



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»

НАПРАВЛЕНИЕ ПОДГОТОВКИ _____ «09.03.04 Программная инженерия»

ОТЧЕТ ПО ДОМАШНЕЙ РАБОТЕ №1

Название: _____ Графовое представление программы

Дисциплина: _____ Анализ алгоритмов

Студент	<u>ИУ7-53Б</u>	_____	<u>И. О. Артемьев</u>
	Группа	Подпись, дата	И. О. Фамилия

Преподаватель	_____	<u>Л. Л. Волкова</u>
	Подпись, дата	И. О. Фамилия

Москва, 2021 г.

1. Код программы

Листинг 1.1 – Реализация алгоритма четно-нечетной сортировки

```
1 #include <stdio.h>
2 #include <stdlib.h>
3 #include <time.h>
4
5 #define SUCCESS 0
6
7
8 int main(void) {
9
10     int array_len;                                // (1)
11
12     fscanf(stdin, "%d", &array_len);              // (2)
13     int *array = (int *) malloc(array_len * sizeof(int)); // (3)
14
15     srand(time(NULL));                             // (4)
16     for (int i = 0; i < array_len; ++i) {          // (5)
17         array[i] = random() % 100;                // (6)
18     }
19
20     int is_sorted = 0;                             // (7)
21     while (is_sorted == 0) {                       // (8)
22         is_sorted = 1;                             // (9)
23         int temp;                                  // (10)
24
25         for (int i = 1; i < array_len - 1; i += 2) { // (11)
26             if (array[i] > array[i + 1]) {          // (12)
27                 temp = array[i];                    // (13)
28                 array[i] = array[i + 1];            // (14)
29                 array[i + 1] = temp;                 // (15)
30                 is_sorted = 0;                      // (16)
31             }
32         }
33
34         for (int i = 0; i < array_len - 1; i += 2) { // (17)
35             if (array[i] > array[i + 1]) {          // (18)
36                 temp = array[i];                    // (19)
37                 array[i] = array[i + 1];            // (20)
38                 array[i + 1] = temp;                 // (21)
39                 is_sorted = 0;                      // (22)
40             }
41         }
42     }
```

```
42     }
43
44     free(array);                                // (23)
45
46     return SUCCESS;
47 }
```

