

3.3.2, 3.3.6, 3.3.7, 3.3.8, 3.3.9,
3.6.3, 3.6.4, 3.6.5
3.7.1, 3.7.2, 3.7.3
3.8.1, 3.8.2
3.9.3

3.3.2 描述下列正则表达式代表的语言。

- a) $a(a|b)^*a$
- b) $((\epsilon|a)b^*)^*$
- c) $(a|b)^*a(a|b)(a|b)$
- d) $a^*ba^*ba^*ba^*$
- e) $(aa|bb)^*((ab|ba)(aa|bb)^*(ab|ba)(aa|bb)^*)^*$

答案

- (a) 由 a 开头并结尾的由 a 和 b 构成的字符串
- (b) 由 a 和 b 构成的字符串
- (c) 倒数第三位为 a 的由 a 和 b 构成的字符串
- (d) 仅含 3 个 b 的由 a 和 b 构成的字符串
- (e) 含有偶数个 a 和偶数个 b 的由 a 和 b 构成的字符串

注意：请准确描述语言的性质而不是列举满足正则表达式的串

3.3.6 写出满足下列定义的字符

- a) The first ten letters in either upper or lower case
- b) The lowercase consonants
- c) The “digits” in a hexadecimal number
- d) The characters that can appear at the end of a legitimate English sentence

答案

- (a) a-zA-J
- (b) a-j
- (c) 0-9a-f
- (d) .?!

3.3.7 写出匹配字符串 “\” 的正则表达式

答案：

\backslash

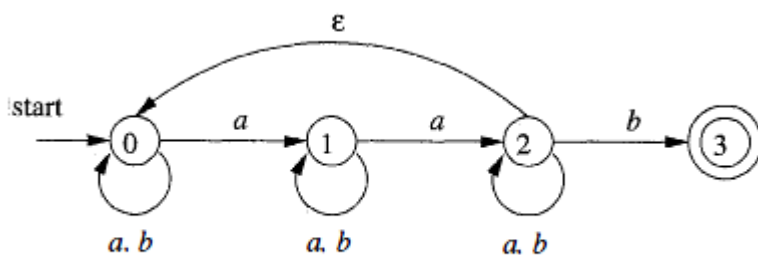


Figure 3.29: NFA for Exercise 3.6.3

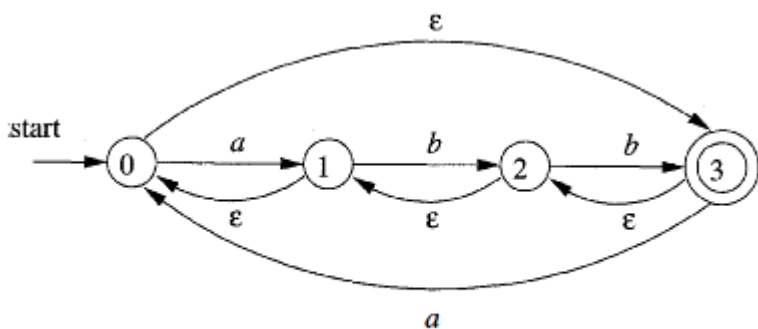


Figure 3.30: NFA for Exercise 3.6.4

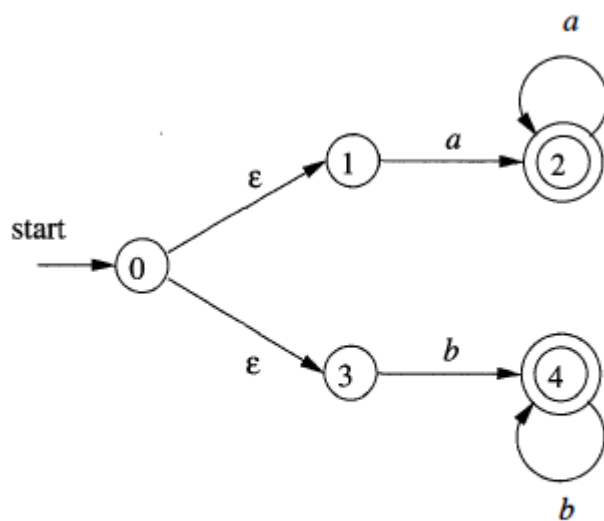


Figure 3.26: NFA accepting $aa^*|bb^*$

3.6.3 对于图 3.29 表示的 NFA,列出 aabb 的所有路径。这个 NFA 能否接受 aabb?

