



Государственное образовательное учреждение высшего профессионального
образования «Московский Государственный Технический Университет
имени Н.Э. Баумана»

ОТЧЕТ

По лабораторной работе №6
По курсу «Анализ алгоритмов»
Тема: «Графовые модели»

Студент:
Группа

Жарова Е.А.
ИУ7-51

Москва, 2017

Оглавление

Постановка задачи.....	1
Описание алгоритма	1
Реализация алгоритма.....	3
Графовые модели	5
Граф управления программы	5
Информационный граф.....	6
Операционная история	6
Информационная история	8
Заключение	8

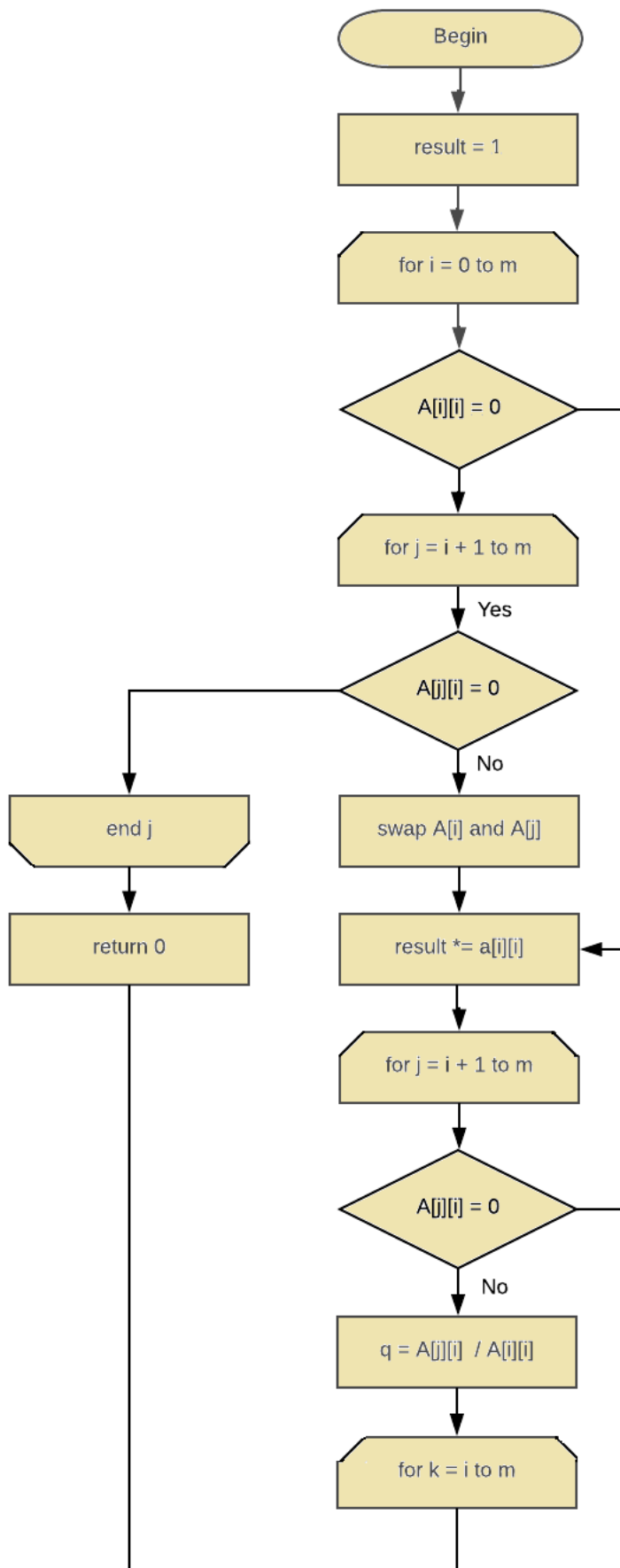
Постановка задачи

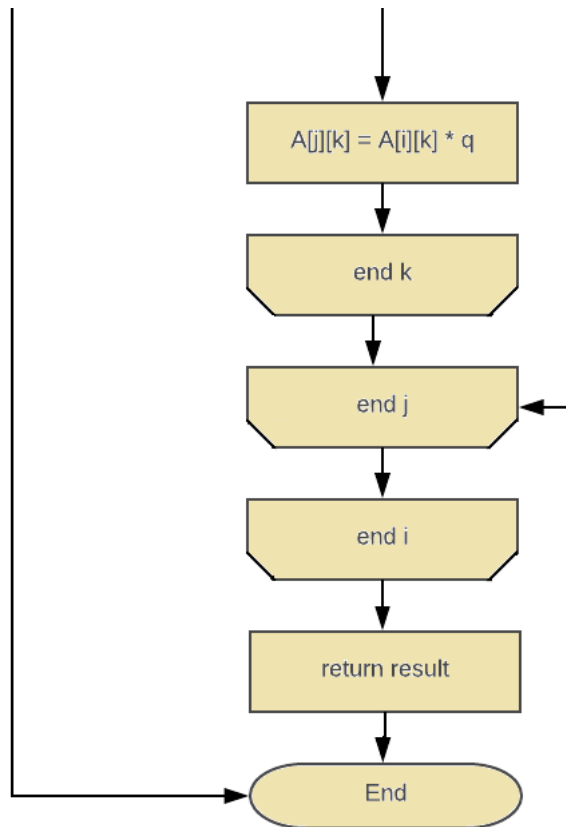
Составить графовые модели для произвольного алгоритма (минимальный размер – 25 строк. Минимальное количество циклов – 2). Мной реализован алгоритм решения СЛАУ методом Гаусса.

Требуемые графовые модели:

- 1) Граф управления программы
- 2) Информационный граф
- 3) Операционная история
- 4) Информационная история

Описание алгоритма





Реализация алгоритма

Алгоритм был реализован на языке C++:

```

double gauss(unsigned m, double** matrix)
{
    double result = 1; //(1)
    for (unsigned i = 0; i < m; ++i)
    {
        // обмен строк, если ячейка matrix[i][i] нулевая на ту, где ячейка на той
        же позиции ненулевая if (fabs(matrix[i][i]) < accuracy) (2)
        {
            for (unsigned j = i + 1; j < m; ++j)
            {
                if (fabs(matrix[j][i]) > accuracy) //(3)
                {
                    double* buffer = matrix[i]; //(4)
                    matrix[i] = matrix[j]; //(5)
                    matrix[j] = buffer; //(6)
                    break;
                }
            }
        }
        // если обмен не произошел – определитель равен нулю if (fabs(matrix[i][i])
        < accuracy) (7)
        {
            return 0; //(8)
        }
        // вычисляем новую итерацию определителя result *= matrix[i][i]; (9)
        // приводим к треугольному виду
        for (unsigned j = i + 1; j < m; ++j)
        {
            if (fabs(matrix[j][i]) >= accuracy) //(10)
            {
                double q = matrix[j][i] / matrix[i][i]; //(11)

