

Required Exercise

Exercise 1

When a C compiler compiles the following statement, how many tokens will it generate? [5 points]

```
int a3 = a * 3;
```

将会生成以下7种tokens：

- <keyword, int>
- <id, a3>
- <id, a>
- <assign, =>
- <assign, *>
- <assign, ;>
- <number, 3>

或者回答"int"、"a3"、"a"、"="、"*"、";"、"3"也可。

Exercise 2

In a string of length n ($n > 0$), how many of the following are there?

1. Prefixes [5 points]
2. Proper prefixes [5 points]
3. Prefixes of length m ($0 < m \leq n$) [5 points]
4. Suffixes of length m ($0 < m \leq n$) [5 points]
5. Proper prefixes of length m ($0 < m \leq n$) [10 points]
6. Substrings [10 points]
7. Subsequences [10 points]

1. $n+1$

从串首取长度为1, 2, 3, ..., n 的字符, 有 n 个prefixes, 加上空串, 共有 $n+1$ 个prefixes。

2. $n-1$

从串的prefixes中去除空串以及串本身, 有 $n-1$ 个proper prefixes。

3. 1

从串首取 m 个字符, 只能形成一种prefix

4. 1

从串首去掉 m 个字符, 剩余的字符只能形成一种suffix

$$5. res = \begin{cases} 0, & m = n \\ 1, & \text{otherwise} \end{cases}$$

当m = n时，不存在长度与串本身长度相等的proper prefix

其余情况下，长度为m的proper prefix个数为1

$$6. \frac{n(n+1)}{2} + 1 \text{ 或 } C(n+1, 2)$$

包含1个字符的子串有n个，包含2个字符的子串有n-1个，包含3个字符的子串有n-2个，...，包含n个字符的子串有1个，包含0个字符的子串有1个。总个数为 $1+2+3+\dots+n+1 = \frac{n(n+1)}{2} + 1$

$$7. 2^n$$

将问题转化为每一个字符是否被选中，总个数为 2^n

Exercise 3

Describe the languages denoted by the following regular expressions:

1. $((\epsilon \mid a)^* b^*)^*$ [5 points]
2. $(a \mid b)^* a (a \mid b) (a \mid b)$ [5 points]
3. $a^* b a^* b a^*$ [5 points]

1. 由a和b组成的任意长度的串
2. 倒数第三个符号为a的由a和b组成的任意长度的串
3. 由任意个数的a和三个b组成的串

(英文版)

1. A string consisting of a and b
2. String of a's and b's that the character third from the last is a
3. A string consisting of any number of a and three b's

Exercise 4

Write regular definitions or regular expressions for the following languages.

1. All strings representing valid telephone numbers in Shenzhen. A valid telephone number contains the country code (86), a hyphen, the area code 010, another hyphen, and eight digits where the first one cannot be zero (e.g., 86-010-62282045). [10 points]
2. All strings of a's and b's that start with a and end with b. [10 points]
3. All strings of lowercase letters that contain the five vowels in order. [10 points]

1. 86-010-[1-9][0-9]{7}

2. a[ab]*b 或 a(a|b)*b

3. 思路一：元音字母每个只能出现一次

pattern $\rightarrow [b-df-hj-np-tv-z]$

want $\rightarrow \text{pattern}^* a \text{ pattern}^* e \text{ pattern}^* i \text{ pattern}^* o \text{ pattern}^* u \text{ pattern}^*$

