

Федеральное государственное автономное
образовательное учреждение
высшего профессионального образования
«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт космических и информационных технологий

институт

Кафедра «Информатика»

кафедра

ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ № 1

Детерминированные конечные автоматы

Тема / Аббревиатура, слово-символ (слово-бренд)

Преподаватель

дата

А.С. Кузнецов

инициалы, фамилия

/

подпись

Студент

КИ14-17Б

код (номер) группы

дата

А.А. Артамонов

инициалы, фамилия

/

подпись

Красноярск 2016

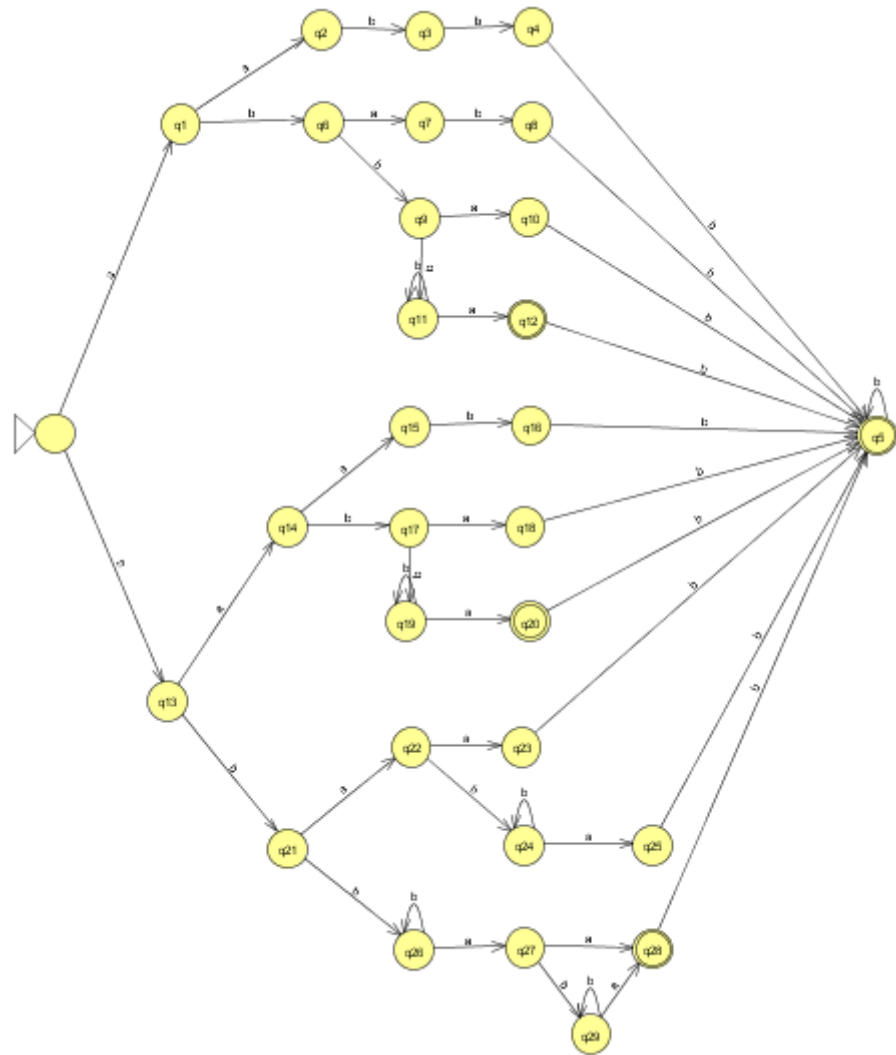
Цель работы

Реализация и исследование детерминированных конечных автоматов.

