



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 по курсу "Анализ алгоритмов"

Тема Графовые модели программ

Студент Гурова Н.А.

Группа ИУ7-54Б

Оценка (баллы) _____

Преподаватель Волкова Л. Л., Строганов Ю. В.

СОДЕРЖАНИЕ

| | | |
|----------|---|----------|
| 1 | Технологическая часть | 2 |
| 1.1 | Выбор языка программирования | 2 |
| 1.2 | Исходный код программы | 2 |
| 1.3 | Модели программ | 3 |
| | СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ | 7 |

1 Технологическая часть

1.1 Выбор языка программирования

При выполнении домашнего задания использовался язык программирования C++ [1].

1.2 Исходный код программы

В листинге 1.1 представлена реализация алгоритма Z-буфера.

Листинг 1.1 – Алгоритм Z-буфера

```
1 screen_t *screen; // (1)
2 std::vector<triangle_t*> triangles; // (2)
3 screen.init(); // (3)
4 triangles.init(); // (4)
5
6 for (int y = screen->height - 1; y >= 0; y--) // (5)
7 {
8     if (y < screen->height && y >= 0) // (6)
9     {
10         auto *depth_array = (double*) malloc(sizeof(double) *
11             screen->width); // (7)
12
13         for (int i = 0; i < screen->width; i++) // (8)
14             depth_array[i] = 1e33; // (9)
15
16         for (int i = 0; i < triangles.size(); i++) // (10)
17             process_level(triangles[i], screen, y, depth_array);
18             // (11)
19
20         for (int x = 0; x < screen->width; x++) // (12)
21             if (not screen->change[x][y] && depth_array[x] == INT_MAX)
22                 // (13)
23                 color_pixel(screen, &(default_color), x, y); // (14)
24
25         free(depth_array); // (15)
26     }
27 }
```

```
25
26 screen.set_all_pixel_true();           // (16)
27 screen.convert();                       // (17)
28 screen.show();                          // (18)
29 screen.set_all_pixel_false();           // (19)
30 screen.free();                           // (20)
31 triangles.free();                       // (21)
```

1.3 Модели программ

На рисунке 1.1 представлен граф управления программы.

На рисунке 1.2 представлен информационный граф программы.

На рисунке 1.3 представлена операционная история программы.

На рисунке 1.4 представлена информационная история программы.