```

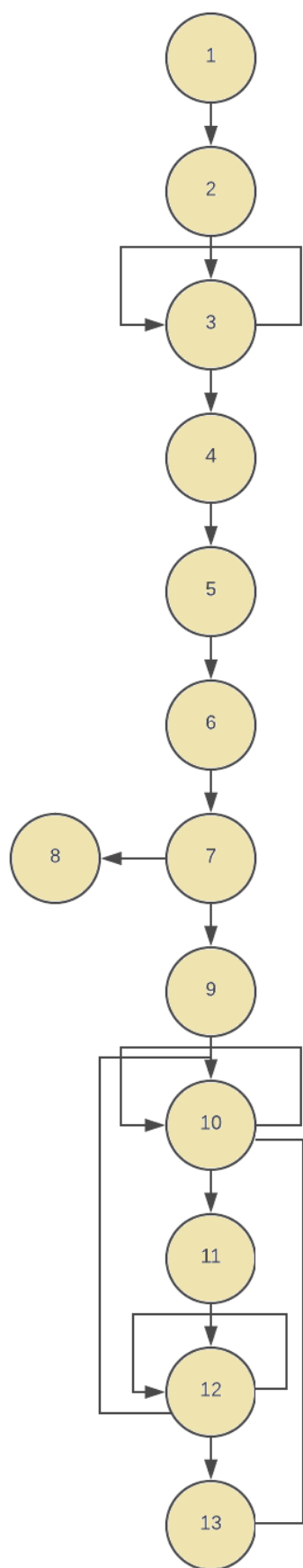
```

        for (unsigned k = i; k < m; ++k)
        {
            matrix[j][k] -= matrix[i][k] * q; //(12)
        }
    }
}
return result; //(13)
}

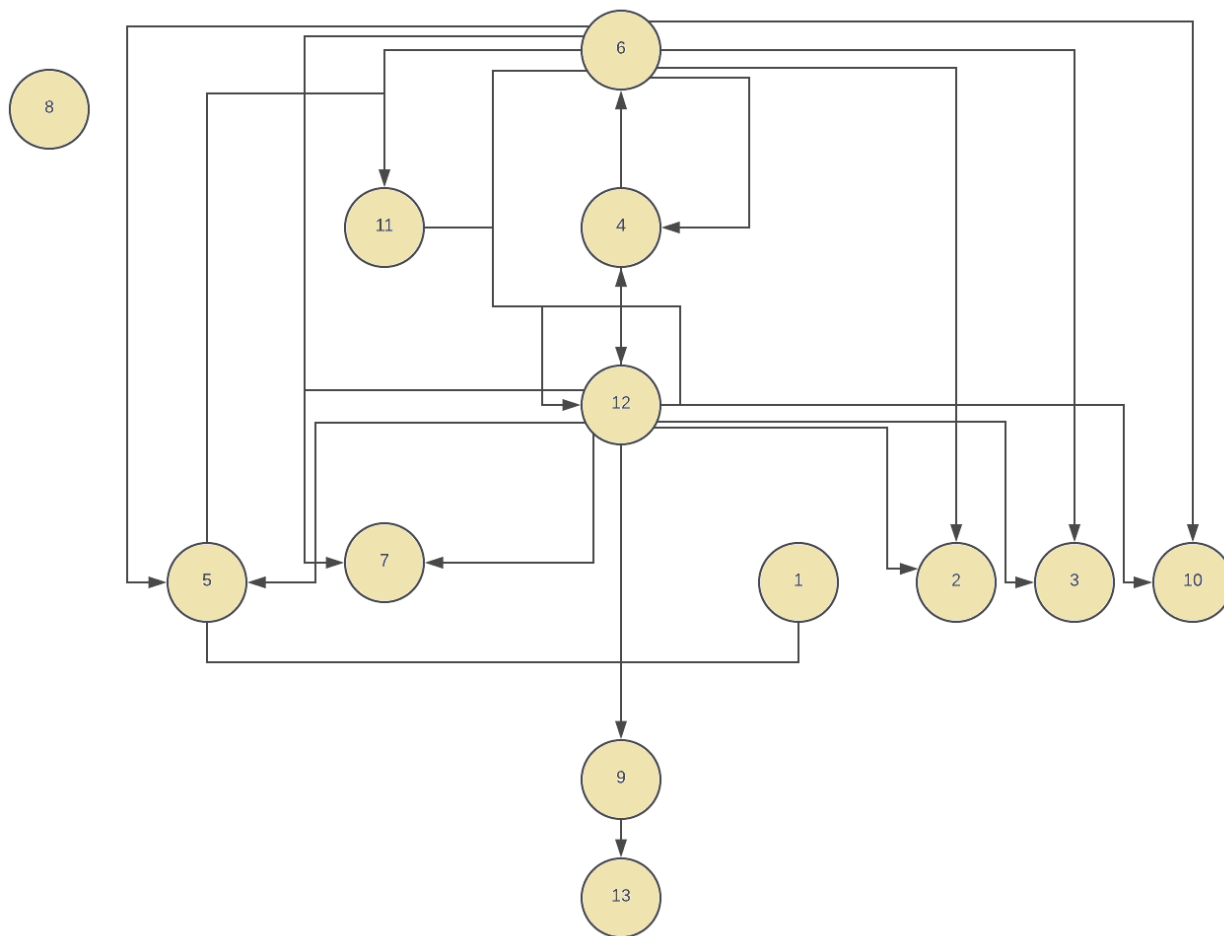
```