Задача

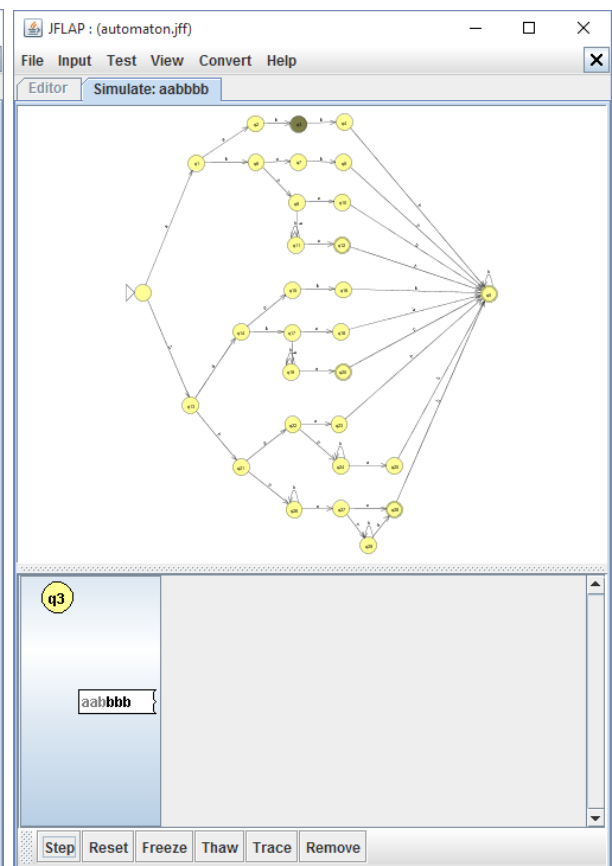
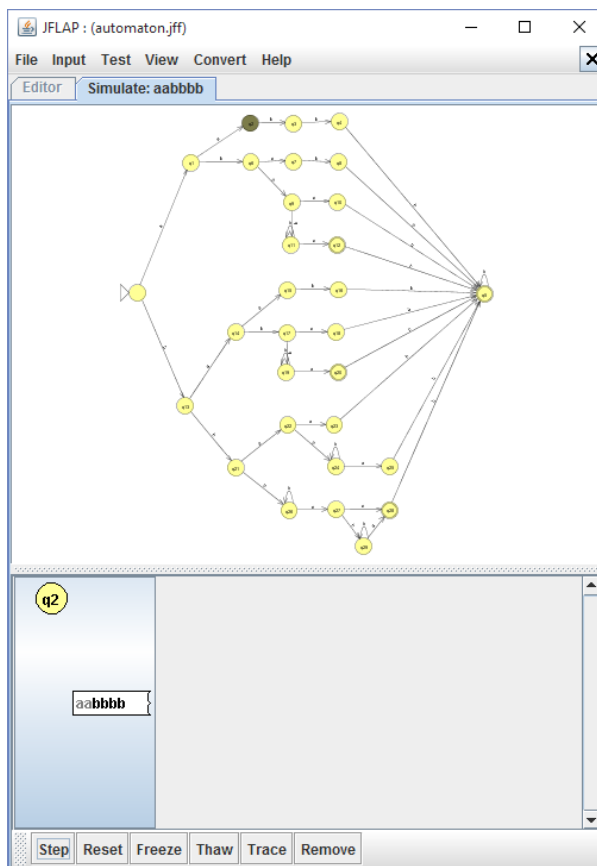
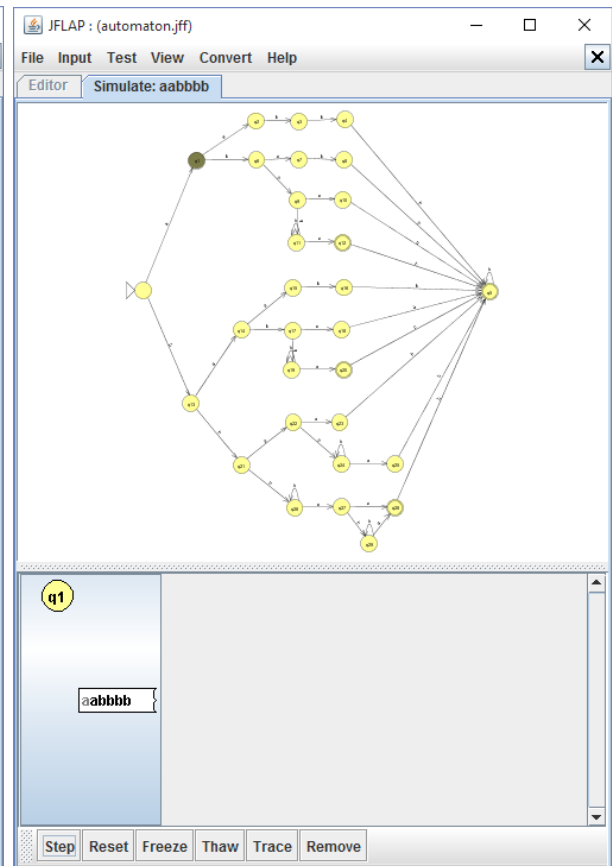
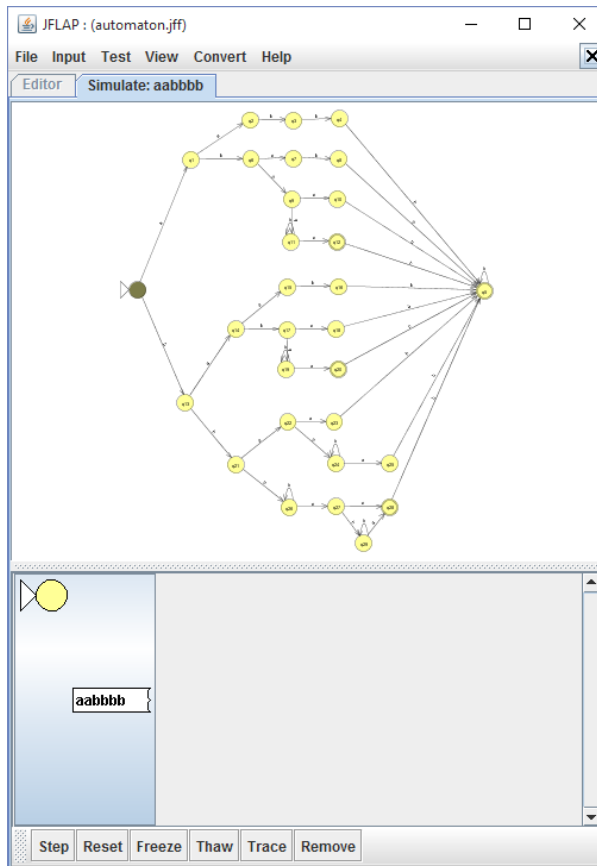
Необходимо построить ДКА в системе JFLAP и произвести программную реализацию. В коде программы обязательно наличие сущностей и функций, относящихся к табличному представлению автомата. Использование функций обработки строковых данных **запрещено**. Результат работы, выдаваемый программой на экран, внешне должен быть схож, а фактически эквивалентен результату, выдаваемому JFLAP на тех же тестовых цепочках. Построить ДКА, допускающий в алфавите $\{a, b\}$ все строки, где количество символов a равно 2, и количество символов b больше 2.

