

**杭州电子科技大学**

**《编译原理课程实践》**

**实验报告**

题 目： NFA转DFA实现

学 院： 卓越学院

专 业： 计算机科学与技术

班 级： 23186211

学 号： 23050817

姓 名： 吴锴

完成日期： 2024-10-30

1. **实验目的**

1. 掌握⾮确定有限⾃动机（NFA）与确定有限⾃动机（DFA）的基本概念及其转换⽅法。

2. 了解NFA到DFA转换过程中的⼦集构造算法。

3. 实现NFA到DFA的转换算法，并验证DFA的正确性。

4. 设计合理的数据结构，延续上⼀次实验的结构，以便为后续DFA最⼩化实验任务做好准备。

5. 提⾼编程能⼒及算法设计和优化的技能。

1. **实验内容与实验要求**

实验内容

1. 理论背景：NFA是⼀种可以处理多条路径的状态机，⽽DFA是其确定版本，不存在多条路径。通过⼦集构造算 法（Subset Construction），可以将NFA转换为等价的DFA，从⽽实现字符串匹配的确定性处理。

2. 任务描述：实现将NFA转换为DFA的算法，并对转换后的DFA进⾏验证。同时，设计适合DFA的数据结构，使 其兼容前⼀次实验的NFA数据结构。

3. 实验步骤 ： 理解⼦集构造算法的原理，包括ε-闭包的计算和状态集合的映射。 利⽤⼦集构造算法，将NFA转换为DFA。 设计并实现DFA的数据结构，确保其能够表示状态集合、状态转换、初始状态和接受状态。 验证DFA的正确性，对⽐DFA与NFA在同⼀组测试输⼊上的匹配结果。

实验要求：

1.输⼊输出要求

输⼊：⼀个NFA（包括状态集、转换表、初始状态和接受状态集合）和多个测试字符串。 输出：⽣成的DFA状态集合及其转换关系，指明每个测试字符串是否被DFA接受。

2.算法要求

实现⼦集构造算法，将NFA状态集合的⼦集映射为DFA的单个状态。 处理ε-闭包及其状态转换，⽣成对应的DFA。

3.数据结构要求

在上⼀实验的基础上，设计DFA的数据结构，包含状态集合、转换关系、初始状态和接受状态集合的表 示。 确保数据结构可以⽀持后续的DFA最⼩化任务，便于后续实验任务的延续。

4.程序要求

使⽤C/C++、Java、Python等语⾔编写程序，代码结构清晰，具备良好的注释。 提供详细的实验报告，包括算法设计、实现过程、测试结果和问题分析。

5.实验报告要求

【整合到最后提交的个⼈所有实验报告中，加上⽬录】 描述实验⽬的和内容。 解释⼦集构造算法的原理、步骤和数据结构的设计思路。 给出测试⽤例和结果，分析测试数据的正确性。 总结实验的收获和遇到的挑战。

1. **设计方案与算法描述**

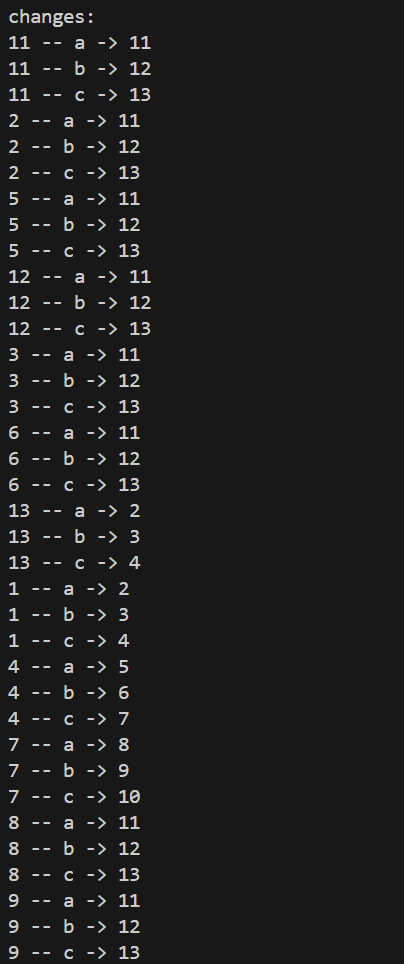
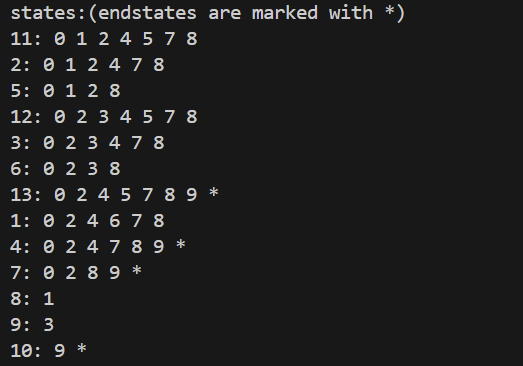
由于状态的唯一性，故而采用c++标准库的set数据类型，用于存储每个状态中的节点，继承之前NFA的设计思路，采用c++标准库的map来保存状态之间的转换，也便于之后最小化DFA时对其进行查找。

算法上模拟实现子集构造法，将NFA状态集合的子集映射为DFA的单个状态，处理ε-closure闭包及其状态转换，生成对应的DFA。

1. **测试结果**

使用上节实验课设计的REX2NFA，将(a|b)\*c转换为NFA后，将NFA输入程序，通过本次课设计的NFA2DFA程序，得到的NFA如下：

1.输入的NFA： 2.得到的结果：

****(a|b)\*c

6 9

0 1 a

1 5 E

2 3 b

3 5 E

4 0 E

4 2 E

5 4 E

5 7 E

6 4 E

6 7 E

7 8 E

8 9 c

1. **源代码**

见附录文件夹 NFA2DFA