

# 《编译原理》习题参考答案(一)

Bug report: [zpli@mail.ustc.edu.cn](mailto:zpli@mail.ustc.edu.cn)

Or find: 电一楼二楼全球计算实验室 李兆鹏

## 第二章

2.3 叙述由下列正规式描述的语言

a)  $0(0|1)^*0$

b)  $((\epsilon|0)1^*)^*$

c)  $(0|1)^*0(0|1)(0|1)$

d)  $0^*10^*10^*10^*$

e)  $(00|11)^*((01|10)(00|11)^*(01|10)(00|11)^*)^*$

Answer:

- a) 以 0 开始和结尾, 而且长度大于等于 2 的 0、1 串
- b) 所有 0,1 串 (含空串)
- c) 倒数第三位是 0 的 0、1 串
- d) 仅含 3 个 1 的 0、1 串
- e) 偶数个 0 和偶数个 1 的 0、1 串 (含空串)

2.4 为下列语言写出正规定义:

f) 由偶数个 0 和偶数个 1 构成的所有 0 和 1 的串

g) 由偶数个 0 和奇数个 1 构成的所有 0 和 1 的串

Answer:

标准答案见《编译原理习题精选》P1-P2 1.1&1.2 题

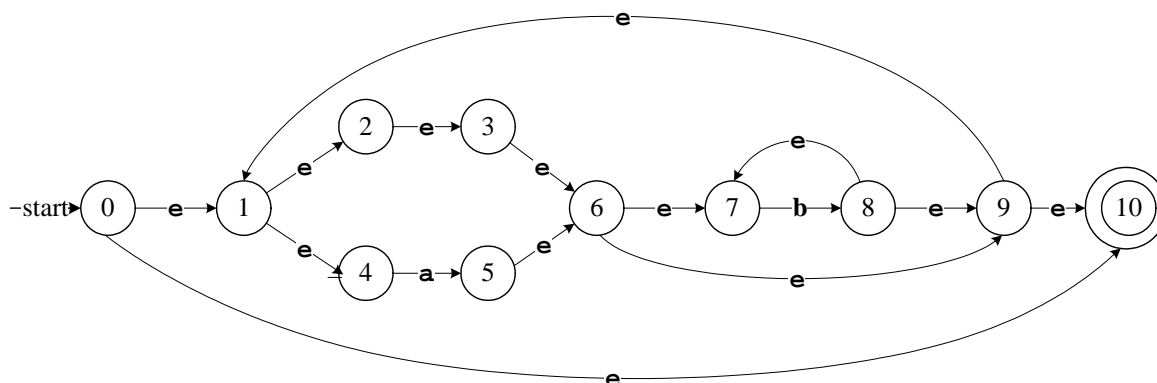
2.7 用算法 2.4 为下列正规式构造非确定的有限自动机, 给出它们处理输入串 ababbab 的转换序列。

c)  $((\epsilon|a) b^*)^*$

d)  $(a|b)^*abb(a|b)^*$

Answer:

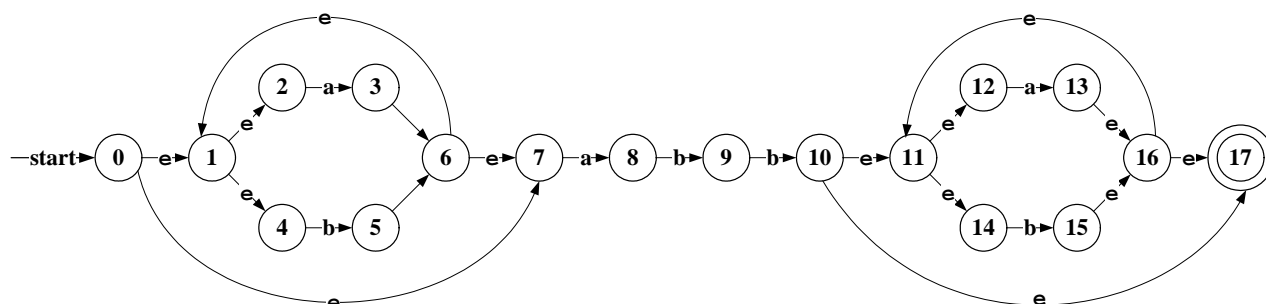
c) NFA:



输入串 ababbab 的转换序列:

0 1456789 145678 789 1456789 10  
Or 0 1456789 1456789 1236789 1456789 10

d) NFA:



输入串 ababbab 的转换序列:

0 1236 1456 789 10 11 12 13 16 11 14 15 16 17

2.8 用算法 2.2 把习题 2.7 的 NFA 转换成 DFA。给出它们处理输入串 ababbab 的状态转换序列。

**Answer:** //针对 2.7 (c)

3 个不同的状态集合:

A = {0, 1, 2, 3, 4, 6, 7, 9, 10}

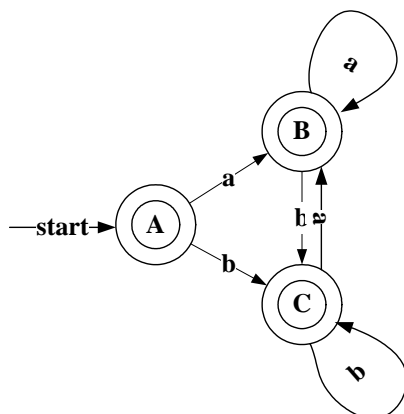
B = {1, 2, 3, 4, 5, 6, 7, 9, 10}

C = {1, 2, 3, 4, 6, 7, 8, 9, 10}

**NFA 的转换表:**

状态	输入符号	
	a	b
A	B	C
B	B	C
C	B	C

子集构造法应用于 2.7(c) 得 DFA:

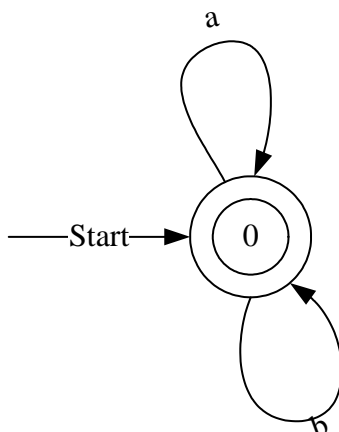


2.11 我们可以从正规式的最简 DFA 同构来证明两个正规式等价。使用这种技术，证明下面正规式等价。

- (a)  $(a|b)^*$
- (b)  $(a^*|b^*)^*$
- (c)  $((\epsilon|a)b^*)^*$

**Answer:**

(c) NFA 见 2.7 (c) 用子集构造法化简得 DFA 见 2.8。  
最简的 DFA 为：



同理可求出 (a) (b) 正规式对应的最简 DFA 亦是如此，故 (a) (b) (c) 等价。

2.15 修改算法 2.4, 使之尽可能少使用  $\epsilon$  转换，并保持所产生的 NFA 只有一个接收状态。

**Answer:**

算法 2.4 改进

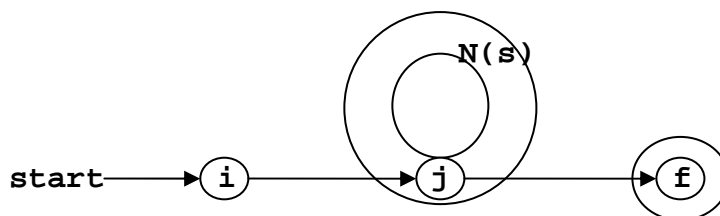
(3) (a)

$N(s)N(t)$  开始状态合并作为  $N(s|t)$  的开始状态

$N(s)N(t)$  接受状态合并作为  $N(s|t)$  的接受状态

(3) (c)

$N(s)$  开始状态与接受状态合并成一个状态  $j$ 。



注意：必须保留一个  $i$  和  $f$ ，否则可能在算法递归产生 NFA 过程中会出问题

### 第三章

#### 3.2 考虑文法

$$S \rightarrow aSbS \mid bSaS \mid \epsilon$$

- (a) 为句子 abab 构造两个不同的最左推导，以此说明该文法是二义的。
- (b) 为 abab 构造对应的最右推导。
- (c) 为 abab 构造对应的分析树。
- (d) 这个文法产生的语言是什么？

Answer:

(a)  $S \Rightarrow lm aSbS$   
 $\Rightarrow lm abS$   
 $\Rightarrow lm abaSbS$   
 $\Rightarrow lm ababS$   
 $\Rightarrow lm abab$  -----①

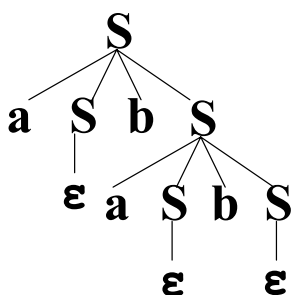
$S \Rightarrow lm aSbS$   
 $\Rightarrow lm abSaSbS$   
 $\Rightarrow lm abaSbS$   
 $\Rightarrow lm ababS$   
 $\Rightarrow lm abab$  -----②