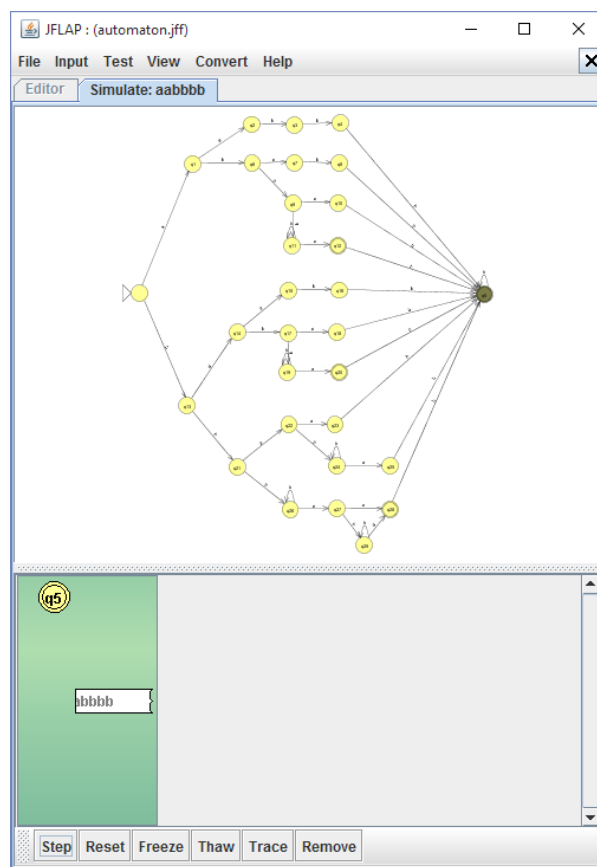
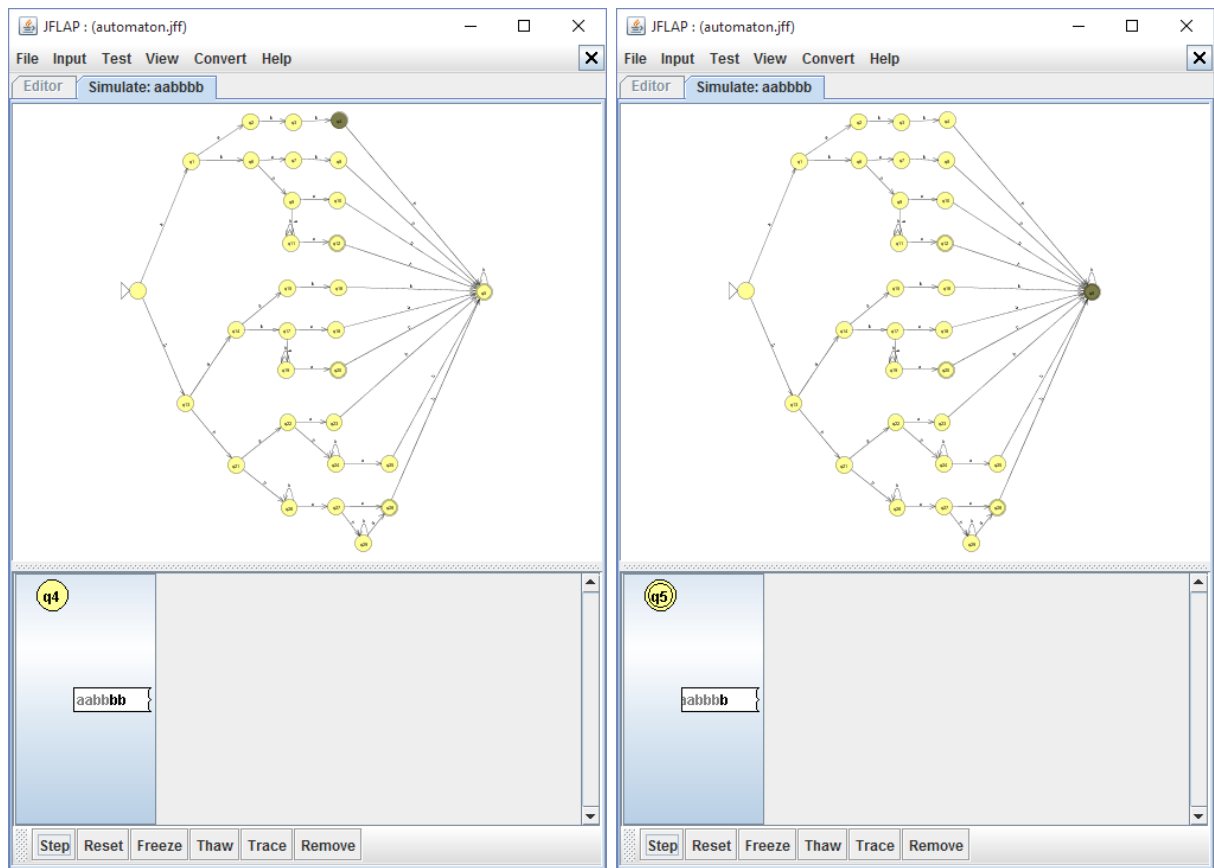
Граф переходов ДКА