Графовые модели

Граф управления программы



Информационный граф



Операционная история

Для операционной истории использовался расчет определителя матрицы:

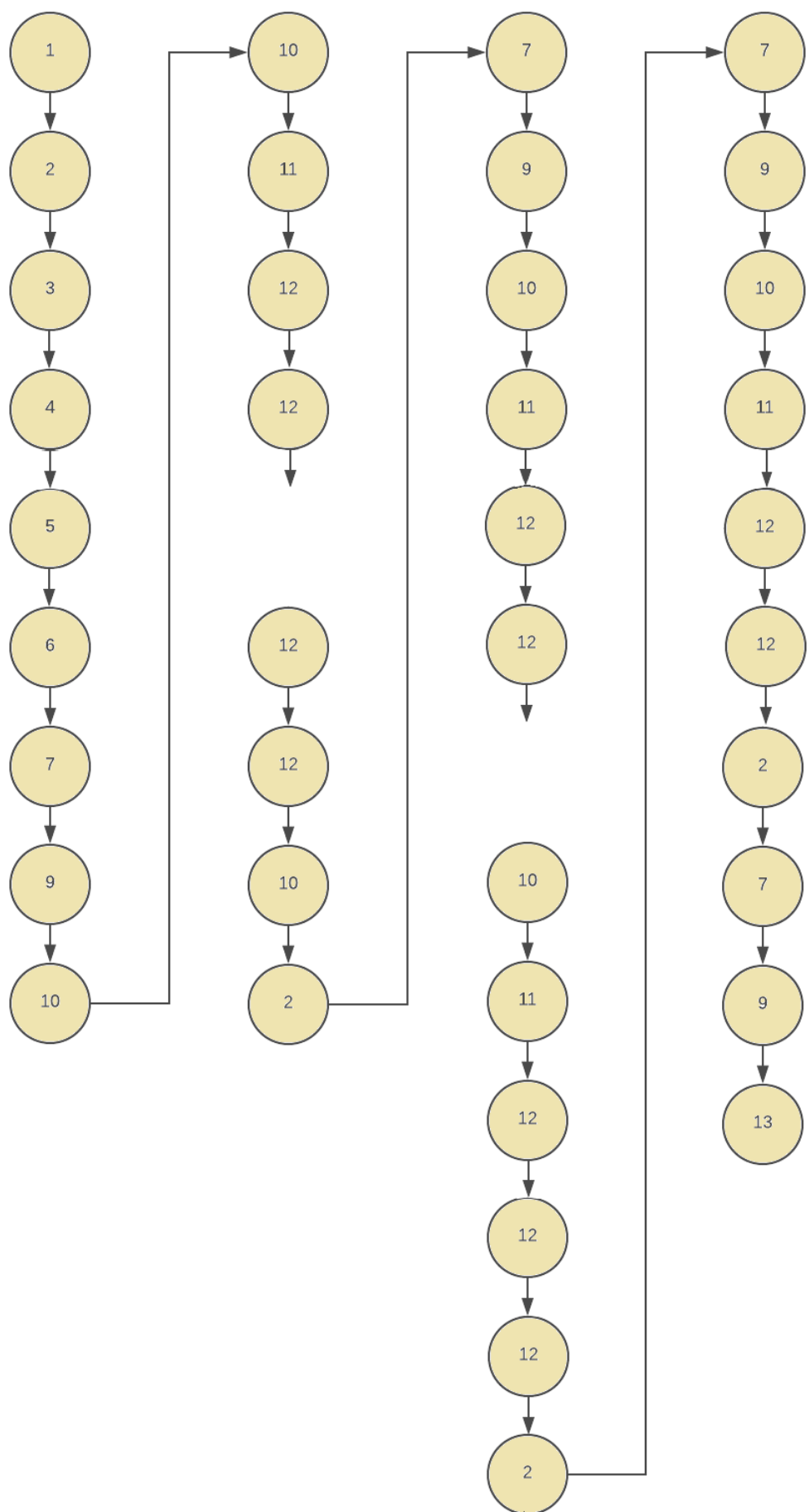
0 4 3 2

3 3 8 7