2. Конструкторский раздел

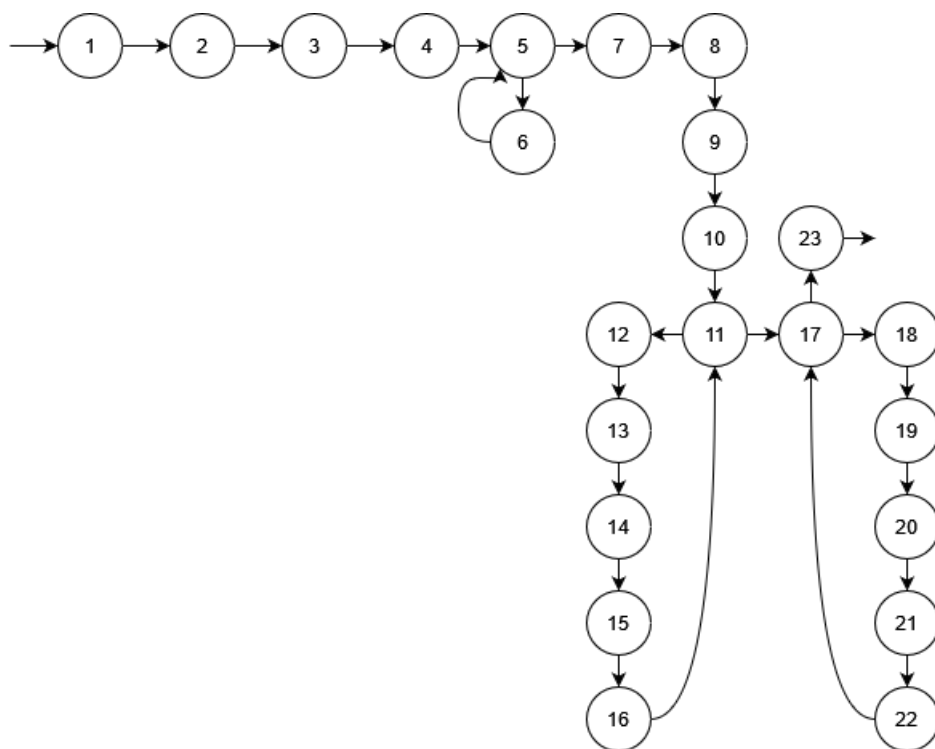


Рисунок 2.1 – Операционный граф