Распознавание тестовых цепочек

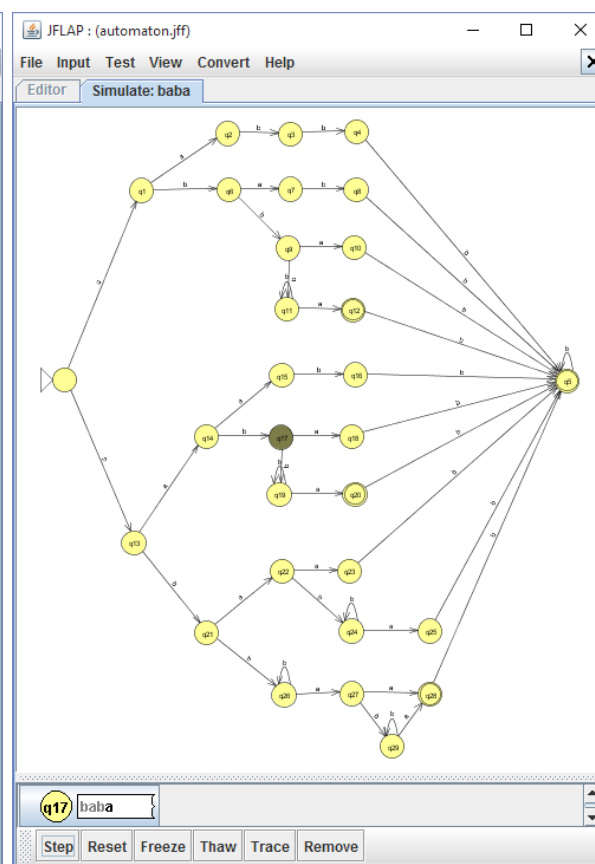
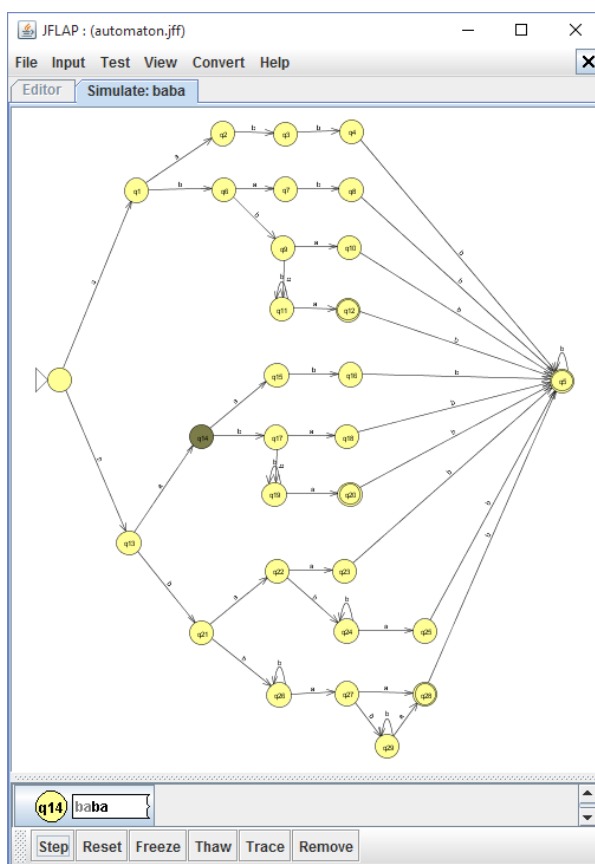
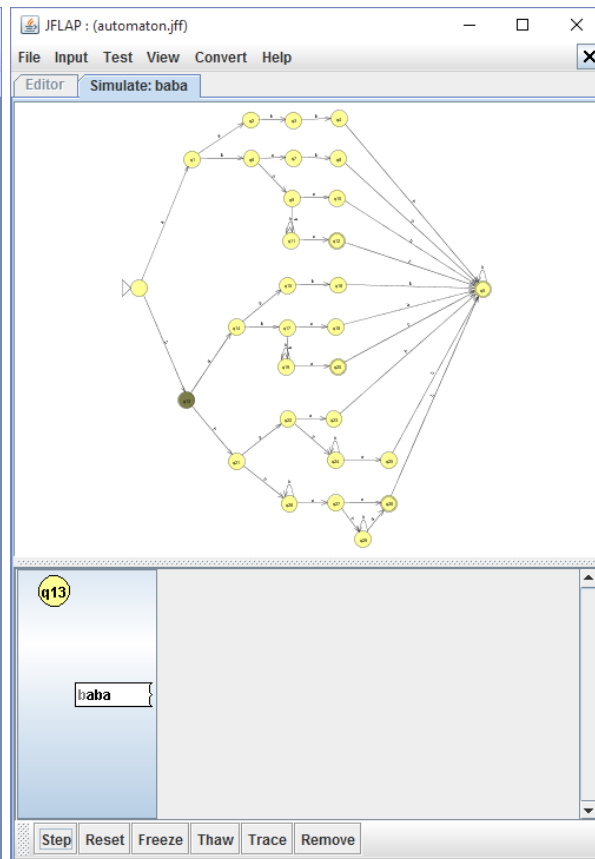
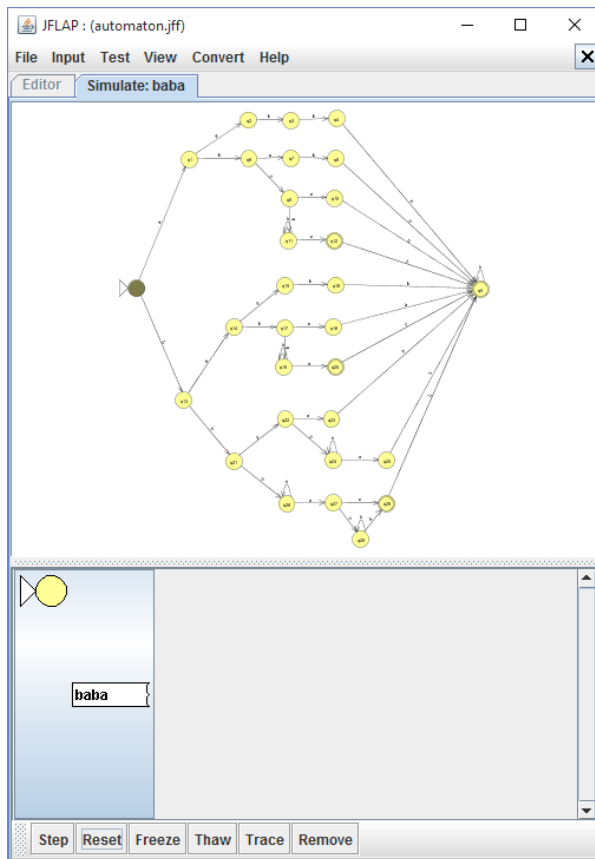
Входная строка: «aabbbb»

