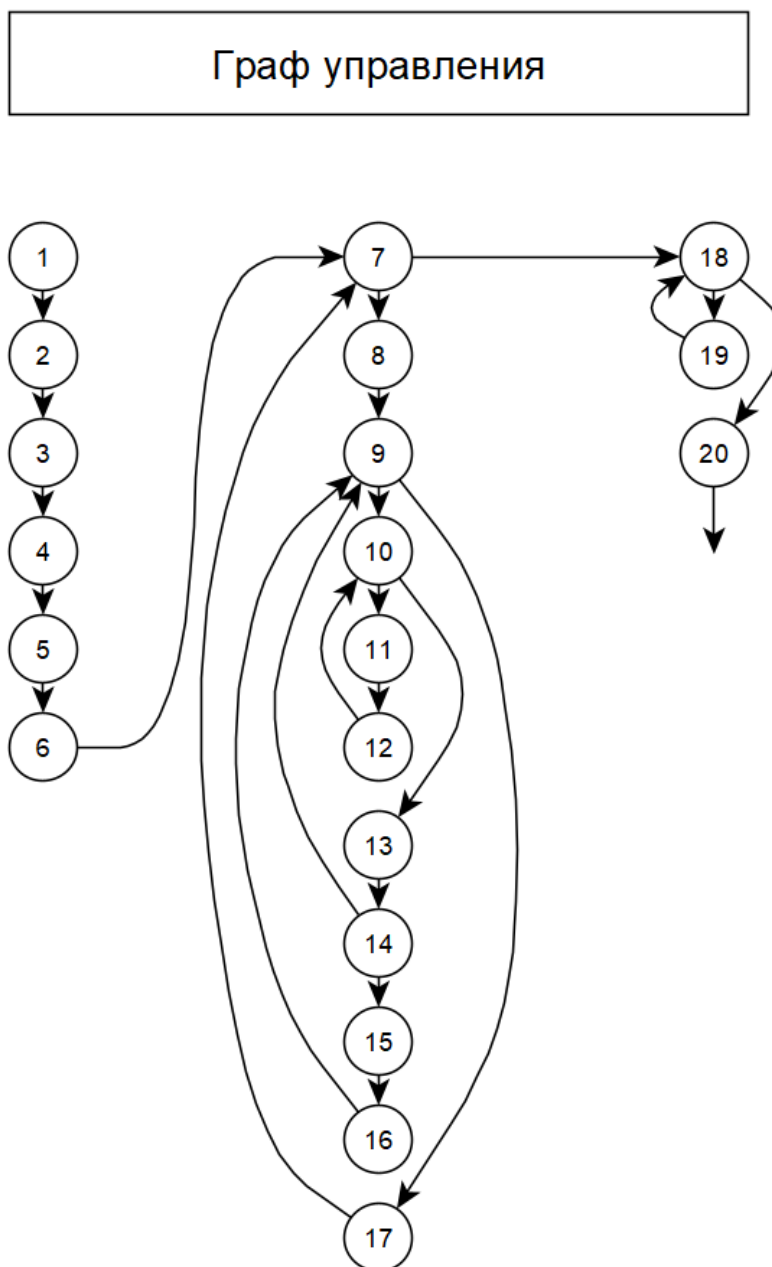


Рисунок 3.1 — Граф управления

4 Информационная история

Информационная история при анализе алгоритма — это информационное отношение между вершинами графа, отражающее, как одна вершина использует в качестве аргумента значение, полученное в другой.

Информационная история для муравьиного алгоритма представлен на рисунке 4.1.