思路二：元音字母可出现多次

pattern $\rightarrow [b-df-hj-np-tv-z]$

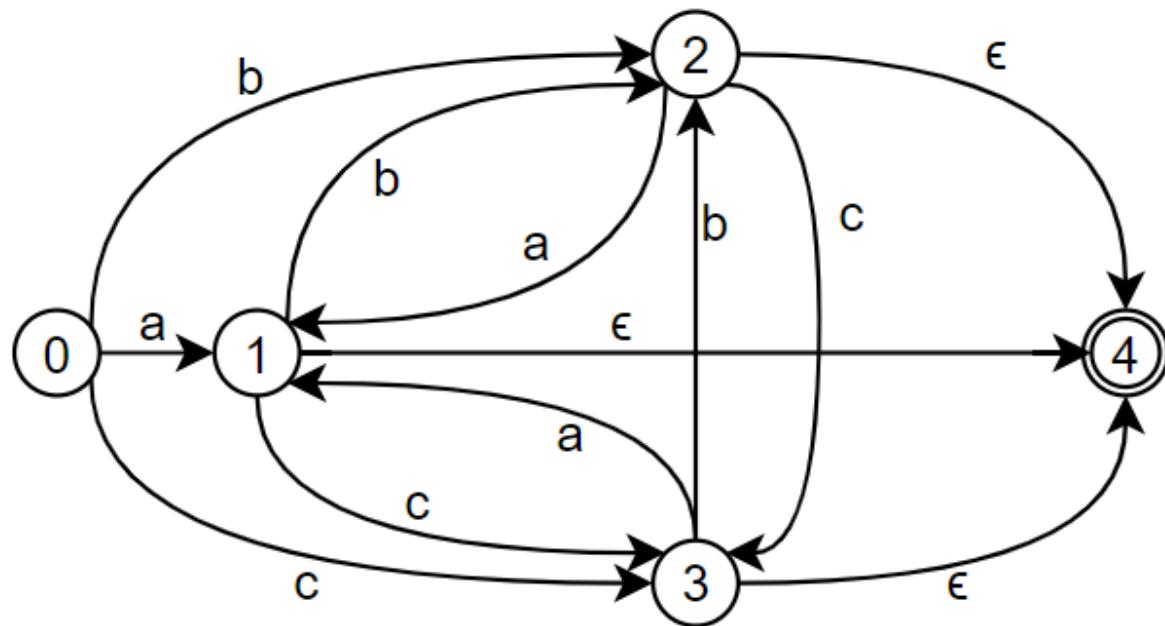
want $\rightarrow \text{pattern}^* a (\text{pattern}|a)^* e (\text{pattern}|e)^* i (\text{pattern}|i)^* o (\text{pattern}|o)^* u (\text{pattern}|u)^*$

Optional Exercises (10 bonus points)

Exercise 1

Suppose we have a alphabet $\Sigma = a, b, c$, write regular definitions to describe all strings over Σ without repeated letters. [Hint: You may draw an NFA for the language and convert the NFA to regular definitions

