

2016 152

一、判断题（本题共 5 道小题，每小题 2 分，总计 10 分）

1. 用高级语言书写的源程序都必须通过编译，产生目标代码后才能投入运行。（×）
2. 甲机上的某编译程序在乙机上能直接使用的必要条件是甲机和乙机的操作系统功能完全相同。（√）
3. 最左推导常被称为规范推导。（×）
4. 一个句型的句柄一定是文法某产生式的右部。（√）
5. 一个 LL(1) 文法一定是无二义的。（×）

二、选择题（本题共 5 道小题，每小题 2 分，总计 10 分）

1. 下面关于解释程序的描述正确的是 (D)  
A) 解释程序的特点是处理程序时不产生目标代码。  
B) 解释程序是为打开编译程序技术的僵局而开发的。  
C) 解释程序不适用于 Pascal 语言。  
D) 解释程序接受某个语言的程序并立即运行这个源程序。
2. 在词法分析阶段不能识别的是 (C)  
A) 运算符      B) 标识符      C) 四元式      D) 常数
3. 文法分为四种类型即 0 型、1 型、2 型、3 型。其中 3 型文法是 (C)  
A) 上下文无关文法      B) 上下文有关文法  
C) 正规文法      D) 短语文法
4. 在自底向上的语法分析方法中，分析的关键是 (A)  
A) 寻找句柄      B) 消除递归      C) 寻找句型      D) 选择候选式
5. 正规式  $M_1$  和  $M_2$  等价是指 (C)  
A)  $M_1$  和  $M_2$  的状态数相等      B)  $M_1$  和  $M_2$  状态数和有向边条数相等  
C)  $M_1$  和  $M_2$  所识别的语言集相等      D)  $M_1$  和  $M_2$  的有向边条数相等

三、简答题（本大题共 3 道小题，每小题 10 分，总计 30 分）

1. 给出文法  $G[S]: A \rightarrow xSx|y$  所生成的语言。

答案:  $L(G) = \{x^n y x^n | n \geq 0\}$

2. 写一文法，使其语言是偶正整数的集合，要求：允许 0 打头（假定 0 为正整数）。

答案:  $G[S] = (\{S, P, D, N\}, \{0, 1, 2, \dots, 9\}, P, S)$

P:  $S \rightarrow PD|D$

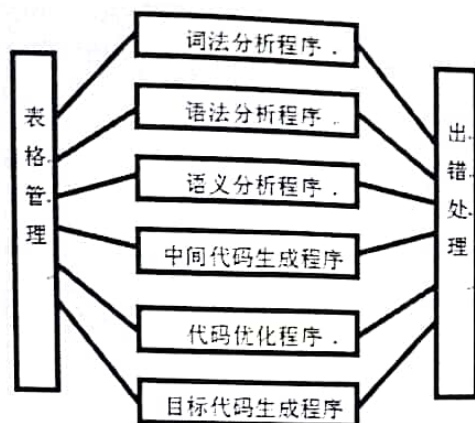
$P \rightarrow NP|N$

$D \rightarrow 0|2|4|6|8$

$N \rightarrow 0|1|2|3|4|5|6|7|8|9$

3. 画出编译程序的总体结构图，简述各部分的主要功能。

答案:



词法分析器：输入源程序，进行词法分析，输出单词符号。

语法分析器：在词法分析的基础上，根据语言的语法规则(文法规则)把单词符号串分解成各类语法单位，并判断输入串是否构成语法上正确的程序。

语义分析器：审查源程序有无语义错误，为代码阶段搜集类型信息。

中间代码生成器：按照语义规则把语法分析器归约(或推导)出的语法单位翻译成一定形式的中间代码，比如说四元式。

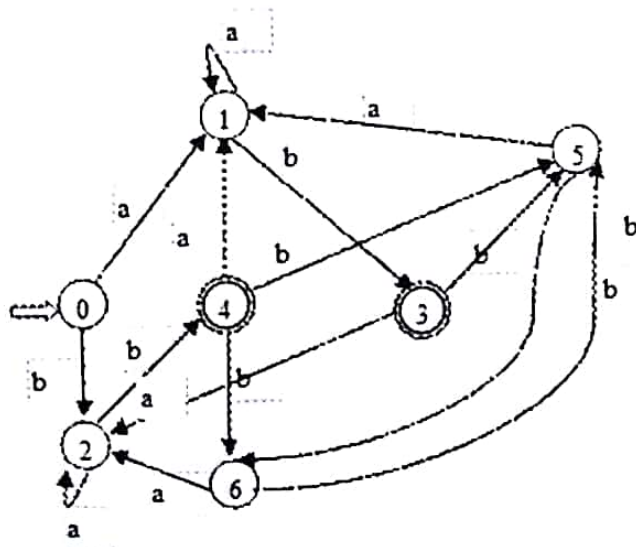
优化：对中间代码进行优化处理。

目标代码生成器：把中间代码翻译成目标语言程序。

表格管理模块保存一系列的表格，登记源程序的各类信息和编译各阶段的进展情况。编译程序各阶段所产生的中间结果都记录在表格中，所需信息多数都需从表格中获取，整个编译过程都在不断地和表格打交道。

出错处理程序对出现在源程序中的错误进行处理。此外，编译的各阶段都可能出现错误，出错处理程序对发现的错误都及时进行处理。

#### 四、画图题(本题共1道小题，总计10分)



答案:

令  $P_0 = (\{0,1,2,5,6\}, \{3,4\})$  用b

进行分割:

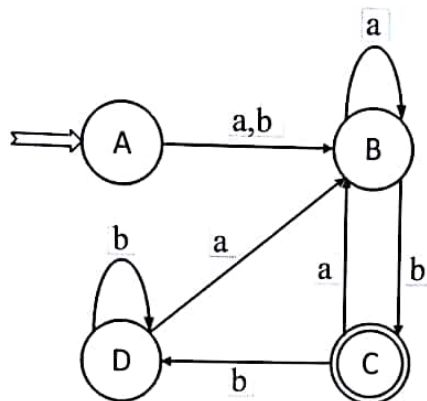
$P_1 = (\{0,5,6\}, \{1,2\}, \{3,4\})$  再用  $b$

进行分割:

$P_2 = (\{0\}, \{5,6\}, \{1,2\}, \{3,4\})$  再用  $a, b$  进行分割, 仍不变。

再令  $\{0\}$  为  $A$ ,  $\{1,2\}$  为  $B$ ,  $\{3,4\}$  为  $C$ ,  $\{5,6\}$  为  $D$ 。

最小化为:



五、计算题 (本题共 2 道小题, 每小题 10 分, 总计 20 分)

1. 已知文法  $G[A]$ :

$A \rightarrow baB | \epsilon$

$B \rightarrow bN | a$

$N \rightarrow aBbb | b$

其中  $A$ 、 $B$  和  $N$  为非终结符, 求每一个非终结符的 FIRST 集合和 FOLLOW 集合。

1. 答案:

$FIRST(A) = \{b, \epsilon\}$

$FIRST(B) = \{b, a\}$

$FIRST(N) = \{b, a\}$

$FOLLOW(A) = \{\#\}$

$FOLLOW(B) = \{\#, b\}$

$FOLLOW(N) = \{\#, b\}$

2. 对文法  $G[S]$ :

$S' \rightarrow \#S\#$

$S \rightarrow a | \uparrow | (T)$

$T \rightarrow T^{\wedge} S | S$

$S \rightarrow a$   
 $S \rightarrow \uparrow$   
 $S \rightarrow (T)$

其中,  $S'$ 、 $S$  和  $T$  为非终结符,  $\#$ 、 $a$ 、 $\uparrow$ 、 $($ 、 $)$  和  $\wedge$  为终结符, 求每一个非终结符的 FIRSTVT 集合和 LASTVT 集合。

2. 答案:

$FIRSTVT(S') = \{\#\}$

$FIRSTVT(S) = \{a, \uparrow, ( \}$

$FIRSTVT(T) = \{^{\wedge}, a, \uparrow, ( \}$

$LASTVT(S') = \{\#\}$

$LASTVT(S) = \{a, \uparrow, ) \}$

$LASTVT(T) = \{^{\wedge}, a, \uparrow, ) \}$

六、分析题 (本题共 2 道小题, 每小题 15 分, 总计 30 分)

1. 考虑文法  $G[E]$ :

(0)  $E \rightarrow S$



- (1)  $S \rightarrow a$
- (2)  $S \rightarrow \wedge$
- (3)  $S \rightarrow (T)$
- (4)  $T \rightarrow T, S$
- (5)  $T \rightarrow S$

其中，E、S 和 T 为非终结符，a、 $\wedge$ 、(、)、和,为终结符，画出识别活前缀的 DFA，构造文法的 LR(0)分析表。

1.答案：

	a	$\wedge$	(	)	,	#	S	T
0	S2	S3	S4				1	
1						acc		
2	R1	R1	R1	R1	R1	R1		
3	R2	R2	R2	R2	R2	R2		
4	S2	S3	S4				6	5
5				S7	S8			
6	R5	R5	R5	R5	R5	R5		
7	R3	R3	R3	R3	R3	R3		
8	S2	S3	S4				9	
9	R4	R4	R4	R4	R4	R4		

一、判断题 (本题共 5 道小题, 每小题 2 分, 总计 10 分)

1. 3 型文法一定是 2 型文法。(√)
2. 如果一个文法存在某个句子对应两颗不同的语法树, 则文法是二义的。(√)
3. 一个上下文无关文法的开始符, 可以是终结符号或非终结符号。(×)
4. 计算机高级语言翻译成低级语言只有解释一种方式。(×)
5. 在规范规约中用最左素短语来刻画可归约串。(×) *算符优先(优先级)*

二、选择题 (本题共 5 道小题, 每小题 2 分, 总计 10 分)

1. 与编译系统相比, 解释系统 (D)

- A) 比较简单, 可移植性好, 执行速度快
- B) 比较简单, 可移植性差, 执行速度慢
- C) 比较复杂, 可移植性好, 执行速度快
- D) 比较简单, 可移植性好, 执行速度慢

2. 编写一个计算机高级语言的源程序后, 到正式上机运行之前, 一般要经过的步骤是 (B)。 (1) 编辑 (2) 编译 (3) 连接 (4) 运行

- A) (1)(2)(3)(4) B) (1)(2)(3) C) (1)(4) D) (1)(3)

3. 不是 DFA 构成成分的是 (C)

- A) 有穷字母表 B) 初始状态集合
- C) 终止状态集合 D) 有限状态集合

4. 一个上下文无关文法 G 包括四个组成部分, 它们是: 一组非终结符号, 一组终结符号, 一个开始符号, 以及一组 (D)

- A) 句型 B) 句子 C) 单词 D) 产生式

5. 从文法开始符推导出的任意符号串是该文法的 (C)

- A) 短语 B) 句子 C) 句型 D) 句柄

二、简答题 (本大题共 3 道小题, 每小题 10 分, 总计 30 分)

1. 给出文法所生成的语言。

G[S]:

$S \rightarrow Ac|aB$   
 $A \rightarrow ab$   
 $B \rightarrow bc$

答案:  $L(G[S]) = \{abc\}$

2. 写一个文法 G, 使其语言为  $L(G) = \{a^n b^n c^m \mid n > 0 \text{ 为奇数}, m > 0 \text{ 为偶数}\}$ 。

答案:

G[S]:  $S \rightarrow AC$   
 $A \rightarrow aaAbb \mid ab$   
 $C \rightarrow ccC \mid cc$

3. 一个典型的编译程序的前端通常由哪些部分组成? 各部分的主要功能是什么?