答案:

aabb 的所有路径

01223 001111 012000 000000 01222 00011 00123

存在路径 1223 和 0123 所以能接受 aabb

3.6.4 对于图 3.30 表示的 NFA,列出 aabb 的所有路径。这个 NFA 能否接受 aabb?

答案:

010123
 0101212
 030123
 0301212
 030303232
 0303032123
 03030321212

由于存在 03210 这样的环,所以这里有无数种路径
 存在路径终止于 3,所以能接受 aabb

3.6.5 给出以下 NFA 的 Transition Table

(a) 图 3.29

(b) 图 3.30

(c) 图 3.26

答案:

(a)

State	a	b	e
0	{0,1}	{0}	空集
1	{1,2}	{1}	空集
2	{2}	{2,3}	{0}
3	空集	空集	空集

(b)

State	a	b	e
0	{1}	空集	{3}
1	空集	{2}	{0}
2	空集	{3}	{1}
3	{0}	空集	{2}

(c)

State	a	b	e
0	空集	空集	{1,3}
1	{2}	空集	空集
2	{2}	空集	空集
3	空集	{4}	空集
4	空集	{4}	空集

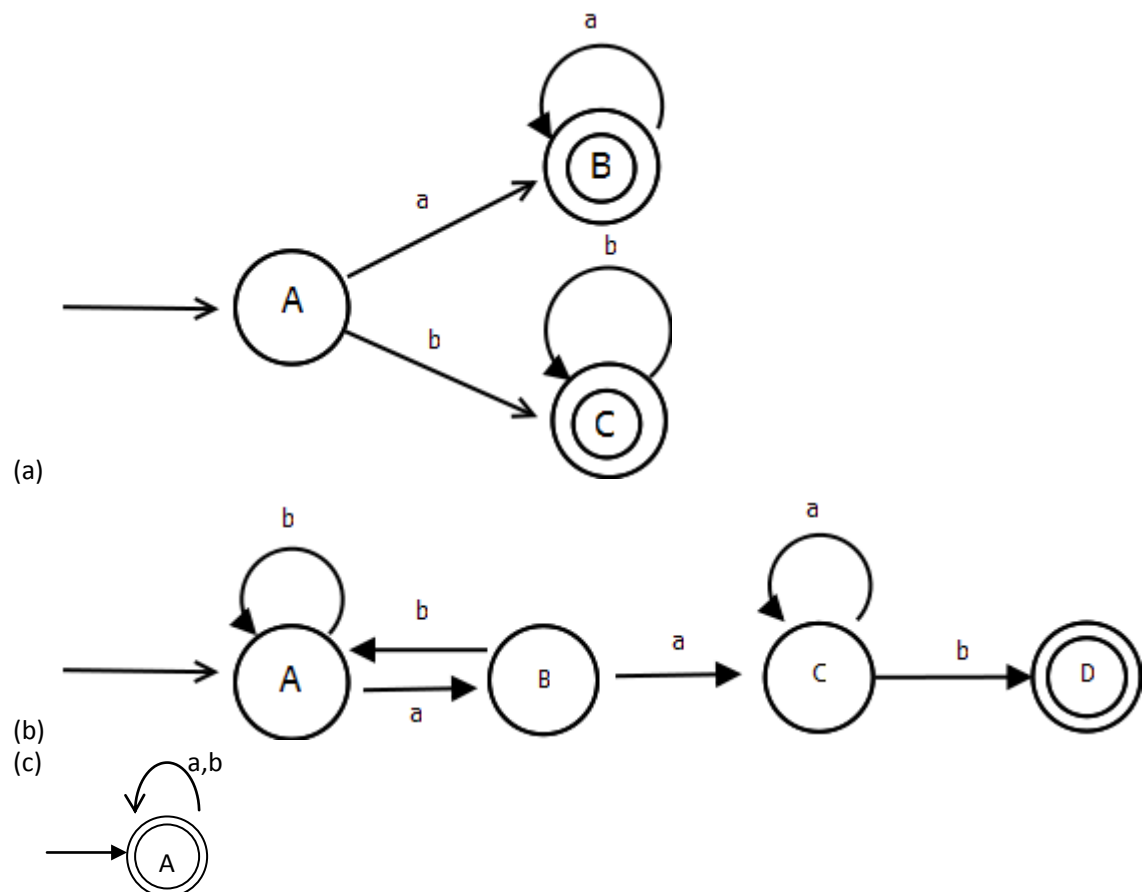
3.7.1 把下列 NFA 转化为 DFA

(a)图 3.26

(b)图 3.29

(c)图 3.30

答案:



注意:以上答案并不唯一,等价即可

3.7.2 用算法 3.22 模拟 NFA(输入为 aabb)

(a)图 3.29

(b)图 3.30

```

1)  $S = \epsilon\text{-closure}(s_0);$ 
2)  $c = \text{nextChar}();$ 
3) while (  $c \neq \text{eof}$  ) {
4)      $S = \epsilon\text{-closure}(\text{move}(S, c));$ 
5)      $c = \text{nextChar}();$ 
6) }
7) if (  $S \cap F \neq \emptyset$  ) return "yes";
8) else return "no";
  
```

Figure 3.37: Simulating an NFA

答案:

(a)

S	nextChar
{0}	a

{0,1}	a
{0,1,2}	b
{0,1,2,3}	b
{0,1,2,3}	eof

F={3} 所以返回 yes

(b)

