



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»

Домашнее задание №1
по дисциплине "Анализ алгоритмов"
по теме "графовые представления"

Студент Коняев Е.А

Группа ИУ7-53Б

Преподаватели Волкова Л.Л., Строганов Ю.В.

Дата сдачи отчета _____

Оценка (баллы) _____

Оглавление

1	Выполнение задания	3
1.1	Выбор языка программирования и среды разработки	3
1.2	Реализации алгоритмов	3
1.3	Графовые представления	4
	Список использованных источников	6

Глава 1

Выполнение задания

1.1 Выбор языка программирования и среды разработки

Для реализации трех алгоритмов сортировок был выбран язык C, так как данный язык является быстродейственным.

Средой разработки был выбран CLion. Данный выбор обусловлен тем, что данная среда предоставляет возможность разработки приложений под C/C++ и имеет инструменты для отладки кода.

1.2 Реализации алгоритмов

В листинге 1.1 представлен итерационный алгоритм нахождения расстояния Левенштейна. Именно данный алгоритм будет рассмотрен в графовом представлении.

Листинг 1.1 – Листинг итерационного алгоритма нахождения расстояния Левенштейна

```
1  int dist_lev(char *str_1, char *str_2, matrix *d, int print_table_flag)
2  {
3      matrix_t *d = create_matrix(strlen(str_1) + 1, strlen(str_2) +
4      1); //1
5      d->elements[0][0] = 0; //2
6      for (size_t i = 1; i < d->rows; ++i) //3
7          d->elements[i][0] = i; //4
8      for (size_t i = 1; i < d->cols; ++i) //5
9          d->elements[0][i] = i; //6
10
11     for (size_t i = 1; i < d->rows; ++i) //7
12         for (size_t j = 1; j < d->cols; ++j) { //8
13             int offset = 1; //9
14             if (str_1[i - 1] == str_2[j - 1]) //10
15                 offset = 0; //11
16             d->elements[i][j] = get_min(d->elements[i][j - 1] + 1,
17             d->elements[i - 1][j] + 1,
18             d->elements[i - 1][j - 1] +
19             offset); //12
20         }
```

```

20     if (print_table_flag) //13
21         print_matrix(d); //14
22
23     int res = d->elements[d->rows - 1][d->cols - 1]; //15
24     free_matrix(d); //16
25
26     return res; //17
27 }

```

1.3 Графовые представления

На рисунках ниже представлен граф операционный, информационный, операционной истории и информационной истории соответственно для алгоритма поиска расстояния Левенштейна.