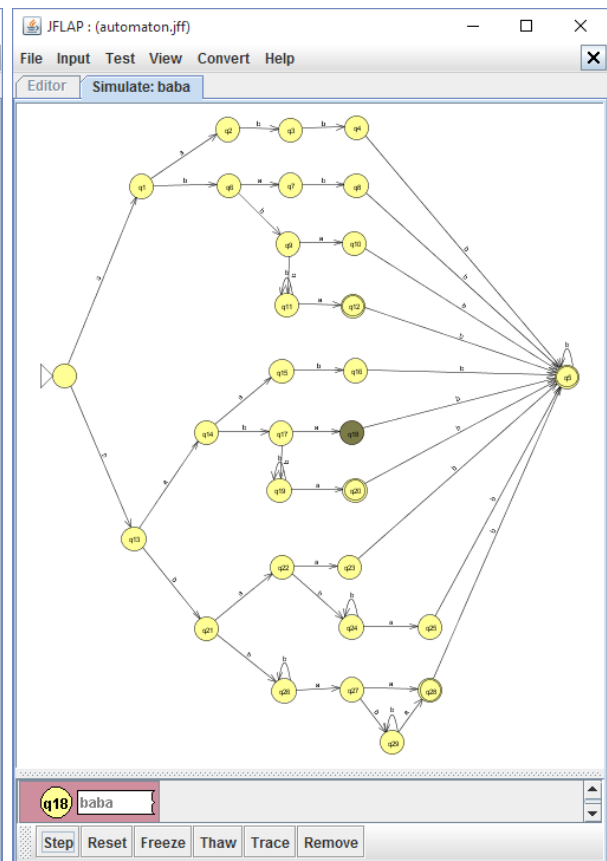
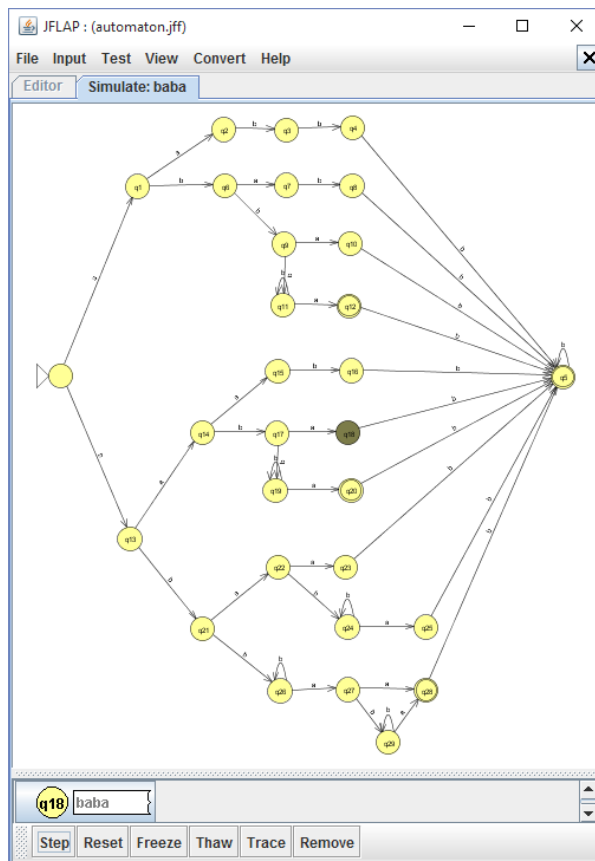




Результат – строка принята.