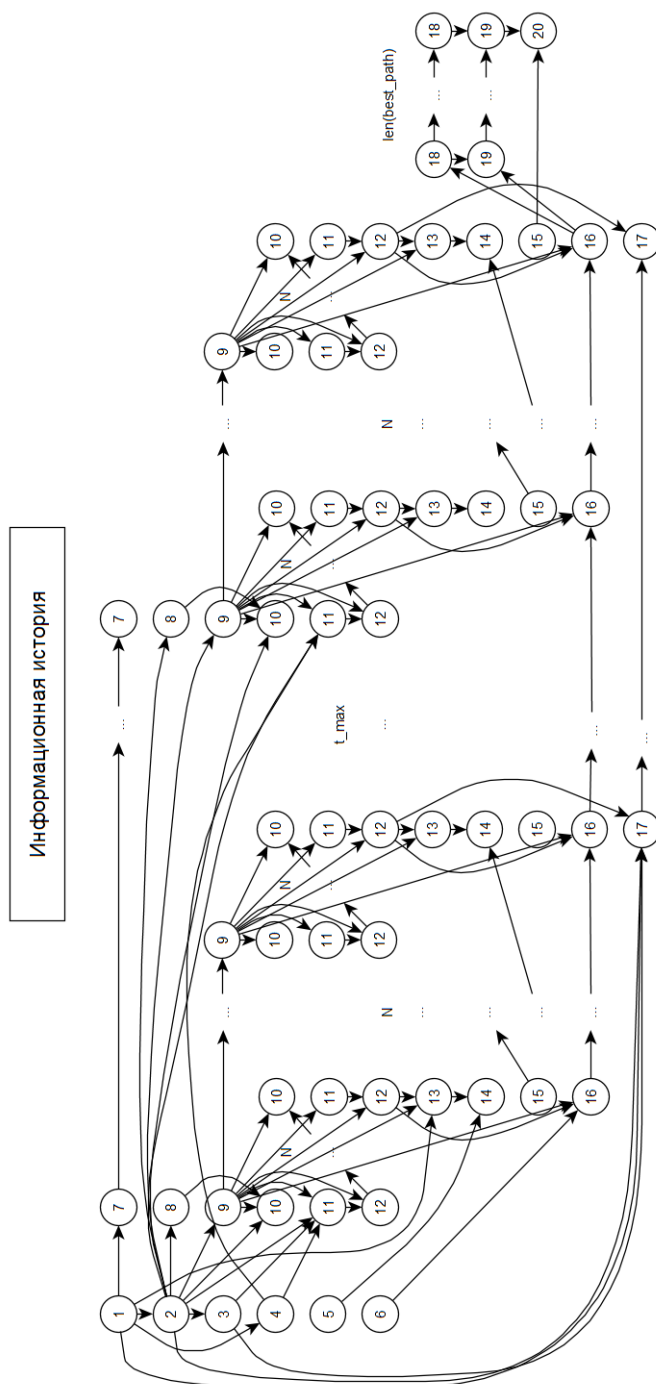


Рисунок 4.1 — Информационная история

5 Операционная история

Операционная история — это операционное отношение между вершинами, означающее, что одна вершина может быть выполнена сразу после другой. Является строго линейной.

Операционная история для муравьиного алгоритма представлен на рисунке 5.1.

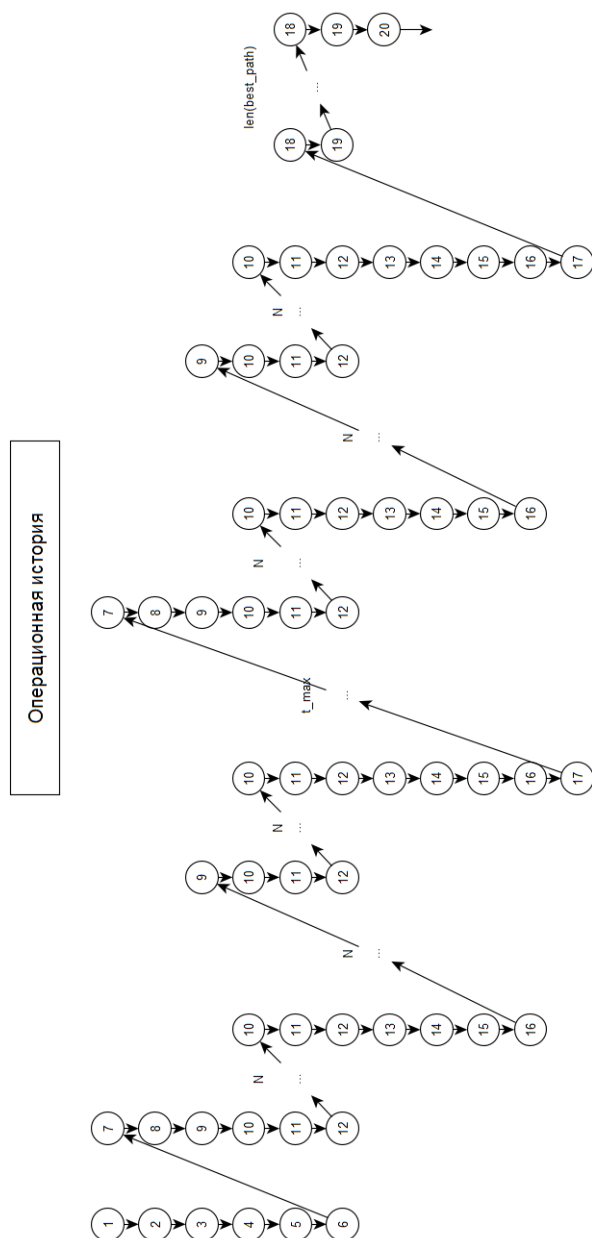


Рисунок 5.1 — Операционная история

ЗАКЛЮЧЕНИЕ

В ходе выполнения лабораторной работы поставленная цель была достигнута. Были решены все задачи:

- реализована основная функция муравьиного алгоритма;
- построен информационный граф;
- построена информационную историю;
- построен граф управления;
- построена операционную историю.