可知，对于句子 abab 存在两个不同的最左推导，所以该文法是二义的

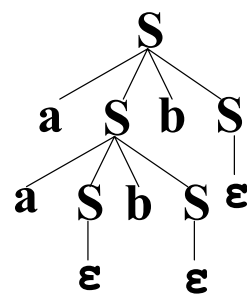
(b)  $S \Rightarrow rm aSbS$   
 $\Rightarrow rm aSb$   
 $\Rightarrow rm abSaSb$   
 $\Rightarrow rm abSab$   
 $\Rightarrow rm abab$  -----③

$S \Rightarrow rm aSbS$   
 $\Rightarrow rm aSbaSbS$   
 $\Rightarrow rm aSbaSb$   
 $\Rightarrow rm aSbab$   
 $\Rightarrow rm abab$  -----④

(c) ①④对应的分析树:



②③对应的分析树:



(d) 该文法产生 a、b 个数相等的 ab 串 (含空串)

## 3.4 文法

$$R \rightarrow R' \mid 'R \mid RR \mid R^* \mid (R) \mid a \mid b$$

产生字母表{a,b}上所有不含  $\epsilon$  的正规式。注意，第一条竖线是正规式的符号“或”，而不是文法产生式右部各选择之间的分隔符，另外“\*”在这儿是一个普通的终结符。该文法是二义的。

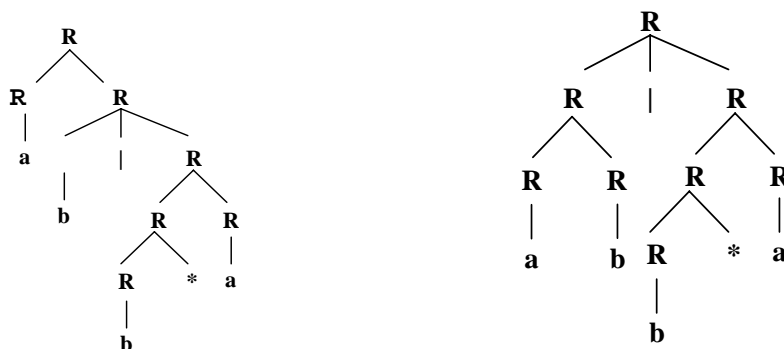
(b) 为该文法写一个等价的非二义的文法。它给予算符\*、链接和|的优先级和结合性同2.2节中定义的一致。

(c) 按上面两个文法构造句子  $ab|b^*a$  的分析树。

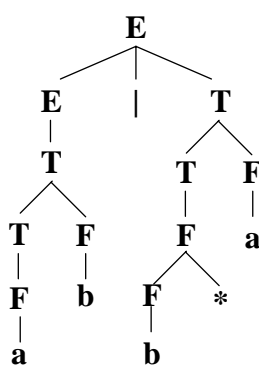
**Answer :**

(b) 标准答案见《编译原理习题精选》P17

(c) 原文法的分析树： //文法二义，存在多个分析树



无二义文法的分析树：



3.6 为字母表 {a, b} 上的下列每个语言设计一个文法，其中哪些语言是正规的？

(a) 每个 a 后面至少有一个 b 跟随的所有串

(b) a 和 b 的个数相等的所有串

Answer:

(a)  $S \rightarrow abS \mid bS \mid \varepsilon$

该语言是正规，对应的正规式是  $(ab|b)^*$

(b) 解法一：

$S \rightarrow aSbS \mid bSaS \mid \varepsilon$

解法二：

$S \rightarrow aB \mid bA \mid \varepsilon$

$A \rightarrow aS \mid bAA$

$B \rightarrow bS \mid aBB$

解法三：

$S \rightarrow abS \mid aSb \mid Sab \mid baS \mid bSa \mid Sba \mid \varepsilon$

该文法不是正规的。

3.8 (a) 消除习题 3.1 文法的左递归。

(b) 为 (a) 的文法构造预测分析器。

Answer:

(a)  $S \rightarrow (L) \mid a$

$L \rightarrow S L1$

$L1 \rightarrow ,S L1 \mid \varepsilon$

(b) 上述文法的预测分析器为：

```

procedure match(t:token);
begin
  if lookahead = t then
    lookahead := nexttoken()
  else error()
end;

procedure s;
begin
  if lookahead = '(' then
    begin
      match('('); L(); match(')')
    end
  else if lookahead = 'a' then
    match('a')
  else error()
end;

```

```
procedure L;  
begin  
  S();L1()  
end;  
  
procedure L1;  
begin  
  if lookahead = ',' then  
  begin  
    match(','); S(); L1()  
  end  
  else match(''  
end;
```

---

**The End** 2004-3-7