答案:

一个典型的编译程序的前端通常包含 7 个组成部分, 它们是词法分析程序、语法分析程序、语义分析程序、中间代码生成程序、中间代码优化程序、表格管理程序和错误处理程序。其各部分的主要功能简述如下。

词法分析程序: 从左到右扫描源程序, 识别单词及其相关属性。

语法分析程序: 分析源程序的结构, 判别它是否为相应程序设计语言中的一个合法程序。

语义分析程序: 审查源程序有无语义错误, 为代码生成阶段收集类型信息。

中间代码生成程序: 将源程序变成一种内部表示形式。

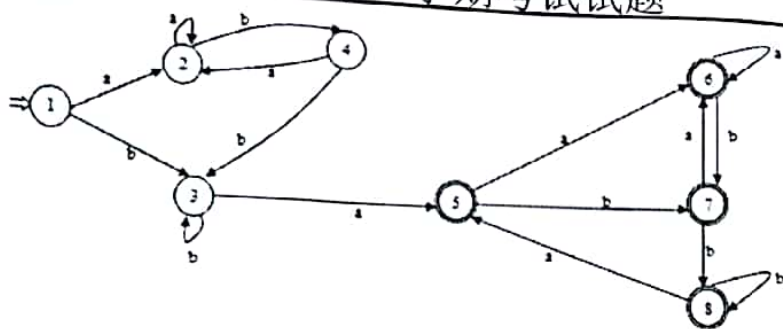
代码优化程序: 对中间代码进行变换或改造, 使生成的代码更为高效。

表格管理程序: 负责建立、填写和查找等一系列表格工作。

错误处理程序: 处理和校正源程序中存在的词法、语法和语义错误。

四、画图题 (本题共 1 道小题, 总计 10 分)

使用分割法将以下 DFA 最小化, 并画出最小化的 DFA。



答案：

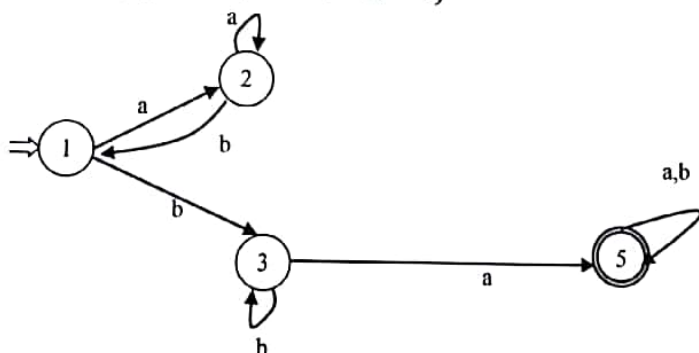
②  $\{1, 2, 3, 4\}$  和  $\{5, 6, 7, 8\}$

②  $3_a=5, 1_a=2, 2_a=2, 4_a=2$

$\{1, 2, 4\}, \{3\}, \{5, 6, 7, 8\}$

③  $1_b=3, 2_b=4, 4_b=3$

$\{1, 4\}, \{2\}, \{3\}, \{5, 6, 7, 8\}$



### 五、计算题（本题共 2 道小题，每小题 10 分，总计 20 分）

1. 对文法  $G[S]$ :

$S \rightarrow eT \mid RT$

$T \rightarrow DR \mid \varepsilon$

$R \rightarrow dR \mid \varepsilon$

$D \rightarrow a \mid bd$

其中  $S$ 、 $T$ 、 $D$ 、 $R$  为非终结符，求每一个非终结符的 FIRST 集合和 FOLLOW 集合。

1. 答案:

$\text{FIRST}(S) = \{e, \varepsilon\}$

$\text{FOLLOW}(S) = \{\#\}$

$\text{FIRST}(T) = \{a, b, \varepsilon\}$

$\text{FOLLOW}(T) = \{\#\}$

$\text{FIRST}(R) = \{d, \varepsilon\}$

$\text{FOLLOW}(R) = \{a, b, \#\}$

$\text{FIRST}(D) = \{a, b\}$

$\text{FOLLOW}(D) = \{d, \#\}$

2. 对文法  $G[S]$ :

$S' \rightarrow \#S\#$

$S \rightarrow D \mid (R)$

$R \rightarrow R; P \mid P$

$P \rightarrow S \mid i$

$D \rightarrow i$

其中  $S'$ 、 $S$ 、 $R$ 、 $P$  和  $D$  为非终结符， $\#$ 、 $($ 、 $)$ 、 $;$  和  $i$  为终结符，求每一个非终结符的 FIRSTVT 集合和 LASTVT 集合。

2. 答案:

$\text{FIRSTVT}(S') = \{\#\}$

$\text{LASTVT}(S') = \{\#\}$

$\text{FIRSTVT}(S) = \{ (, i \}$

$\text{LASTVT}(S) = \{ ), i \}$

$\text{FIRSTVT}(D) = \{ i \}$

$\text{LASTVT}(D) = \{ i \}$

$\text{FIRSTVT}(P) = \{ (, i \}$

$\text{LASTVT}(P) = \{ ), i \}$

FIRSTVT(R)={;, (, i}

LASTVT(R)={;, ), i}

六、问答题 (本题共2道小题, 每小题15分, 总计30分)

1. 考虑文法 G[E]:

(0)  $E \rightarrow S$

(1)  $S \rightarrow SaA$

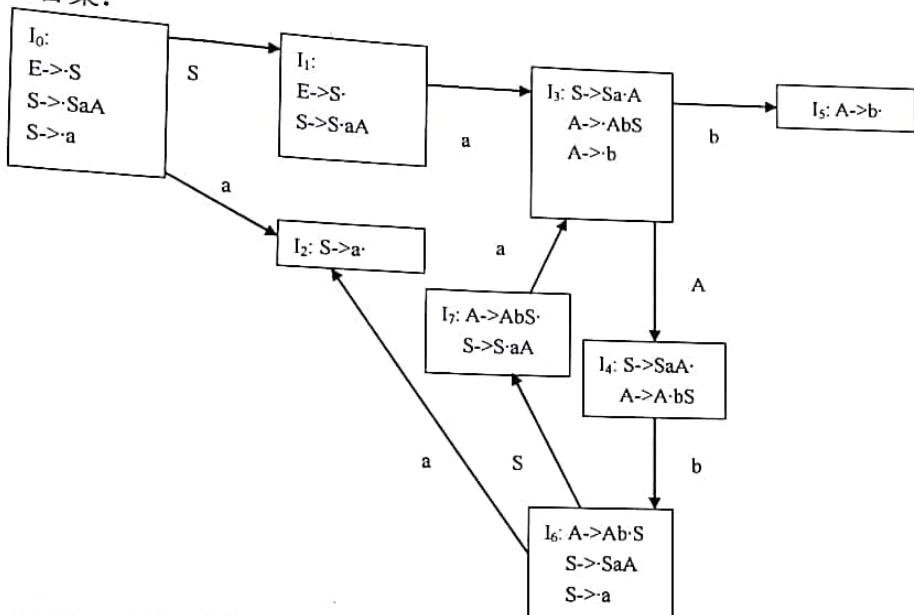
(2)  $S \rightarrow a$

(3)  $A \rightarrow AbS$

(4)  $A \rightarrow b$

其中, E、S 和 A 为非终结符, a 和 b 为终结符, 画出识别活前缀的 DFA, 判断该文法是不是 LR(0)文法, 并说明理由。

1. 答案:



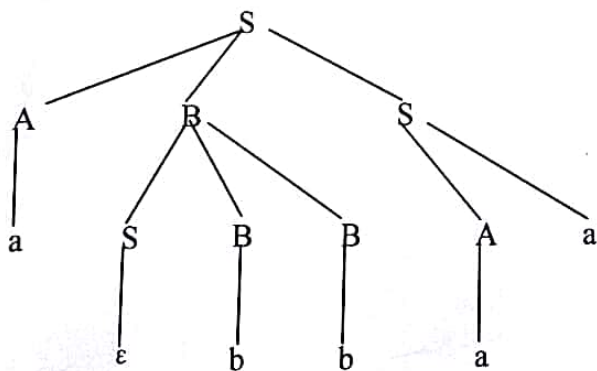
不是 LR(0) 文法,  $I_1$ 、 $I_4$  和  $I_7$  存在着移进-规约冲突。

2. 已知某文法语法树如下图所示:

(1) 写出文法中的产生式规则集合。

(2) 给出句子 abbba 的最左推导。

(3) 给出这个句子的短语、直接短语和句柄。



2. 答案:

(1) 产生式规则:  $S \rightarrow ABS|Aa|\epsilon$

$A \rightarrow a$

$B \rightarrow SBB|b$

(2) 最左推导:

$S \Rightarrow ABS \Rightarrow aBS \Rightarrow aSBB \Rightarrow aBBS \Rightarrow abBS \Rightarrow abbS \Rightarrow abbAa \Rightarrow abbba$

(3) 短语:  $\epsilon, a, b, \epsilon bb, aa, a \epsilon bbaa$

直接短语:  $a, b$

句柄:  $a$



2015 A

一、判断题（本题共 10 道小题，每小题 1 分，总计 10 分）

1. 编译程序与具体的机器有关，与具体的语言无关。（×）
2. 递归下降分析法是自顶向上分析方法。（√）
3. 二义性文法一定不能采用 LL(1) 语法分析方法。（√）
4. 算符优先分析法采用“移近-归约”技术，其归约过程是规范的。（×）
5. 可归前缀是含有句柄的活前缀。（√）
6. 算符优先分析方法是一种自底向上的分析方法，它是以句柄作为每一步归约的对象。（×）
7. 语法分析时必须先消除文法中的左递归。（×）
8. 一个语言可以对应若干文法。（√）
9. 规范归约又称为最左归约。（√）
10. 最左简单子树的末端结点构成的符号串称为句柄。（√）

二、简答题（本大题共 3 道小题，每小题 10 分，总计 30 分）

1. 何谓翻译程序、编译程序和解释程序？它们三者之间有何种关系？

答案：翻译程序是指将用某种语言编写的程序转换成另一种语言形式的程序，如编译程序和汇编程序等。

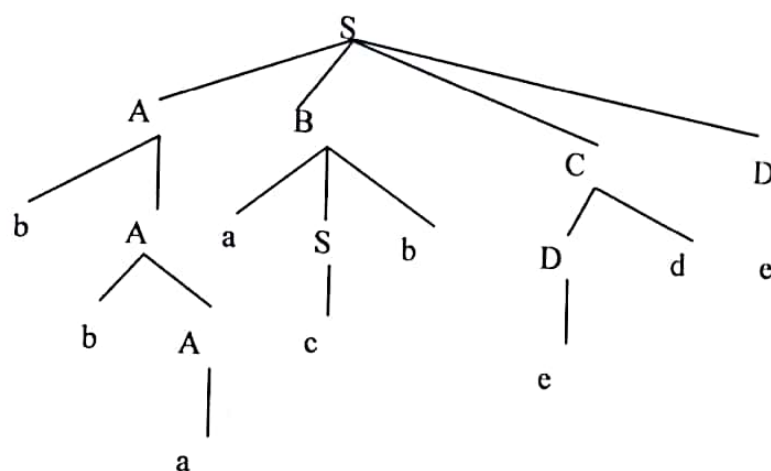
编译程序是把用高级语言编写的源程序转换（加工）成与之等价的另一种用低级语言编写的目标程序的翻译程序。