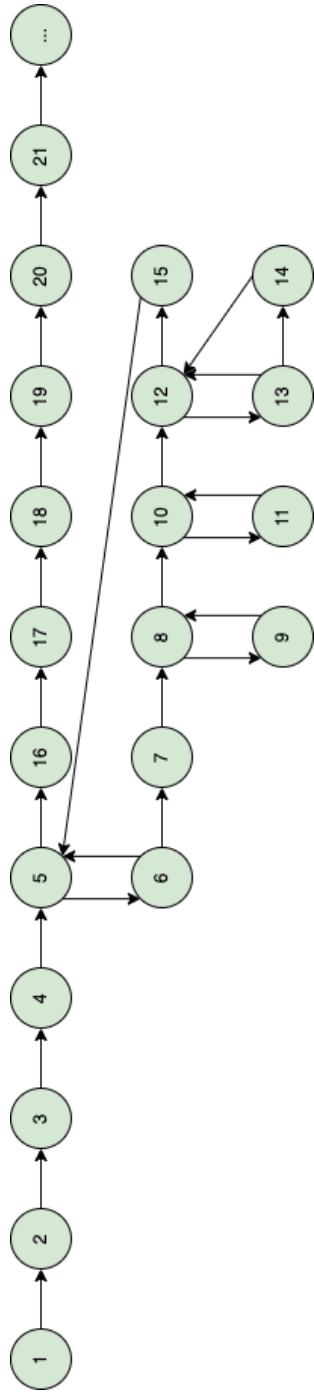


Рисунок 1.1 – Граф управления программы

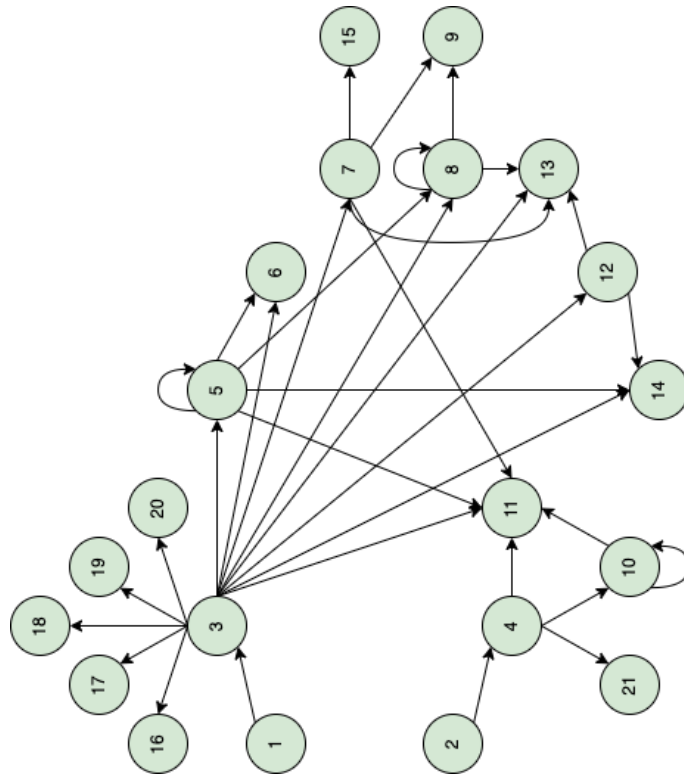
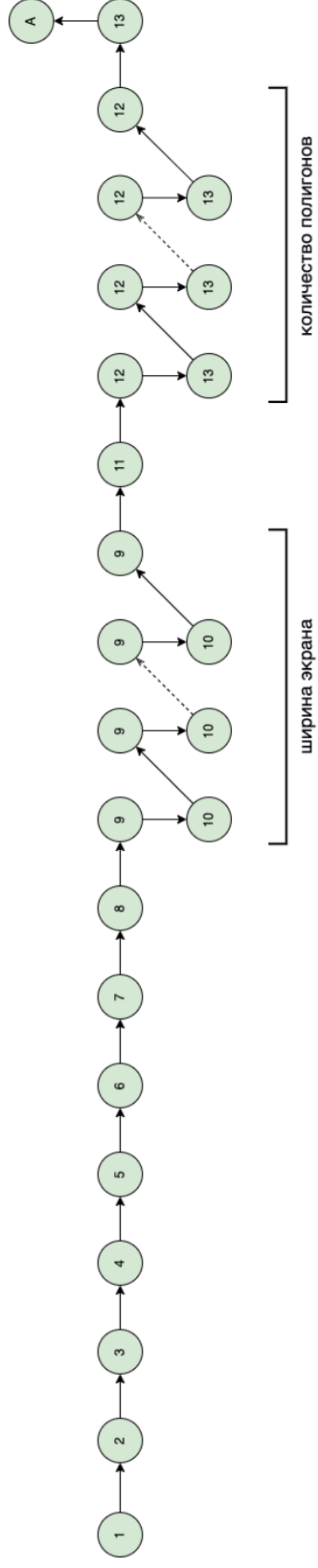


