



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

Название: Графовые модели программы

Дисциплина: Анализ алгоритмов

Студент

ИУ7-54Б

(Группа)

Л.Е.Тартыков

(Подпись, дата)

(И.О. Фамилия)

Преподаватель

Л.Л. Волкова

(Подпись, дата)

(И.О. Фамилия)

Москва, 2021

Содержание

1	Исходный код алгоритма	3
1.1	Листинги программ	3
2	Модели программ	5
2.1	Операционный граф программы	5
2.2	Информационный граф программы	5
2.3	Операционная история программы	6
2.4	Информационная история программы	8

1 Исходный код алгоритма

В данном разделе приведен код алгоритма решения СЛАУ методом Гаусса.

1.1 Листинги программ

Ниже представлены листинг кода алгоритма решения СЛАУ методом Гаусса.

Листинг 1.1 – Программный код алгоритма нахождения минимума в матрице (последовательная реализация).

```
1 int solve_slay(double **matrix, int row, int column, int *  
   count_null_elem)  
2 {  
3     int code_error = ERR_OK; (1)  
4     double main_elem = 0; (2)  
5     int main_index = 0; (3)  
6     double temp_del = 0; (4)  
7  
8     for (int k = 0; k < row; k++) (5)  
9     {  
10        main_elem = matrix[k][0]; (6)  
11        main_index = 0; (7)  
12  
13        for (int j = 1; j < column - 1; j++) (8)  
14        {  
15            if (fabs(matrix[k][j]) > fabs(main_elem)) (9)  
16            {  
17                main_elem = matrix[k][j]; (10)  
18                main_index = j; (11)  
19            }  
20        }  
21  
22        if (!(main_elem > 0 || main_elem < 0)){ (12)  
23            code_error = ERR_NOT_OK; (13)  
24        }  
25        else  
26        {  
27            for (int i = 0; i < column; i++) (14)  
28            matrix[k][i] /= main_elem; (15)  
29  
30            for (int i = 0; i < row; i++) (16)
```

```

31         {
32             if (i != k) (17)
33             {
34                 temp_del = -matrix[i][main_index]; (18)
35                 for (int j = 0; j < column; j++){ (19)
36                     matrix[i][j] = matrix[i][j] + \ (20)
37                         temp_del * matrix[k][j];
38                 }
39             }
40         }
41     }
42 }
43
44 if (code_error == ERR_OK) (21)
45 {
46     for (int i = 0; i < row; i++){ (22)
47         if (matrix[i][column - 1] == 0 || \
48             fabs(matrix[i][column - 1]) < EPS) (23)
49             *count_null_elem += 1; (24)
50     }
51 }
52 return code_error;
53 }

```

2 Модели программ

2.1 Операционный граф программы

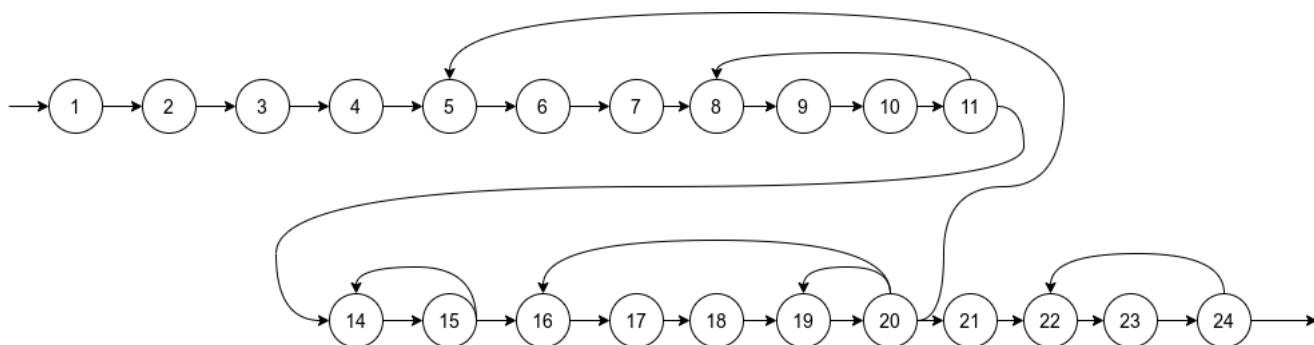


Рисунок 2.1 – Операционный граф программы.