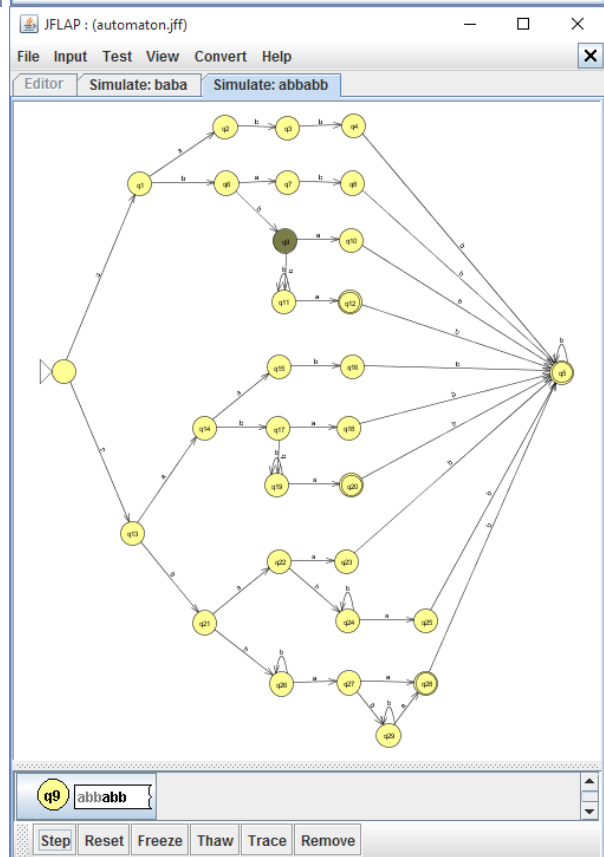
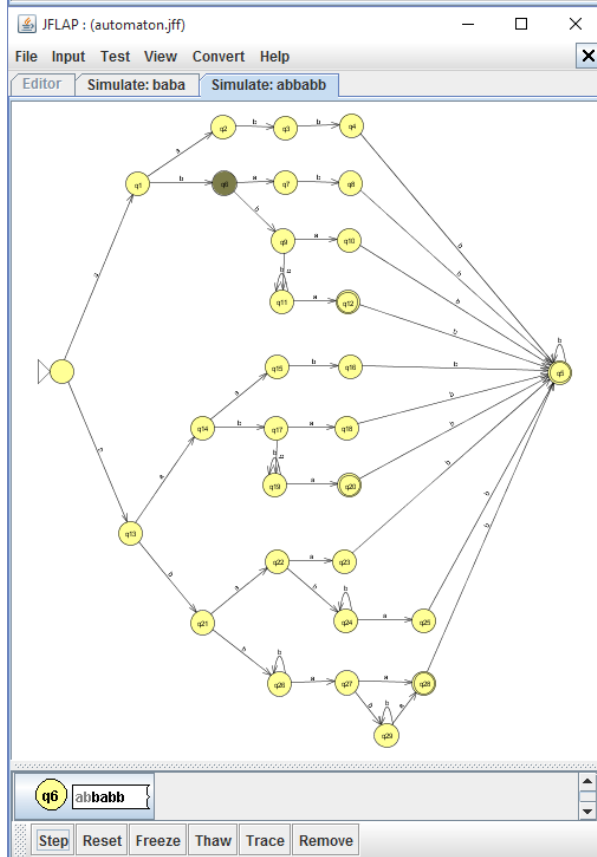
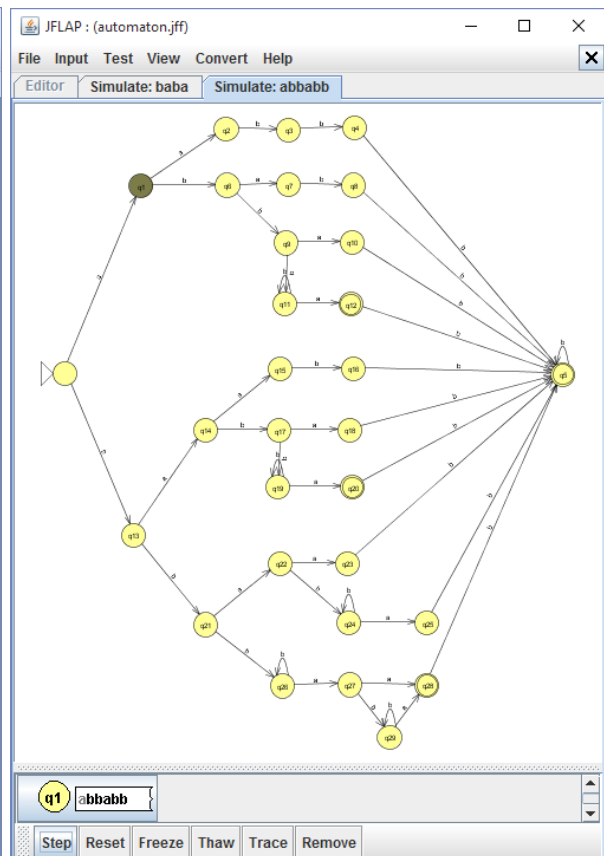
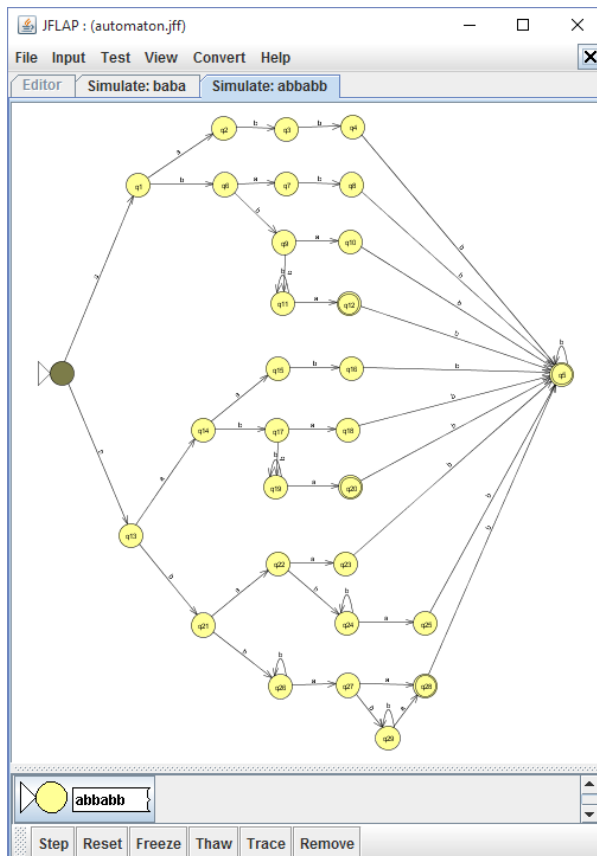
Входная строка: «baba»