S	nextChar
{0,1,2,3}	a
{0,1,2,3}	a
{0,1,2,3}	b
{0,1,2,3}	b
{0,1,2,3}	eof

F={3},所以返回 yes

3.7.3 用算法 3.23 和 3.20 把下列正则表达式转换为 DFA

a) (alb)*

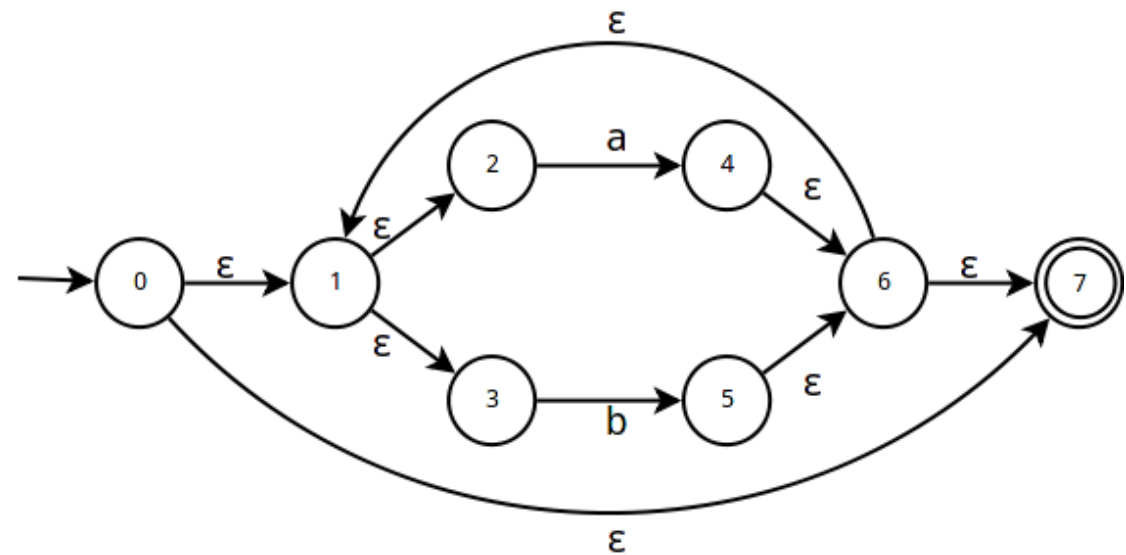
b) (a*lb*)*

c) ((ela)b*)*