2.2 Информационный граф программы

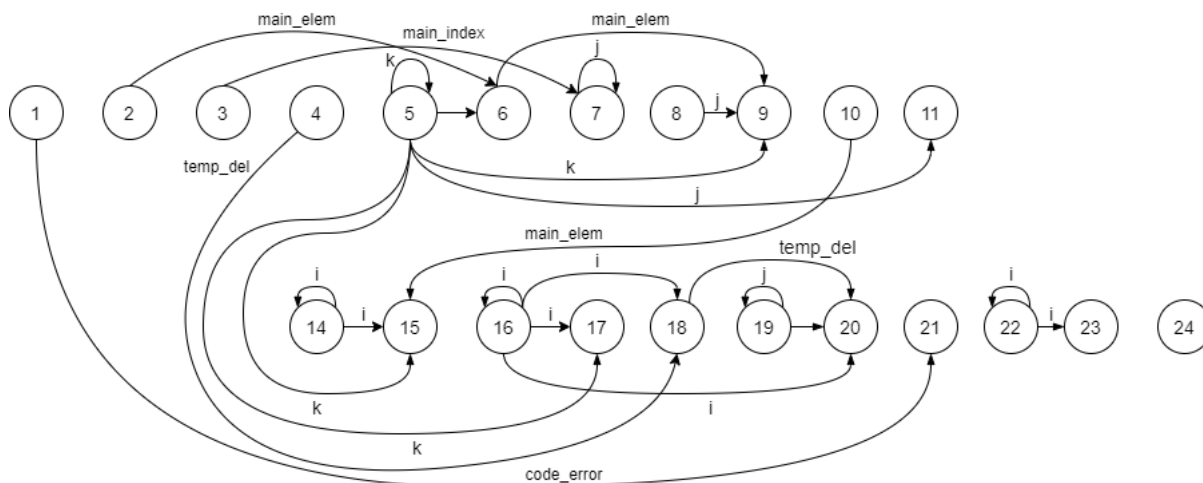


Рисунок 2.2 – Информационный граф программы.