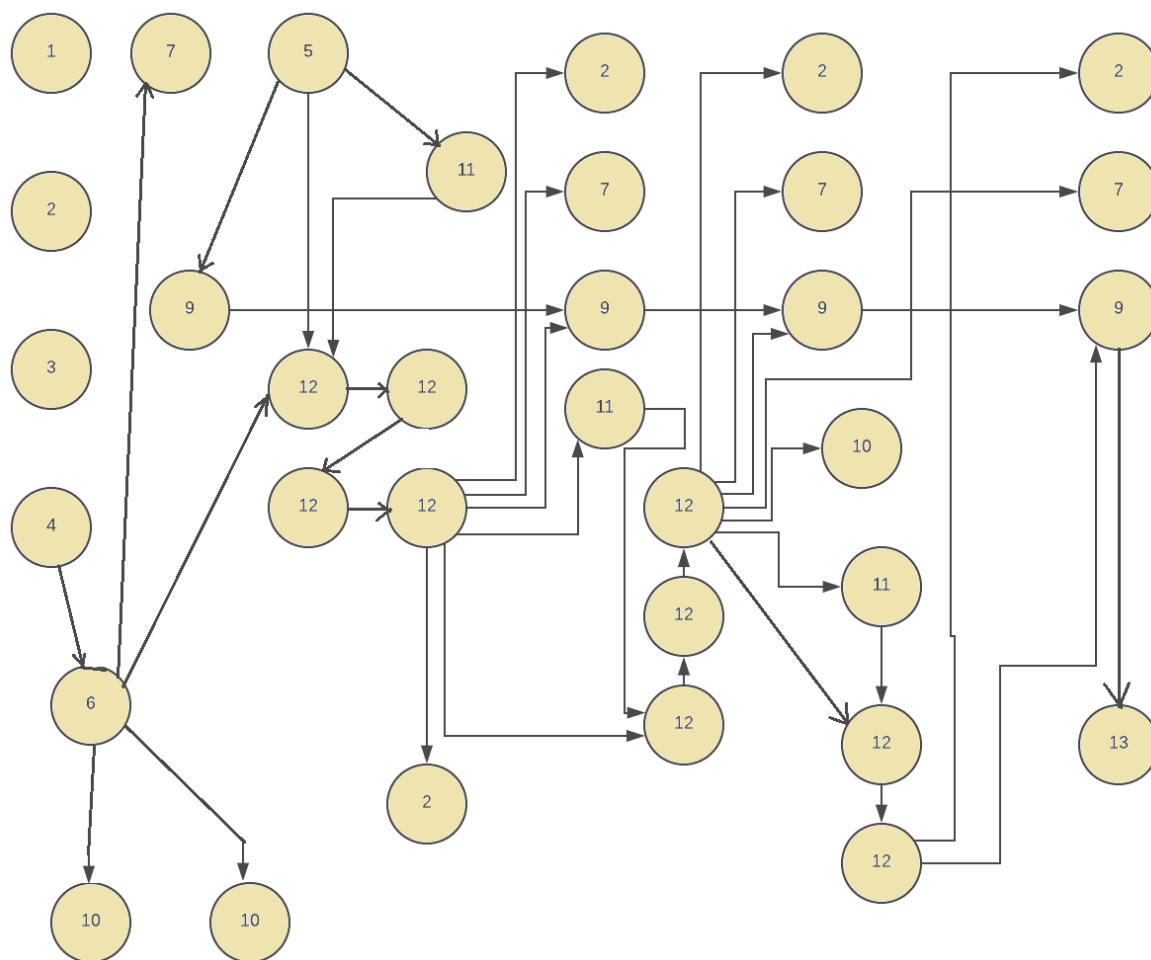
6 1 3 3

0 1 1 3

Определитель данной матрицы = 252



Информационная история



Заключение

В ходе работы был реализован и представлен в виде графовых моделей алгоритм нахождения определителя матрицы методом Гаусса.