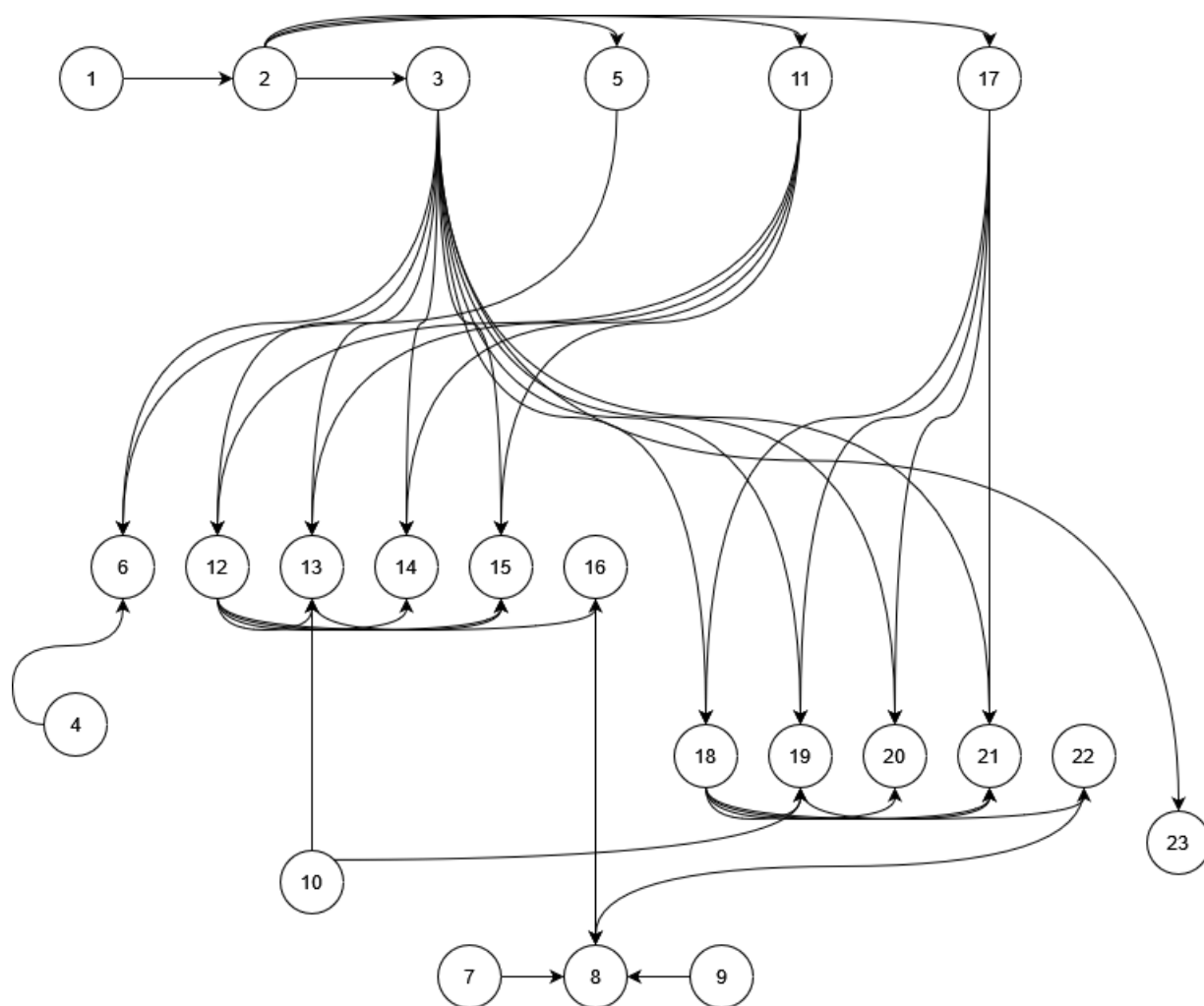


Рисунок 2.2 – Информационный граф

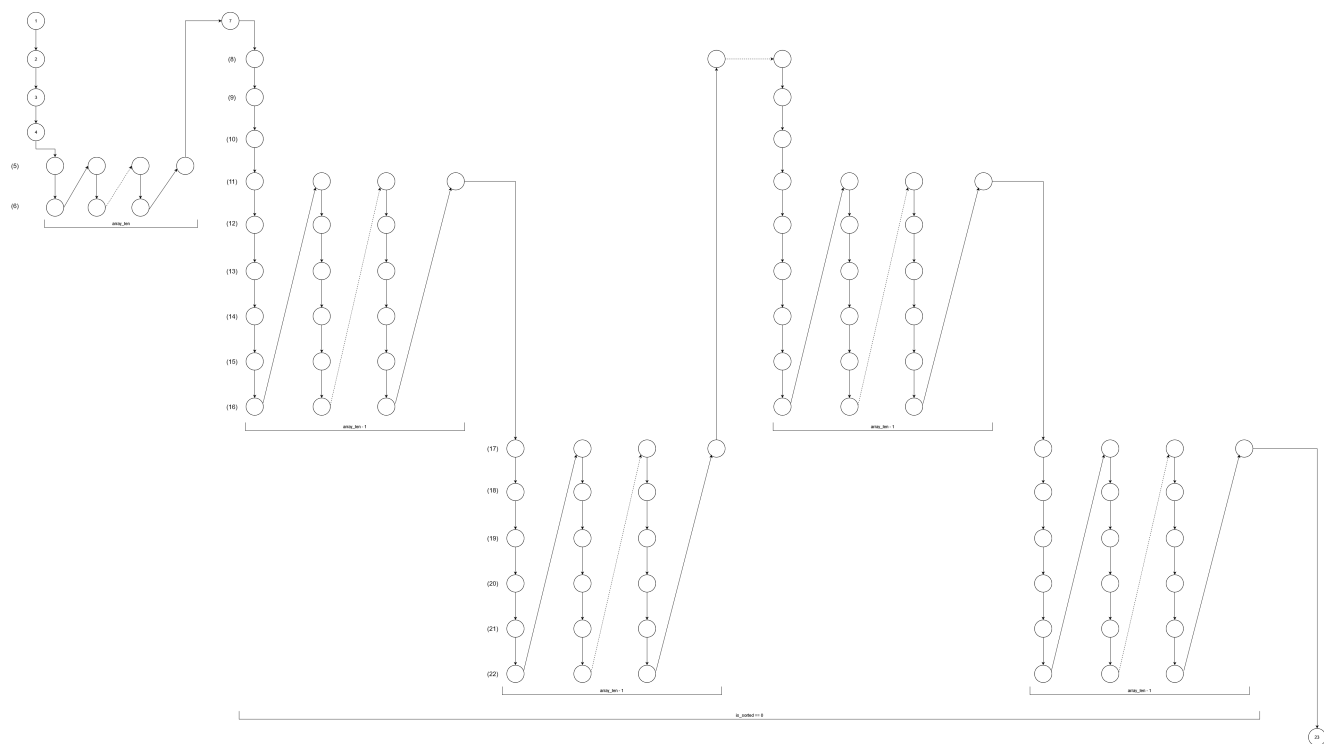