2.3 Операционная история программы

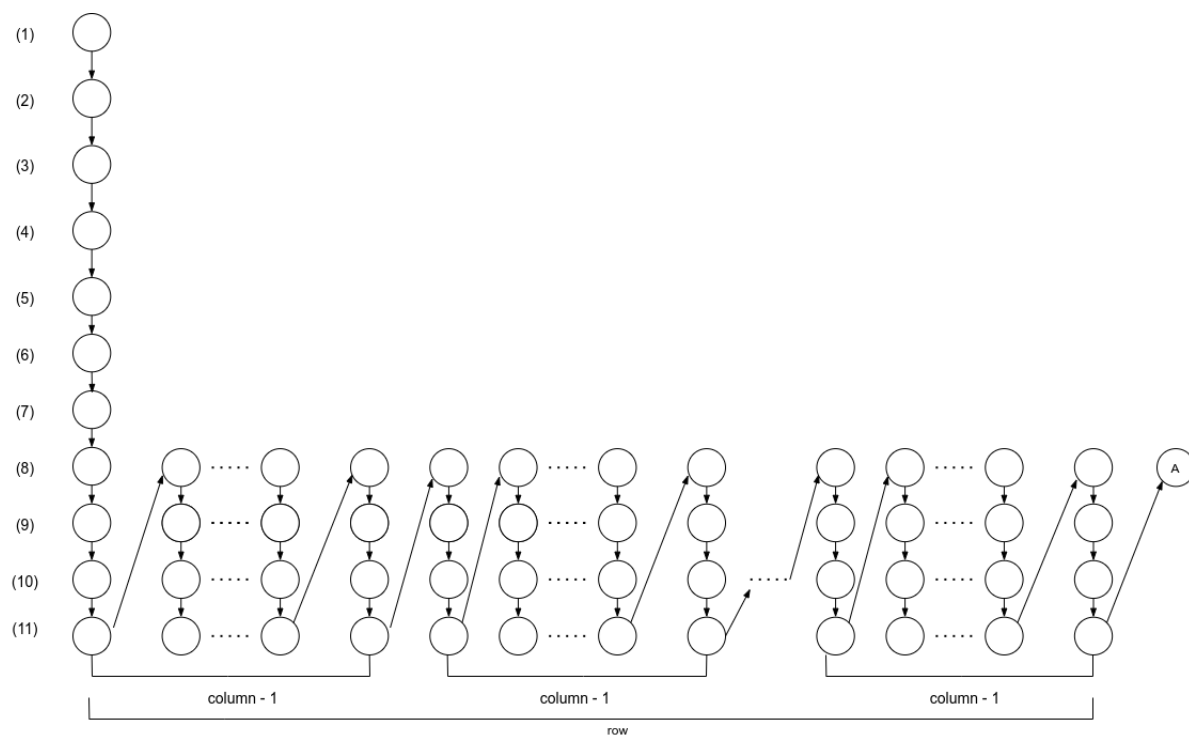


Рисунок 2.3 – Операционная история программы.

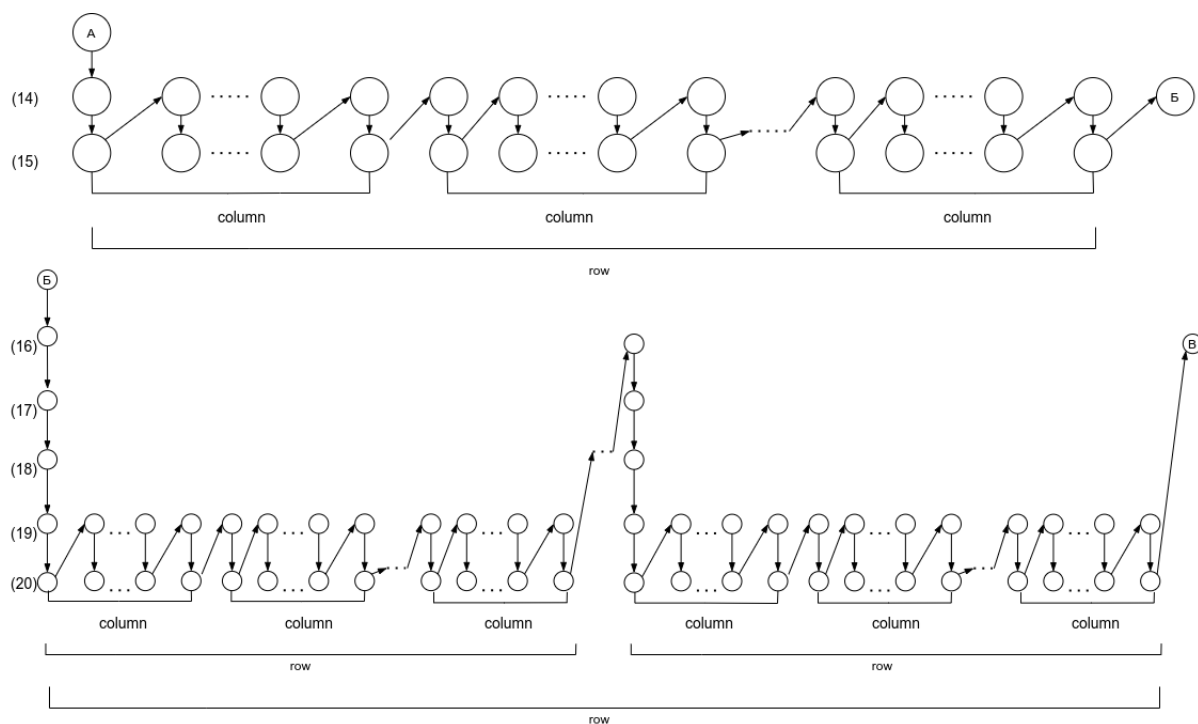


Рисунок 2.4 – Операционная история программы.

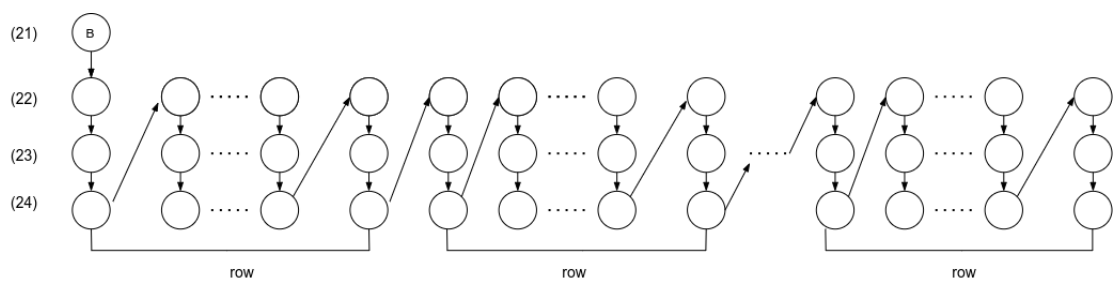


Рисунок 2.5 – Операционная история программы.