Рисунок 1.2 – Информационный граф программы



высота экрана

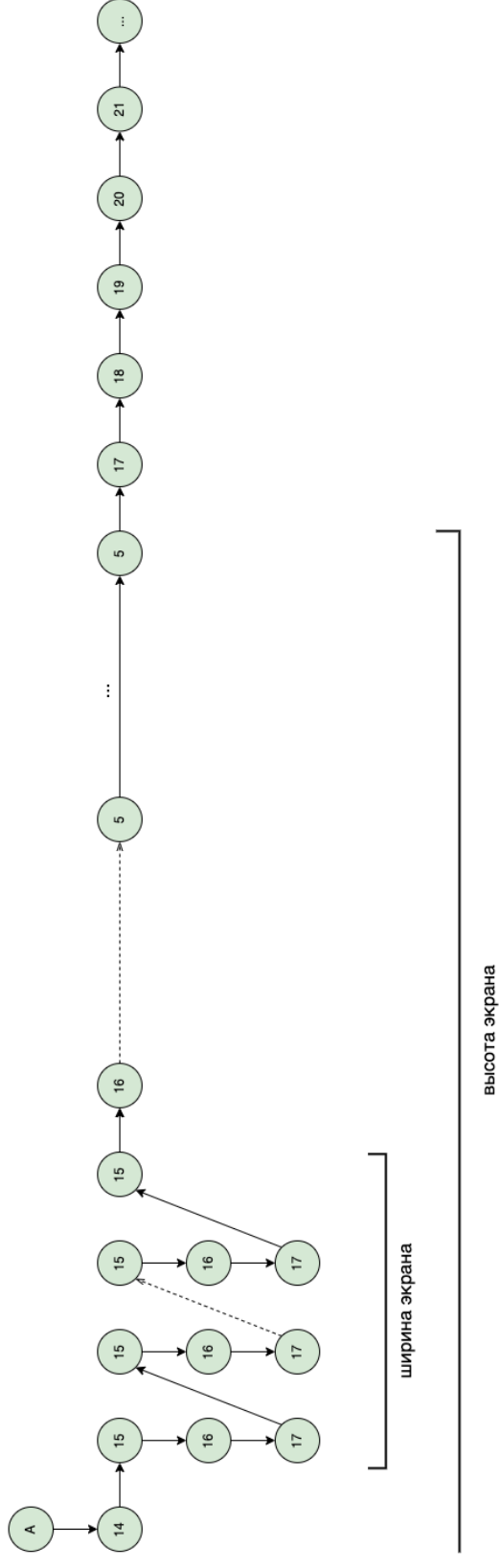


Рисунок 1.3 – Операционная история программы

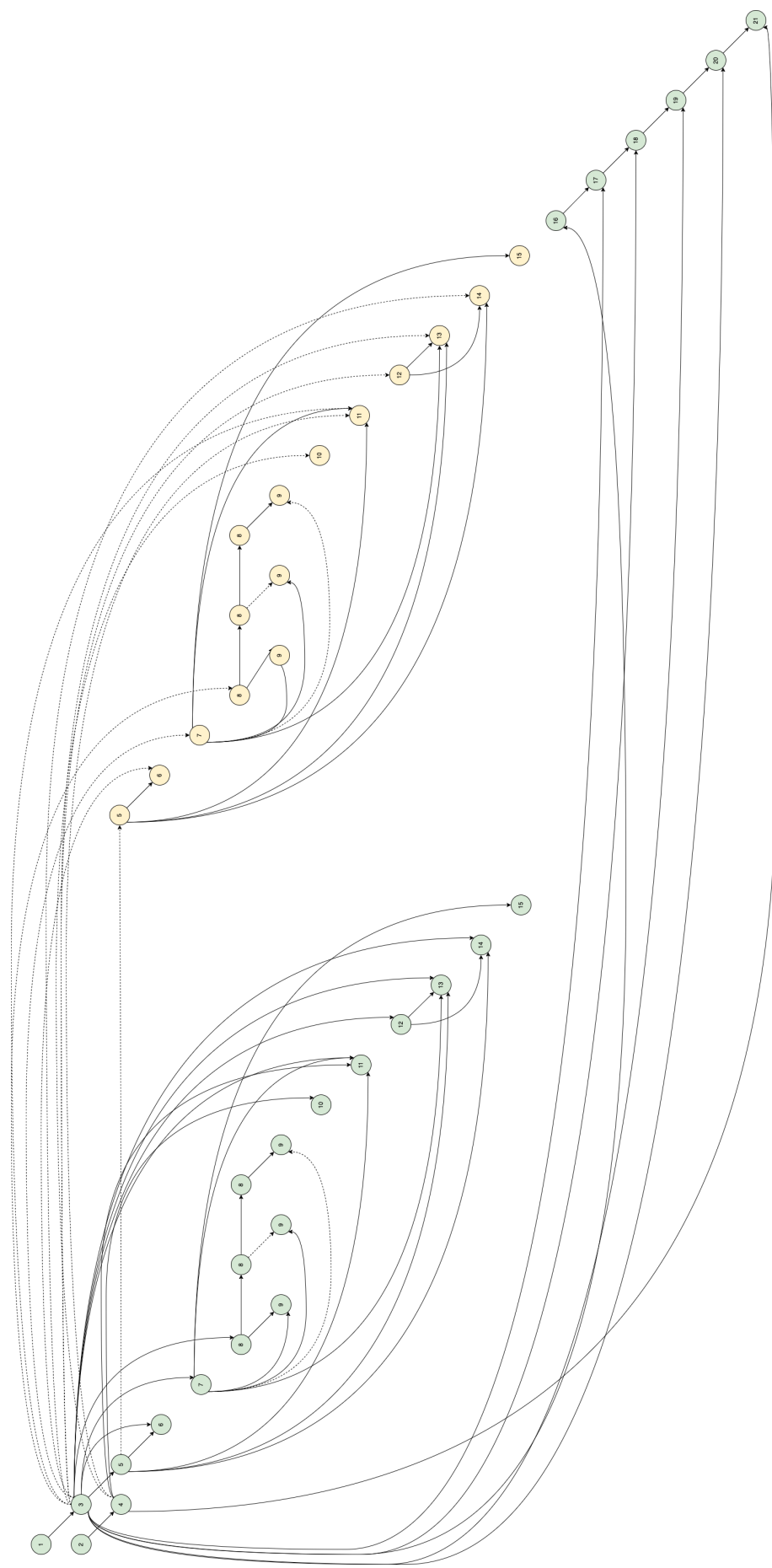


Рисунок 1.4 – Информационная история программы

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Рэнди Д. С. С++ для чайников. — Вильямс, 2018. — С. 400. — (Для чайников).