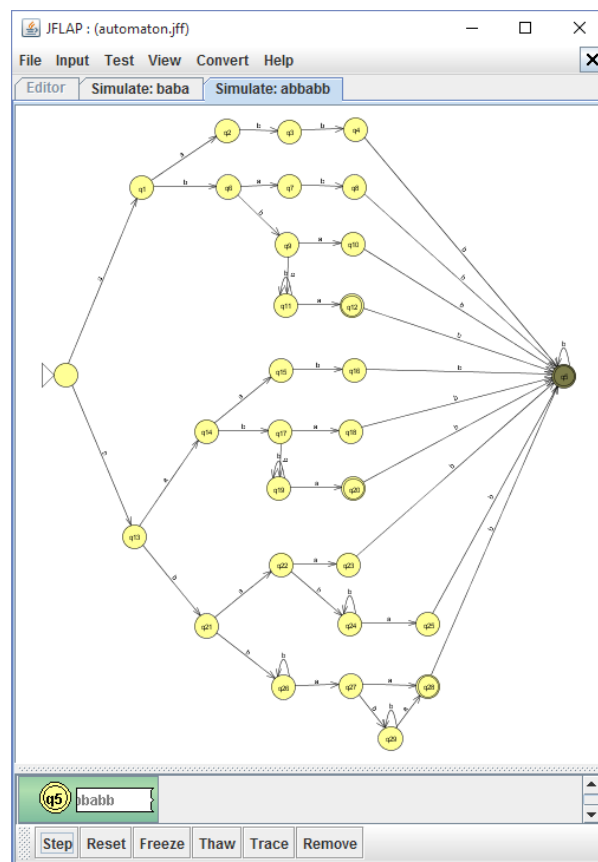
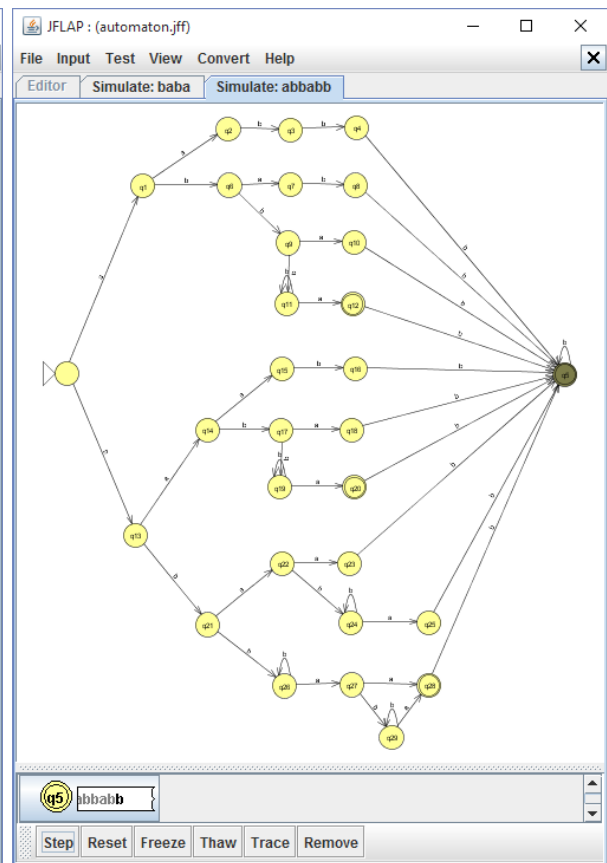
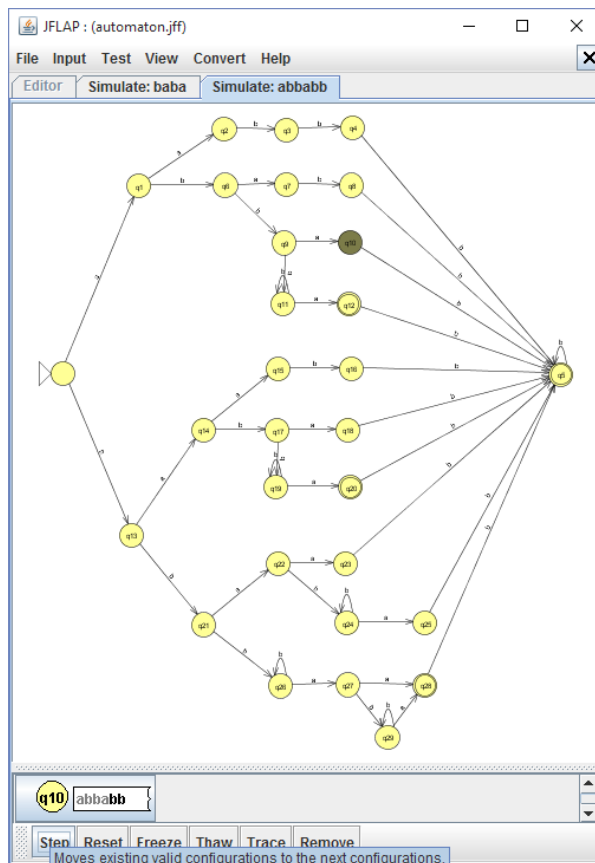