d) (alb)*abb(alb)*

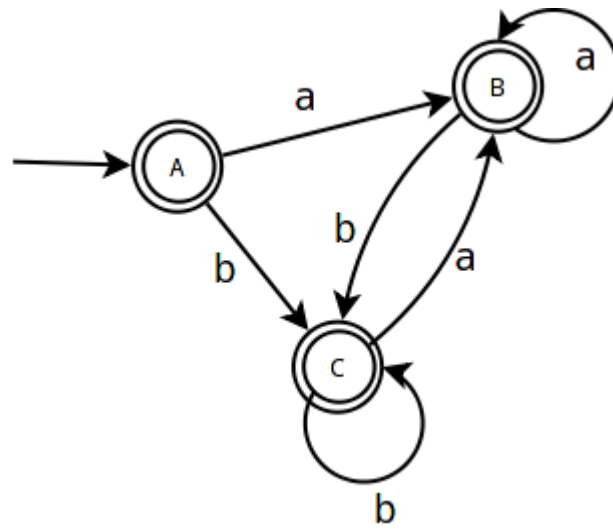
答案:

a) NFA

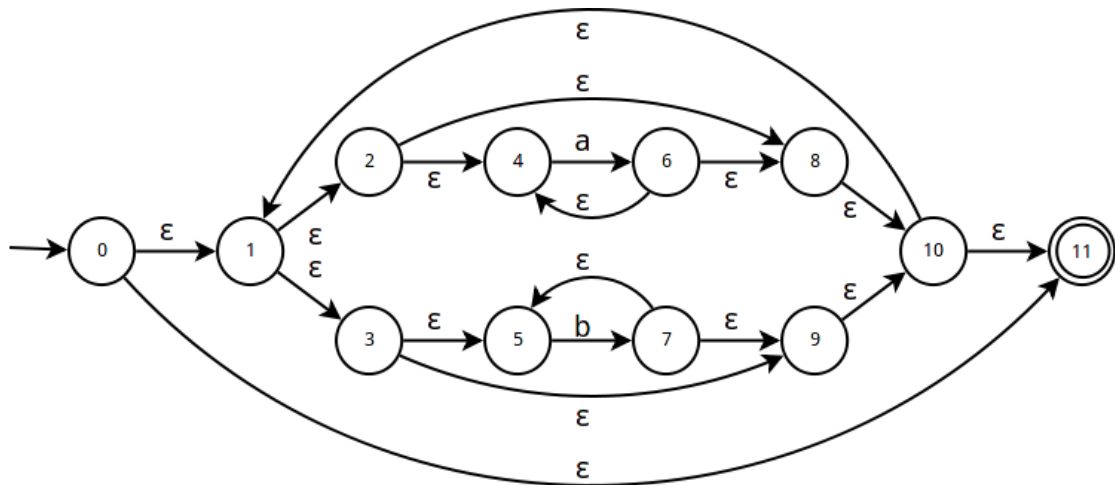


DFA

NFA State	DFA State	a	b
{0,1,2,3,7}	A	B	C
{1,2,3,4,6,7}	B	B	C
{1,2,3,5,6,7}	C	B	C

