# 编译原理第二次作业

#### 3.8.4

#### 题目:

设计一个算法来识别形如 r1/r2 的 Lex 向前看模式, 其中 r1 和 r2 都是正则表达式。说明该算法如何处理如下输入:

- 1) (abcd|abc)/d
- 2) (a|ab)/ba
- 3) aa\*/a\*

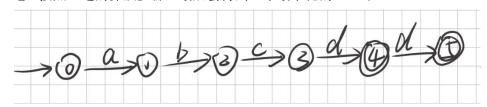
# 解答:

#### 算法步骤

- ① 从输入字符串开头开始扫描,直到找到第一个"/"字符;
- ② 将"/"前面部分作为 r1,后面部分作为 r2;
- ③ 分别对 r1 和 r2 进行正则表达式解析,得到对应的 NFA;
- ④ 对于 r1 的 NFA 在每个终止状态添加一条 ε 的转移边到 r2 的起始状态;
- ⑤ 将新 NFA 转换成 DFA (可以通过子集构造法),并且进行最小化处理;
- ⑥ 输出最小化 DFA。
- ⑦ 对最小化的 DFA 进行词法分析
  - a) 初始化当前的状态为起始状态
  - b) 逐个读取输入字符,并根据当前状态和输入字符进行状态转移
  - c) 如果在某个状态下无法进行状态转移,则识别失败,表示输入字符不符合正则 表达式定义;
  - d) 如果读取完所有输入字符后,当前的状态为接受状态,则识别成功,表示符合 该模式下正则表达式的定义

### 1) (abcd|abc)/d

- ① 对于表达式按照"/"进行分割,得到子表达式 r1 = "abcd|abc" 和 r2 = "d"
- ② 按照上述的算法步骤,最后会得到一个最小化的 DFA 即

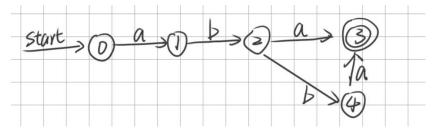


③ 读取输入字符串,并按照上述最小 DFA 进行状态转移,如果在某个状态下无法转移匹配失败;

如果在读取完所有输入字符串为接受状态则匹配成功。

#### 2) (a|ab)/ba

- ① 对于表达式按照 "/" 进行分割,得到子表达式 r1 = "alab" 和 r2 = "ba"
- ② 按照上述的算法步骤,最后会得到一个最小化的 DFA 即

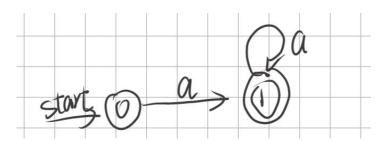


③ 读取输入字符串,并按照上述最小 DFA 进行状态转移,如果在某个状态下无法转移匹配失败;

如果在读取完所有输入字符串为接受状态则匹配成功。

### 3) aa\*/a\*

- ① 对于表达式按照"/"进行分割,得到子表达式 r1 = "aa\*" 和 r2 = "a\*"
- ② 按照上述的算法步骤,最后会得到一个最小化的 DFA 即



③ 读取输入字符串,并按照上述最小 DFA 进行状态转移,如果在某个状态下无法转移匹配失败;