Рисунок 2.3 – Операционная история

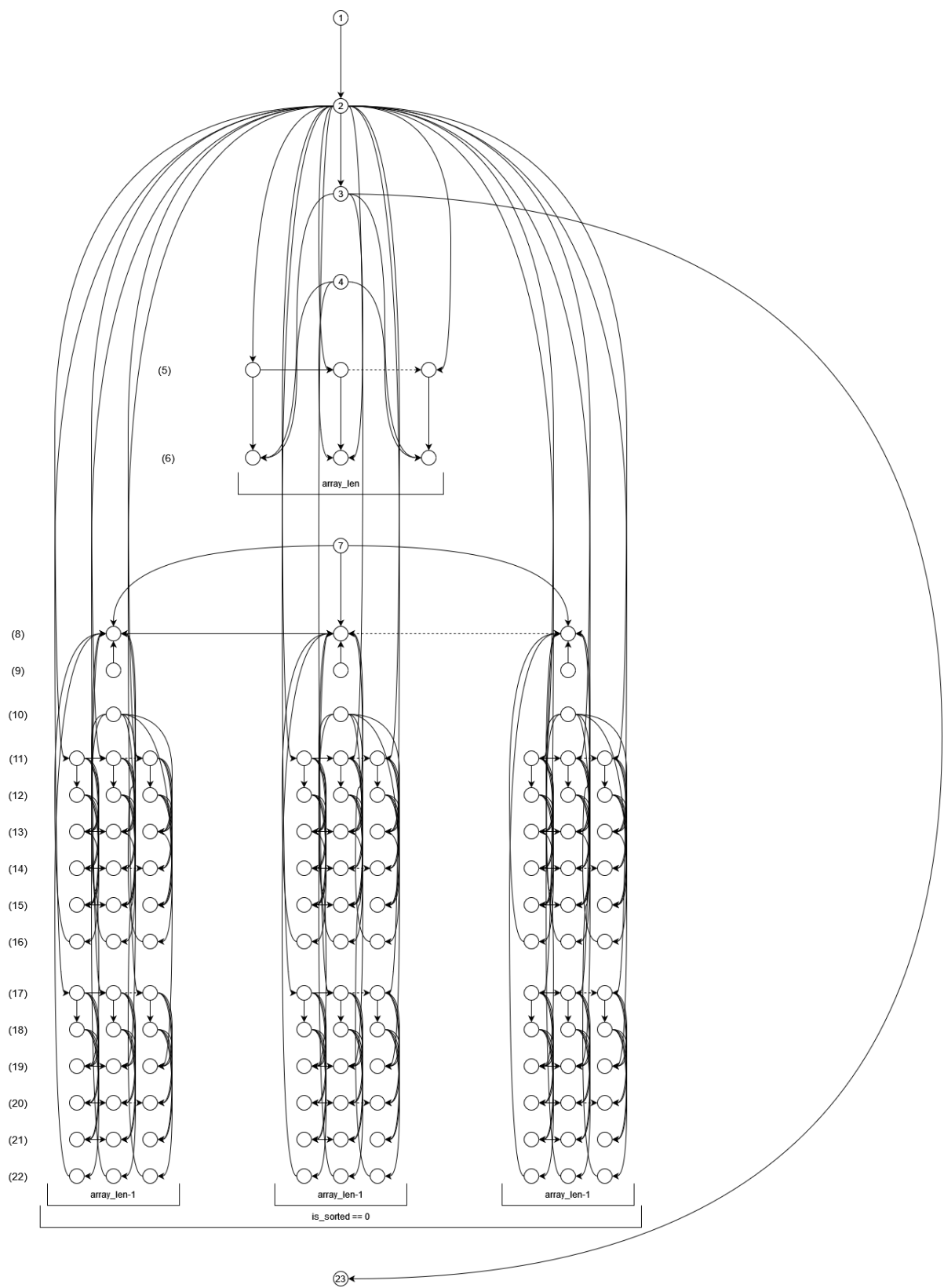


Рисунок 2.4 – Информационная история