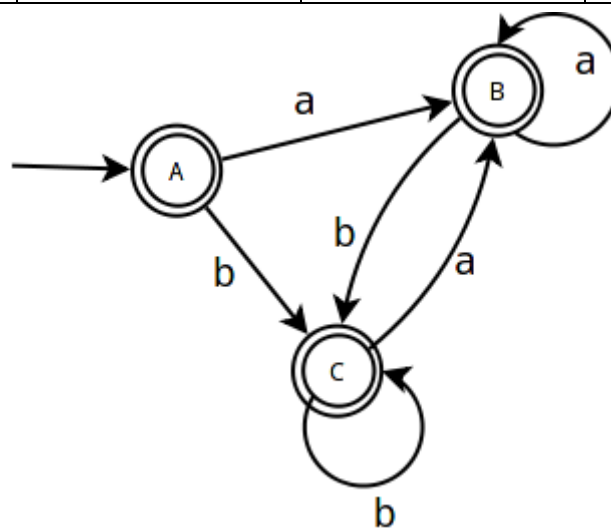


b) NFA

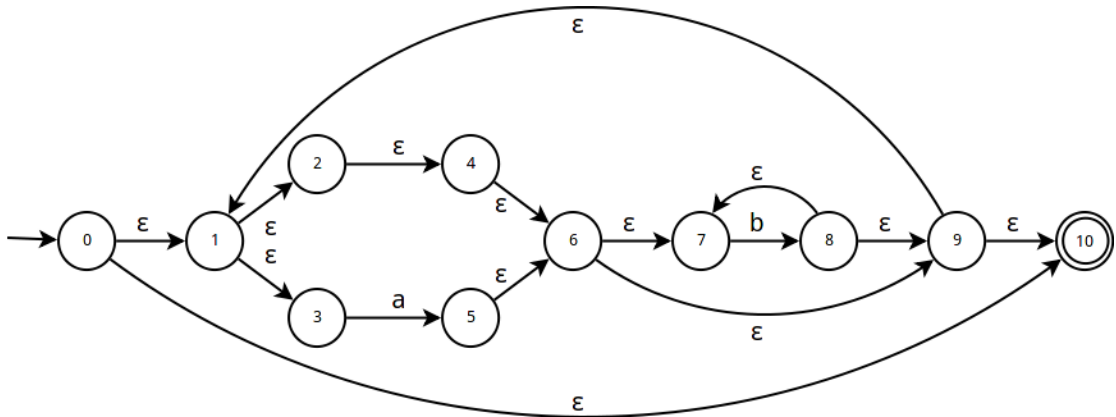


DFA

NFA State	DFA State	a	b
{0,1,2,3,4,5,8,9,10,11}	A	B	C
{1,2,3,4,5,6,8,9,10,11}	B	B	C
{1,2,3,4,5,7,8,9,10,11}	C	B	C