NFA如下图所示

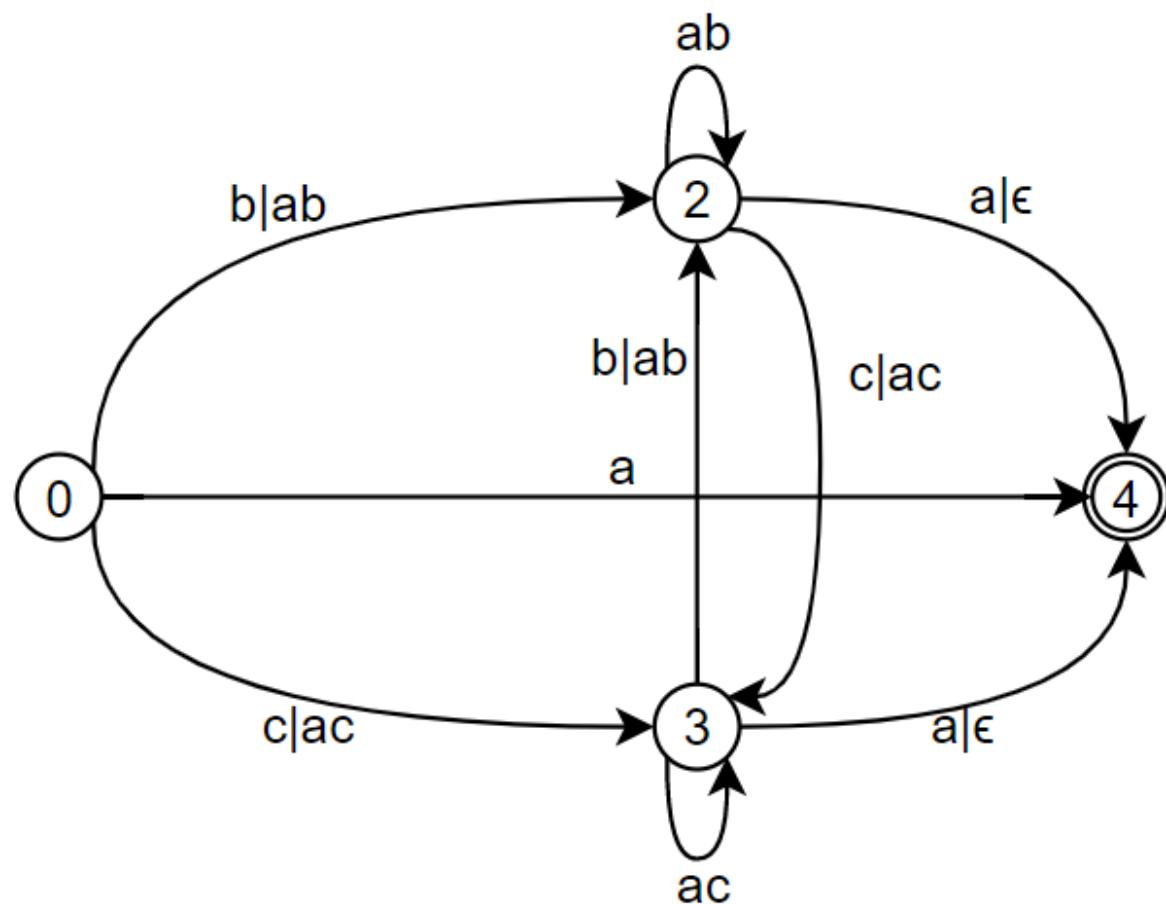


然后通过去除状态1、2、3来简化NFA

去除状态1

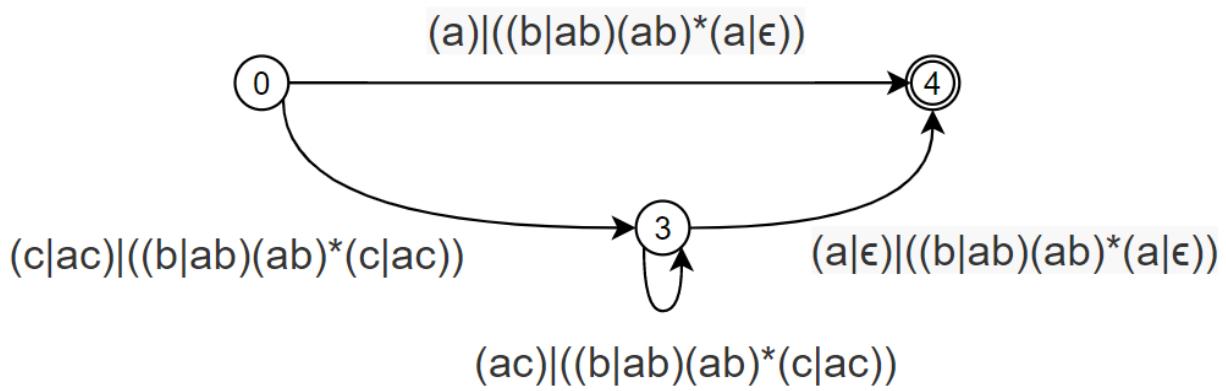
edge	regular expression
0 \rightarrow 2	b ab
0 \rightarrow 3	c ac

edge	regular expression
0 -> 4	a
2 -> 2	ab
2 -> 3	c ac
2 -> 4	a ε
3 -> 2	b ab
3 -> 3	ac
3 -> 4	a ε



去除状态2

edge	regular expression
0 -> 3	(c ac) ((b ab)(ab)* (c ac))
0 -> 4	(a) ((b ab)(ab)*(a ε))
3 -> 3	(ac) ((b ab)(ab)*(c ac))
3 -> 4	(a ε) ((b ab)(ab)*(a ε))



去除状态3

edge	regular expression
0 -> 4	((a) ((b ab)(ab)(a)) (((c ac) (b ab)(ab)(c ac)) (ac) ((b ab)(ab)(c ac)) (a) ((b ab)(ab) * (a))))

将简化后的NFA转为正则表达式的结果为

$((a)|((b|ab)(ab)(a|))|((c|ac)|(b|ab)(ab)(c|ac))|(ac)|((b|ab)(ab)(c|ac))|(a|)|((b|ab)(ab)^*(a|))))$

regex101结果验证

<https://regex101.com/r/5C0PEp/1>