2.4 Информационная история программы

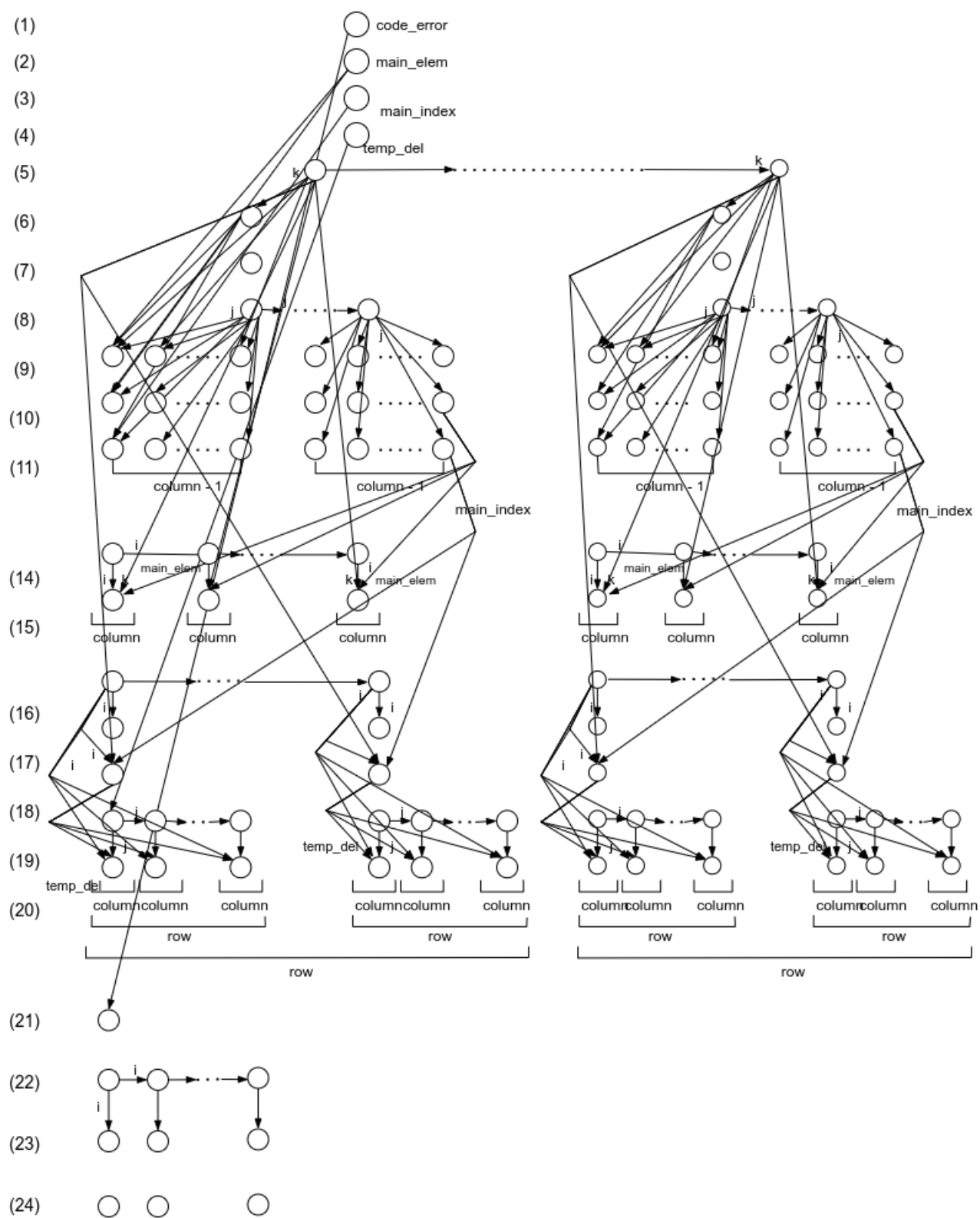


Рисунок 2.6 – Информационная история программы.