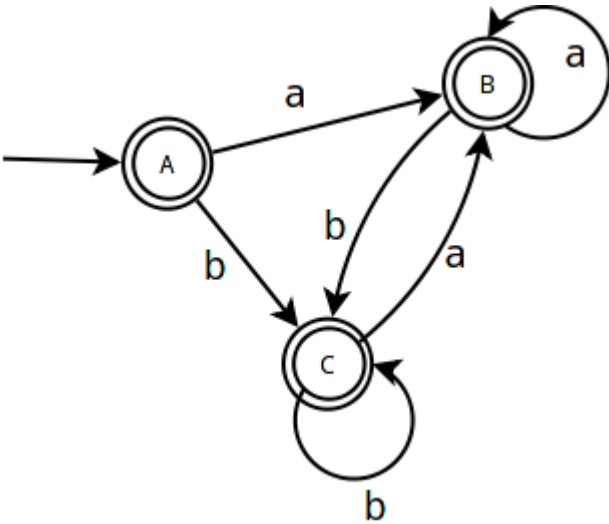


c) NFA

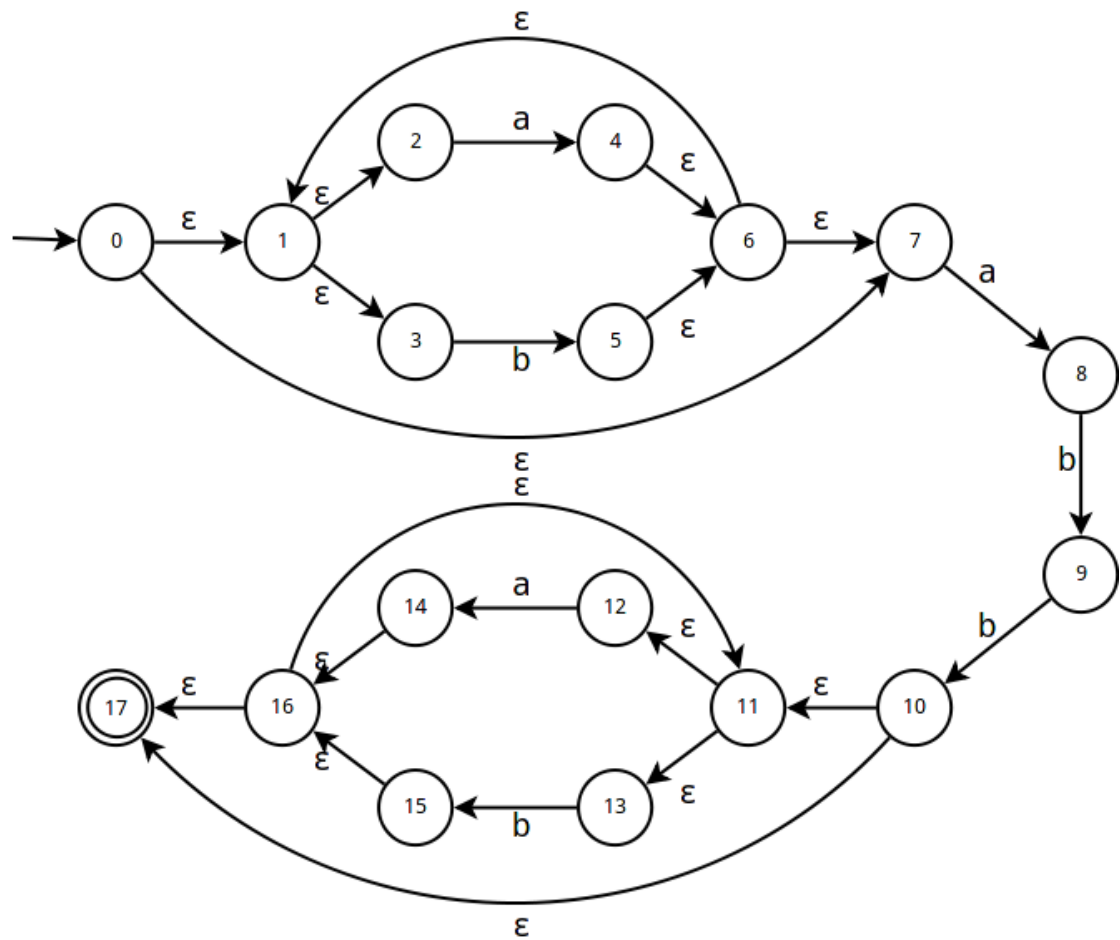


DFA

NFA State	DFA State	a	b
{0,1,2,3,4,6,7,9,10}	A	B	C
{1,2,3,4,5,6,7,9,10}	B	B	C
{1,2,3,4,6,7,8,9,10}	C	B	C