Результат – строка отвергнута.

Входная строка: «abbabb»





Результат – строка отвергнута.

Программная реализация

Исходные коды

```
int[,] transTable = new int[30, 2];

public Form1()

private void FillTransTable()
{
    transTable[0, 0] = 1;
    transTable[0, 1] = 13;
    transTable[1, 0] = 2;
    transTable[1, 1] = 6;
    transTable[2, 0] = -1;
    transTable[2, 1] = 3;
    transTable[3, 0] = -1;
    transTable[3, 1] = 4;
    transTable[4, 0] = -1;
    transTable[4, 1] = 5;
    transTable[5, 0] = -1;
    transTable[5, 1] = 5;
    transTable[6, 0] = 7;
    transTable[6, 1] = 9;
    transTable[7, 0] = -1;
    transTable[7, 1] = 8; ~
    ...
}
```

Листинг 1 – Табличное представление автомата

```
private bool IsAccepted(string text)
{
    int currentState = 0;
    bool finalState = false;

    foreach (char symbol in text)
    {
        if (symbol == 'a')
            currentState = transTable[currentState, 0];
        else if (symbol == 'b')
            currentState = transTable[currentState, 1];
        else return
            false;

        if (currentState == -1)
            return false;
    }

    if (IsFinalState(currentState))
        return true;
    return false;
}
```

Листинг 2 – Функция, возвращающая истину при правильно строке и ложь – при неправильной

```
private bool IsFinalState(int stateNumber)
{
    bool value = false;
    switch (stateNumber)
    {
        case 5:
            return true;
        case 12:
            return true;
        case 20:
            return true;
        case 28:
            return true;
        default:
            break;
    }
    return value;
}
```

Листинг 3 – Функция, возвращающая истину, если состояние под соответствующим номером является конечным, иначе – ложь

Примеры выполнения

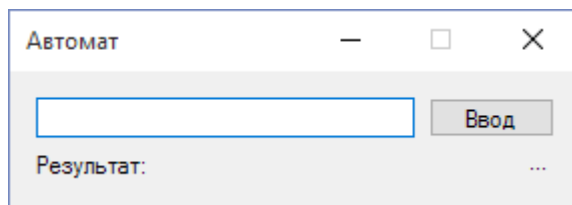


Рисунок N – Начальный интерфейс программы

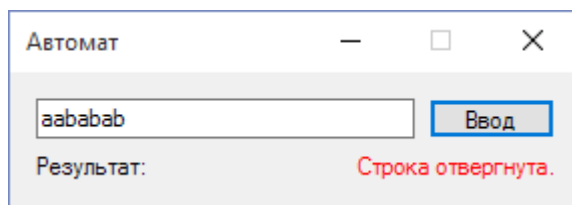


Рисунок N – Вывод отрицательного результата

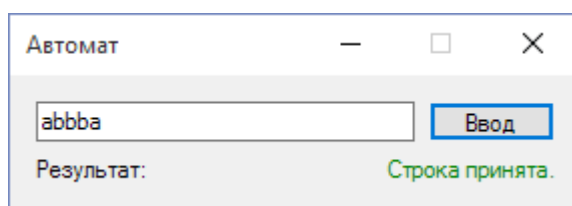


Рисунок N – Вывод положительного результата