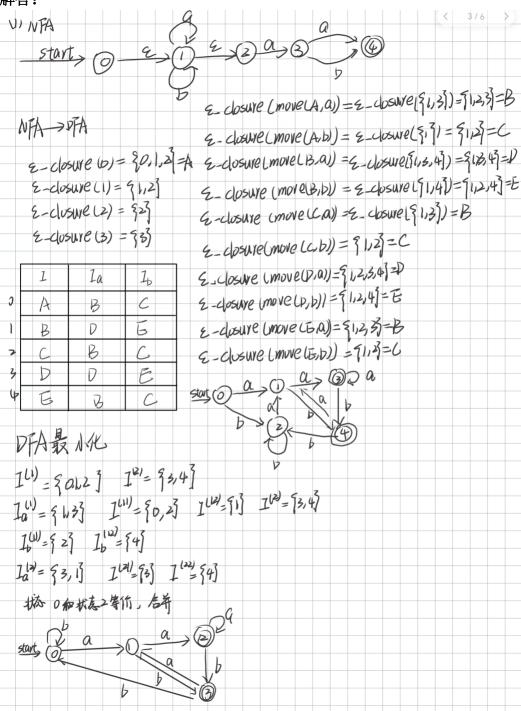
如果在读取完所有输入字符串为接受状态则匹配成功。

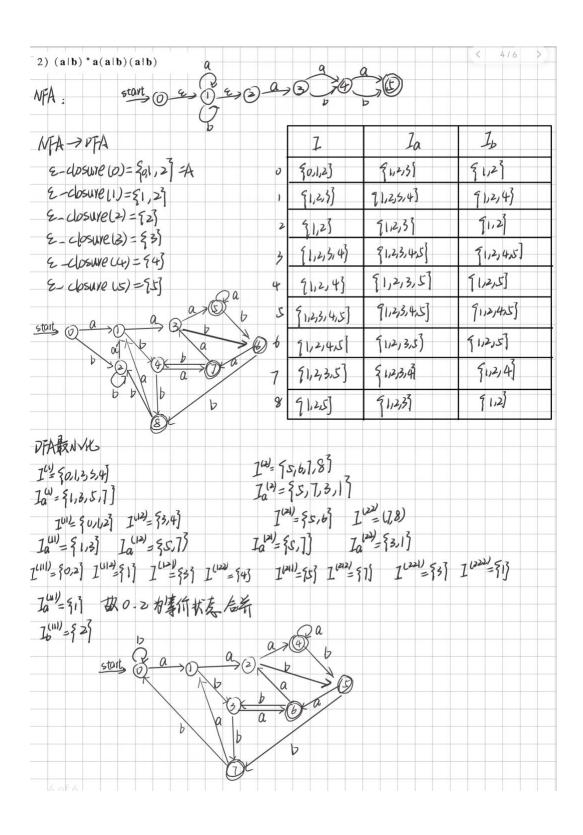
#### 3.9.4

### 题目:

- ! 练习 3. 9. 4: 为下列的正则表达式构造最少状态 DFA:
- -1) (a|b) \*a(a|b)
- (a|b) \* a(a|b)(a|b)
- 3) (a|b) \* a(a|b) (a|b) (a|b)

## 解答:





| 3) (a b) *a(a b)(a b)                      | (a b)    |               |     |               | < 5/             |
|--|----------|---------------|-----|---------------|------------------|
| NFA: a                                     |          | a a           |     | a .           |                  |
| start (0) 2 1 2                            | >@ a>    | 3307          | (E) |               |                  |
|  |          | Ь             |     |               |                  |
| Jeh Th                                     |          |               |     |               |                  |
| NFA->DFA                                   | <u> </u> | 1             |     | Ia            | I <sub>b</sub>   |
| (2_dosuve 10)={0,1,2}                      | 0        | 80,1,2]       | 1   | {1,2,9}       | 2 91,2]          |
| 2_closure (1)={1,2}                        | +++      | 91,2,5]       | 3   | {1,2,5,4}     | 4 912,43         |
| - closure (2) = 923                        | 2        | {1,2]         | ,   | 91,2,3]       | 2 31,2]          |
| E-closure (3) = 53)                        | 3        | 31,2,3,4]     | 5   | {1,2,3,4x}    | 6 {1,2,4,5}      |
| E-doswe (4)={4}                            | 4        | {1,2,4}       | 7   | {1,2,3,5}     | 8 {1,2,5}        |
| &-closure (3) = \$3}<br>&-closure (6)= {6} | S        | [1,2,3,4,5]   | 9   | 31,2,3,4,5,63 | 10 {1,2,4,5,6}   |
| 2- Closule (6)=10)                         | 6        | 31,2,4,53     | ıı  | \$1,2,3,5,6}  | 12 \$ 1,2,5,63   |
|  |          | \$1,2,3,5)    | 13  | \$1,213,4,6]  |                  |
|  | 8        | 31,2,5]       | IS  | {1,2,3,6}     | 16 \$1,2,63      |
|  | 9        | {1,2,3,4,5,6} | 9   | {1,2,3,4,5,6} | 10 \$ 1,2,4,5,63 |
|  | 10       | 31,2,4,5,63   | 1/1 | {1,2,3,5,6}   | 12 \$ 1,25,63    |
|  | 1        | 312,3,5,6]    | 13  | 31,2,3,4,6]   | 14 {1,2,4,6}     |
|  | 12       | ξ1,2,5,b]     | ľ   | {1,2,3,6]     | 16 91,2,63       |
|  | 13       | 31,2,3,4,6)   | 5   |               | b {124,5}        |
|  | 14       | \$1,2,4,6}    | 1   | §1,2,3,5}     | 8 {1,2,5}        |
|  | ß        | 31,2,3,65     | 3   | ₹1,2,3,4}     | 4 81,2,43        |
|  | 16       | ₹1,2,6°}      | 1   | 1123)         | 2 7627           |
|  | -10_     |               | 1   |               |                  |
| 所最N化                                       |          |               |     |               |                  |