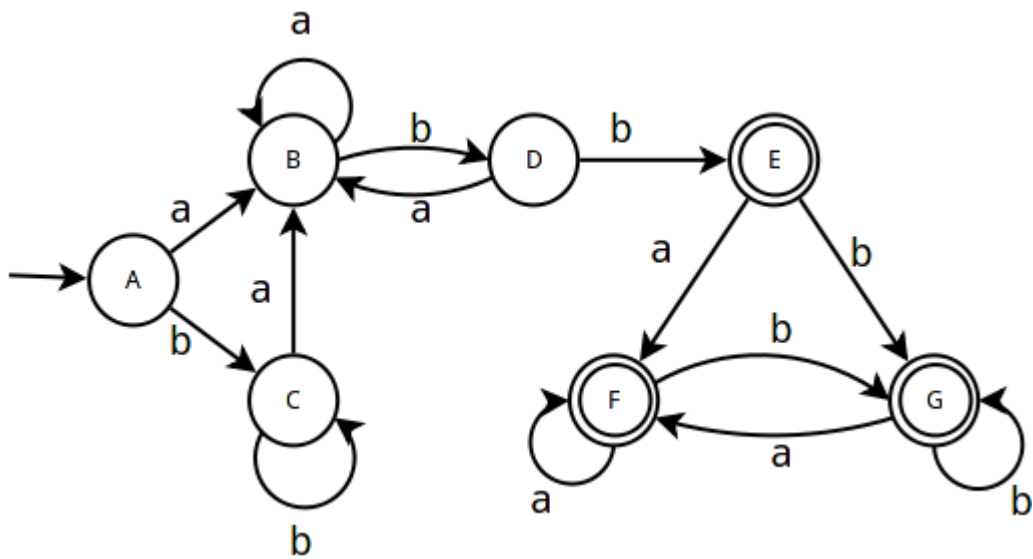


d) NFA



DFA

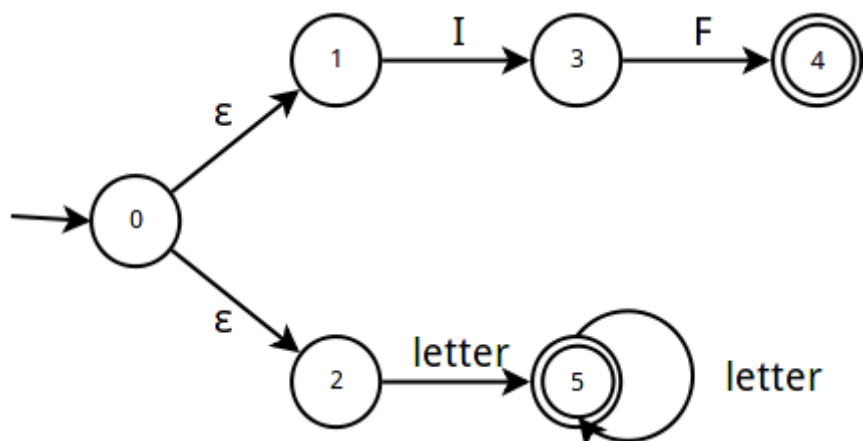
NFA State	DFA State	a	b
{0,1,2,3,7}	A	B	C
{1,2,3,4,6,7,8}	B	B	D
{1,2,3,5,6,7}	C	B	C
{1,2,3,5,6,7,9}	D	B	E
{1,2,3,5,6,7,10,11,12,13,17}	E	F	G
{1,2,3,4,6,7,8,11,12,13,14,16,17}	F	F	G
{1,2,3,5,6,7,11,12,13,15,16,17}	G	F	G



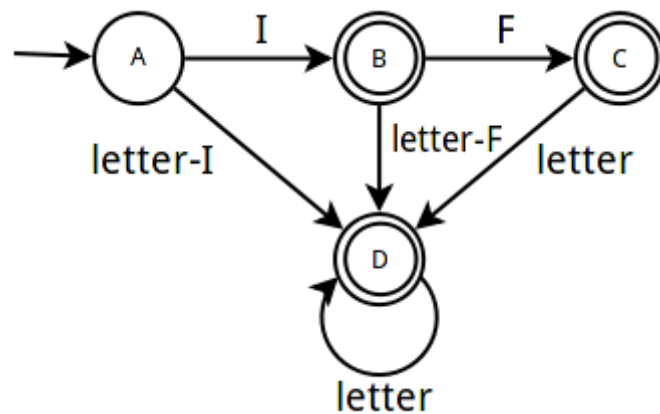
注意：这道题要求大家按照算法构造 NFA 和 DFA，有些同学的 NFA 没有完全按照算法构造。另外，画 NFA 和 DFA 时务必标出开始状态和终止状态。

3.8.1

NFA



DFA



3.8.2 与 3.8.1 类似，请耐心构造相应的 NFA 和 DFA

3.8.3 这 3 个正则表达式对应的最小 DFA 都与如下的 DFA 同构，所以它们等价。