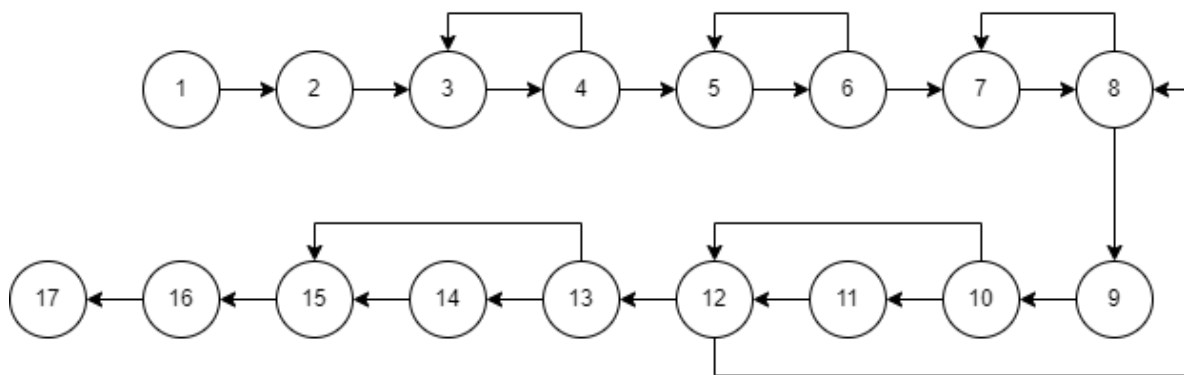


Рисунок 1.1 – Операционный граф

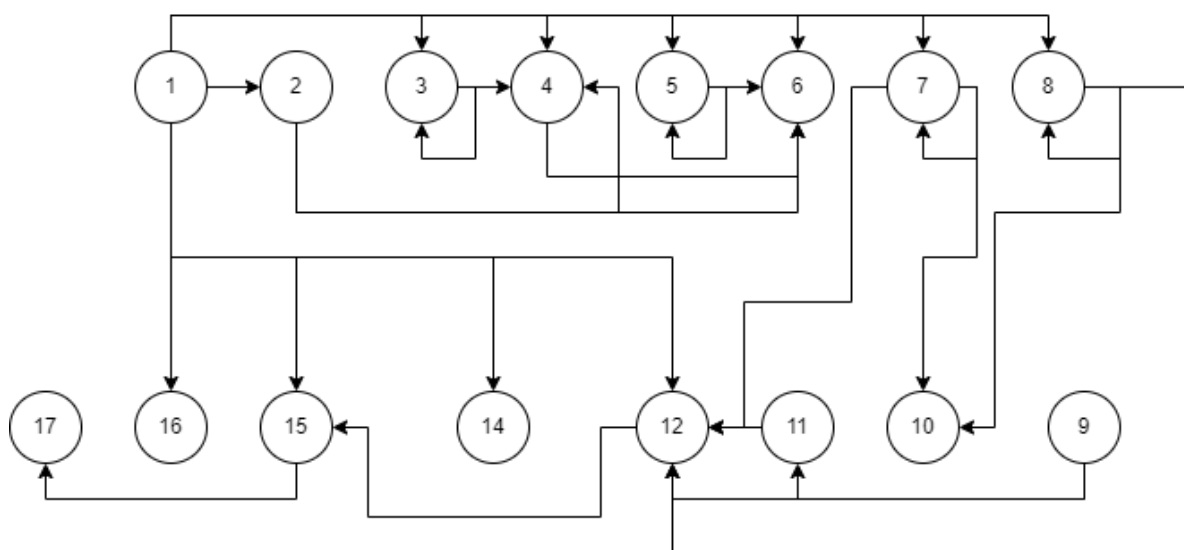


Рисунок 1.2 – Информационный граф

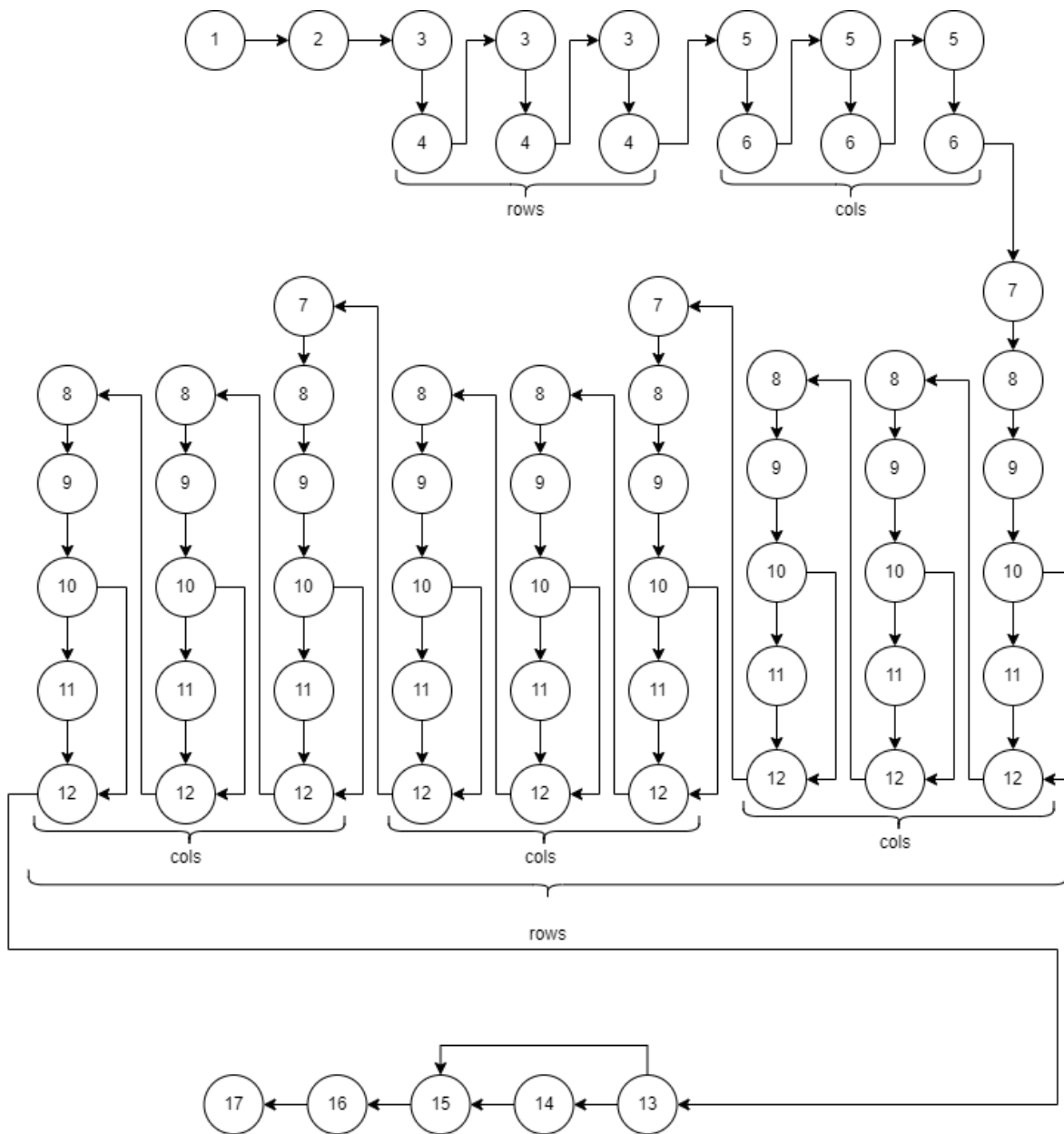


Рисунок 1.3 – Граф операционной истории

Список использованных источников

- [1] The Open Group Base Time [Электронный ресурс]. - URL: <https://pubs.opengroup.org/onlinepubs/9699919799/basedefs/time.h.html> (дата обращения: 21.10.2022).
- [2] Погорелов, Д. А. Сравнительный анализ алгоритмов редакционного расстояния Левенштейна и Дамерау-Левенштейна / Д. А. Погорелов, А. М. Тарзанов. - Синергия Наук. - 2019. URL: <https://elibrary.ru/item.asp?id=36907767> (дата обращения: 21.10.2022).
- [3] Ингерсолл, Г. С. Обработка неструктурированных текстов / Г. С. Ингерсолл, Т. С. Мортон, Э. Л. Фэррис. - ЛитРес, 2022.