解释程序是解释、执行高级语言源程序的程序。解释方式一般分为两种：一种方式是，源程序功能的实现完全由解释程序承担和完成，即每读出源程序的一条语句的第一个单词，则依据这个单词把控制转移到实现这条语句功能的程序部分，该部分负责完成这条语句的功能的实现，完成后返回到解释程序的总控部分再读入下一条语句继续进行解释、执行，如此反复；另一种方式是，一边翻译一边执行，即每读出源程序的一条语句，解释程序就将其翻译成一段机器指令并执行之，然后再读入下一条语句继续进行解释、执行，如此反复。

无论是哪种方式，其加工结果都是源程序的执行结果。目前很多解释程序采取上述两种方式的综合实现方案，即先把源程序翻译成较容易解释执行的某种中间代码程序，然后集中解释执行中间代码程序，最后得到运行结果。广义上讲，编译程序和解释程序都属于翻译程序，但它们的翻译方式不同，解释程序是边翻译（解释）边执行，不产生目标代码，输出源程序运行结果。而编译程序只负责把源程序翻译成目标程序，输出与源程序等价的目标程序，而目标程序的执行任务由操作系统来完成，即只翻译不执行。



2. 设文法  $G = (\{S, A, B, C, D\}, \{a, b, c, d, e\}, P, S)$ , 其中,  $P = \{S \rightarrow ABCD, S \rightarrow c, A \rightarrow bA, A \rightarrow a, B \rightarrow aSb, B \rightarrow c, C \rightarrow Dd, D \rightarrow e\}$ , 画出句子  $bbaacbede$  推导树并求出所有短语、直接短语和句柄。  
答案:



根据推导树可以得到短语为:  $a, ba, bba, c, acb, e, ed, e, bbaacb, acbed, ede, bbaacbed, acbede, bbaacbede$

直接短语为:  $a, c, e$

句柄为:  $a$

3. 给出下面文法所生成的语言

$G[S]: S \rightarrow AB$

$A \rightarrow \epsilon | aA$

$B \rightarrow \epsilon | bBc$

答案:

$L(G) = \{a^i b^n c^n | n, i \geq 0\}$ ,  $a^i$  是  $S \rightarrow AB$  前面的  $A$  生成的,  $b^n c^n$  是后面的  $B$  生成的, 且通过这个产生式连接起来

三、证明 (本题共 1 道小题, 总计 10 分)

1、证明: 在算符文法中任何句型都不含两个相邻的非终结符。

证:

已知算符文法的任何一个产生式中都不含两个相邻的非终结符。

对推导长度  $n$  归纳

$n=1$  时

$S \Rightarrow w_0 \Rightarrow w_1 = \gamma$

必存在产生式规则  $S \rightarrow \gamma$ , 由定义知文法产生规则中无相邻非终结符, 成立。

假设  $n > 1$ ,  $w_{n-1}$  成立

$w_{n-1} = \alpha A \delta$ ,  $A$  为非终结符。由假设知  $\alpha$  的尾符号和  $\delta$  的首符号都不可能是非终结符, 否则与假设矛盾。

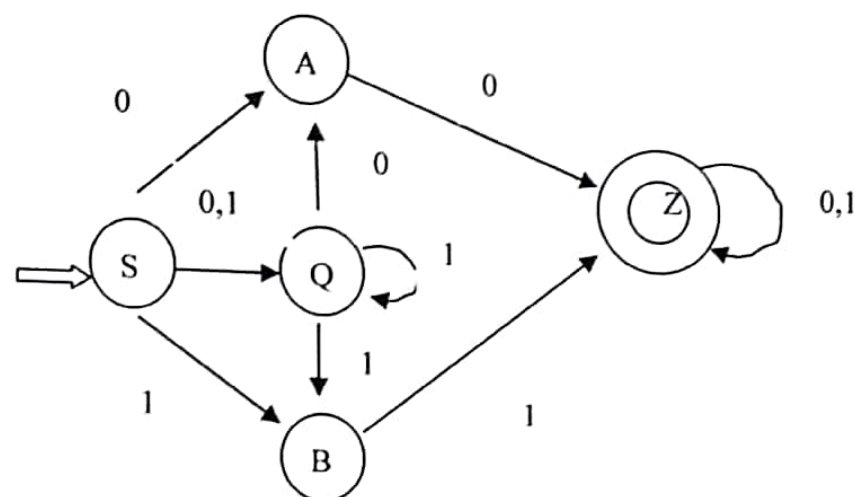
若存在产生  $A \rightarrow \beta$

$$w_{n-1} \Rightarrow w_n = \alpha\beta\delta$$

由于  $\beta$  中无相邻非终结符，则  $\alpha\beta\delta$  中无相邻非终结符。证毕。

四、画图题（本题共 1 道小题，总计 10 分）

答案：



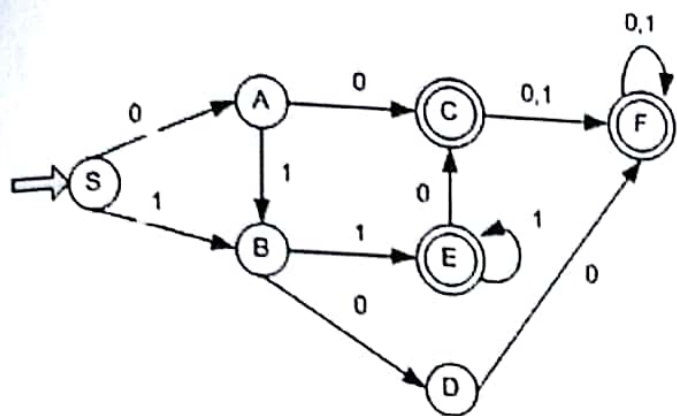
用子集法将 NFA 确定化：

	0	1
S	AQ	QB
AQ	AZ	QB
QB	A	QBZ
AZ	Z	Z
A	Z	
QBZ	AZ	QBZ
Z	Z	Z

重新命名状态子集，令 AQ 为 A，QB 为 B，AZ 为 C，A 为 D，QBZ 为 E，Z 为 F。

	0	1
S	A	B
A	C	B
B	D	E
C	F	F
D	F	
E	C	E
F	F	F

DFA 的状态图：



## 五、计算题（本大题共 2 道小题，每小题 10 分，总计 20 分）

1、已知文法  $G[S]$ :

$S \rightarrow MH|a$     $H \rightarrow LSo|\epsilon$     $K \rightarrow dML|\epsilon$     $L \rightarrow eHf$     $M \rightarrow K|bLM$

试计算每一非终结符的 First 和 Follow 集合。

答案:

$\text{First}(S) = \{a, b, d, e, \epsilon\}$

$\text{First}(M) = \{b, d, \epsilon\}$

$\text{First}(K) = \{d, \epsilon\}$

$\text{First}(H) = \{e, \epsilon\}$

$\text{First}(L) = \{e\}$

$\text{Follow}(S) = \{\#, o\}$

$\text{Follow}(H) = \text{Follow}(S) \cup \text{First}\{f\} = \{\#, f, o\}$

$\text{Follow}(K) = \text{Follow}(M) = \{\#, e, o\}$

$\text{Follow}(L) = \text{Follow}(M) \cup (\text{First}(M) - \{\epsilon\}) \cup \text{Follow}(K) \cup (\text{First}(So) - \{\epsilon\})$

$= \text{Follow}(M) \cup \{b, d\} \cup \text{Follow}(K) \cup \{a, b, d, e, o\}$

$= \{\#, a, b, d, e, o\}$

$\text{Follow}(M) = \text{Follow}(M) \cup (\text{First}(L) - \{\epsilon\}) \cup (\text{First}(H) - \{\epsilon\}) \cup$

$\text{Follow}(S)$

$= \text{Follow}(M) \cup \{e\} \cup \{e\} \cup \{\#, o\} = \{\#, e, o\}$

2、若文法  $G[E']$  为

$E' \rightarrow \#E\#$

$E \rightarrow E-T|T$

$T \rightarrow T * F | F$

$F \rightarrow (E) | i$

计算每个非终结符的 FIRSTVT 集和 LASTVT 集；画出算符优先分析矩阵。

$\text{FIRSTVT}(E') = \{\#\}$

$\text{FIRSTVT}(E) = \{-, *, (, i\}$

$\text{FIRSTVT}(T) = \{*, (, i\}$

$\text{FIRSTVT}(F) = \{(, i\}$



$LASTVT(E') = \{ \# \}$   
 $LASTVT(E) = \{ -, *, ), i \}$   
 $LASTVT(T) = \{ *, ), i \}$   
 $LASTVT(F) = \{ ), i \}$

(1)=

$E' \rightarrow \#E\#$

$\# = \#$

$F \rightarrow (E)$

$( = )$

(2)<

$\#E$

$\# < FIRSTVT(E)$

$-T$

$- < FIRSTVT(T)$

$*F$

$* < FIRSTVT(F)$

$(E$

$( < FIRSTVT(E)$

(3)>

$E\#$

$LASTVT(E) > \#$

$E-$

$LASTVT(E) > -$

$T*$

$LASTVT(T) > *$

$E)$

$LASTVT(E) > )$

	-	*	(	)	i	#
-	>	<	<	>	<	>
*	>	>	<	>	<	>
(	<	<	<	=	<	
)	>	>		>		>
i	>	>		>		>
#	<	<	<		<	=

六、分析题（本题共 1 道小题，总计 20 分）

考虑文法  $G[S']$ :

(0)  $S' \rightarrow S$

(1)  $S \rightarrow aBc$

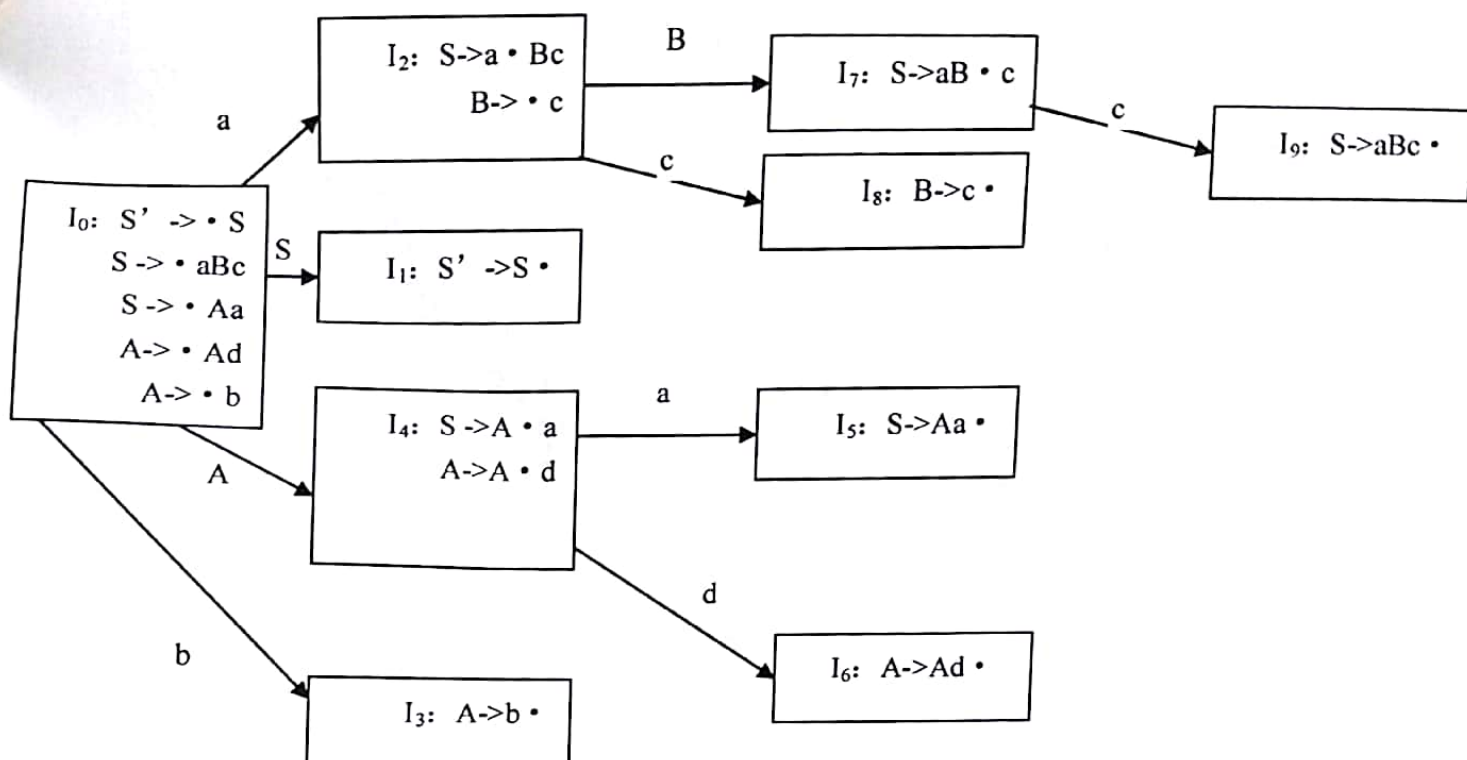
(2)  $S \rightarrow Aa$

(3)  $A \rightarrow Ad$

(4)  $A \rightarrow b$

(5)  $B \rightarrow c$

画出识别活前缀的 DFA，构造文法的 LR(0) 分析表。



	a	b	c	d	#	S	A	B
0	S2	S3				1	4	
1					acc			
2			S8					7
3	R4	R4	R4	R4	R4			
4	S5			S6				
5	R2	R2	R2	R2	R2			
6	R3	R3	R3	R3	R3			
7			S9					
8	R5	R5	R5	R5	R5			
9	R1	R1	R1	R1	R1			

一、判断题(本题共 10 道小题, 每小题 1 分, 总计 10 分)

1. 3 型文法一定是 2 型文法。(✓)
2. 句型不一定是句子。(✓)
3. 直接短语才可能是句柄。(✓)
4. 一个文法对应唯一的语言, 反之亦然。(×)
5. 如果一个文法存在某个句子对应两颗不同的语法树, 则文法是二义的。(✓)
6. 对任意一个正规文法  $G$ , 都存在一个 DFA  $M$ , 满足  $L(G)=L(M)$ 。(✓)
7. 语法分析时必须先消除文法中的左递归。(×)
8. 一个上下文无关文法的开始符, 可以是终结符号或非终结符号。(×)
9. 2 型语言可以含有形如  $AB \rightarrow ab$  的规则。(×)
10. 对于任何一个编译程序来说, 产生中间代码是必不可少的。(×)

二、简答题(本大题共 3 道小题, 每小题 10 分, 总计 30 分)

1. 计算机执行用高级语言编写的程序有哪些途径? 它们之间的主要区别是什么?

答案: 计算机执行用高级语言编写的程序主要途径有两种, 即解释与编译。像 Basic 之类的语言, 属于解释型的高级语言。它们的特点是计算机并不事先对高级语言进行全盘翻译, 将其变为机器代码, 而是每读入一条高级语句, 就用解释器将其翻译为一条机器代码, 予以执行, 然后再读入下一条高级语句, 翻译为机器代码, 再执行, 如此反复。

总而言之, 是边翻译边执行。像 C, Pascal 之类的语言, 属于编译型的高级语言。它们的特点是计算机事先对高级语言进行全盘翻译, 将其全部变为机器代码, 再统一执行, 即先翻译, 后执行。从速度上看, 编译型的高级语言比解释型的高级语言更快。

2. 给定下面文法

$S \rightarrow a|b|(T)$

$T \rightarrow TdS|S$

求句型  $(Sd(T)db)$  的短语, 直接短语, 句柄?

答案: 根据推导过程, 短语包含:  $S, (T), b, Sd(T), Sd(T)db, (Sd(T)db) \dots$

直接短语为  $S, (T), b$ 。

句柄为  $S$ 。

3. 给出下面文法所生成的语言

$G[S]: S \rightarrow AB$

$A \rightarrow \epsilon | aA$

$B \rightarrow \epsilon | bBc$

答案:

$L(G) = \{a^i b^n c^n | n, i \geq 0\}$ ,  $a^i$  是通过前面的  $S \rightarrow AB$  前面的  $A$  生成的,  $b^n c^n$  是通过后面的  $B$  生成的, 且通过这个产生式连接起来。

三、证明(本题共 1 道小题, 总计 10 分)

证明: 如果  $Ab$ (或  $bA$ ) 出现在算符文法的句型  $\gamma$  中, 其中  $A \in V_N, b \in V_T$ , 则  $\gamma$  中任何含此  $b$  的短语必含有  $A$ ?

证: 用反证法

由算符文法的性质 1 知可有:

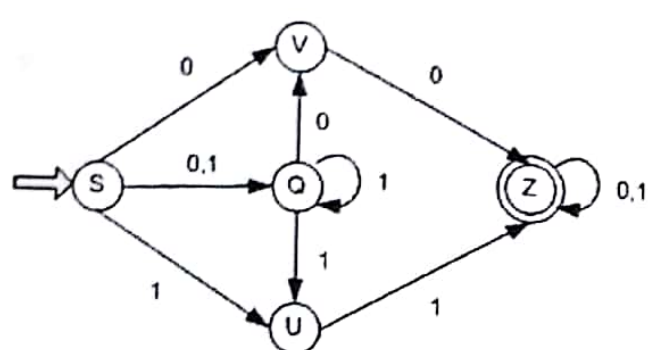
$$S \Rightarrow \gamma = a\dot{b}A\beta$$

若存在  $B \Rightarrow a\dot{b}$ , 这时  $b$  和  $A$  不同时归约, 则必有  $S \Rightarrow BA\beta$ , 这样在句型  $BA\beta$  中, 存在相邻的非终结符  $B$  和  $A$ , 所以与性质 1 矛盾, 证毕。

四、画图题(本题共 1 道小题, 总计 10 分)

答案:





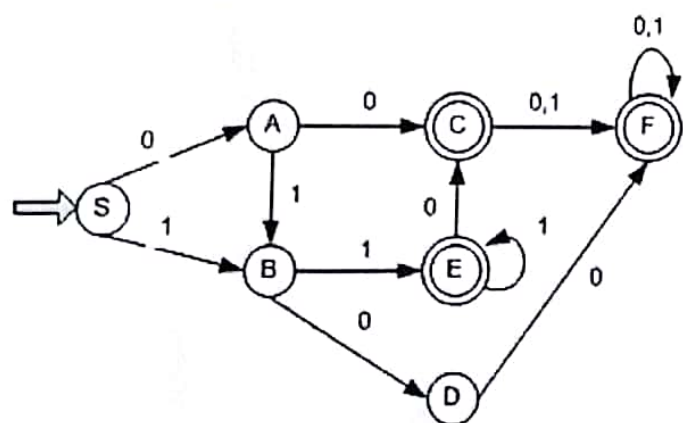
用子集法将 NFA 确定化:

	0	1
S	VQ	QU
VQ	VZ	QU
QU	V	QUZ
VZ	Z	Z
V	Z	
QUZ	VZ	QUZ
Z	Z	Z

重新命名状态子集, 令 VQ 为 A, QU 为 B, VZ 为 C, V 为 D, QUZ 为 E, Z 为 F。

	0	1
S	A	B
A	C	B
B	D	E
C	F	F
D	F	
E	C	E
F	F	F

DFA 的状态图:



五、计算题 (本大题共 2 道小题, 每小题 10 分, 总计 20 分)

1、已知文法  $G[S]$ :

$S \rightarrow MH|a$   $H \rightarrow LSo|\epsilon$   $K \rightarrow dML|\epsilon$   $L \rightarrow eHf$   $M \rightarrow K|bLM$

试计算每一非终结符的 First 和 Follow 集合。

答案:

$\text{First}(S) = \{a, b, d, e, \epsilon\}$

$\text{First}(M) = \{b, d, \epsilon\}$

$\text{First}(K) = \{d, \epsilon\}$

$\text{First}(H) = \{e, \epsilon\}$

$\text{First}(L) = \{e\}$

$$\text{Follow}(S) = \{\#, o\}$$

$$\text{Follow}(H) = \text{Follow}(S) \cup \text{First}\{f\} = \{\#, f, o\}$$

$$\text{Follow}(K) = \text{Follow}(M) = \{\#, e, o\}$$

$$\begin{aligned} \text{Follow}(L) &= \text{Follow}(M) \cup (\text{First}(M) - \{\epsilon\}) \cup \text{Follow}(K) \cup (\text{First}(S_o) - \{\epsilon\}) \\ &= \text{Follow}(M) \cup \{b, d\} \cup \text{Follow}(K) \cup \{a, b, d, e, o\} \\ &= \{\#, a, b, d, e, o\} \end{aligned}$$

$$\begin{aligned} \text{Follow}(M) &= \text{Follow}(M) \cup (\text{First}(L) - \{\epsilon\}) \cup (\text{First}(H) - \{\epsilon\}) \cup \text{Follow}(S) \\ &= \text{Follow}(M) \cup \{e\} \cup \{e\} \cup \{\#, o\} = \{\#, e, o\} \end{aligned}$$

2、若文法  $G[E']$  为

$$E' \rightarrow \#E\#$$

$$E \rightarrow E-T|T$$

$$T \rightarrow T*F|F$$

$$F \rightarrow (E)|i$$

计算每个非终结符的 FIRSTVT 集和 LASTVT 集；画出算符优先分析矩阵。

$$\text{FIRSTVT}(E') = \{\#\}$$

$$\text{FIRSTVT}(E) = \{-, *, (, i\}$$

$$\text{FIRSTVT}(T) = \{*, (, i\}$$

$$\text{FIRSTVT}(F) = \{(, i\}$$

$$\text{LASTVT}(E') = \{\#\}$$

$$\text{LASTVT}(E) = \{-, *, ), i\}$$

$$\text{LASTVT}(T) = \{*, ), i\}$$

$$\text{LASTVT}(F) = \{), i\}$$

(1)=

$$E' \rightarrow \#E\# \quad \# = \#$$

$$F \rightarrow (E) \quad (=)$$

(2)<

$$\#E \quad \# < \text{FIRSTVT}(E)$$

$$-T \quad - < \text{FIRSTVT}(T)$$

$$*F \quad * < \text{FIRSTVT}(F)$$

$$(E \quad ( < \text{FIRSTVT}(E)$$

(3)>

$$E\# \quad \text{LASTVT}(E) > \#$$

$$E- \quad \text{LASTVT}(E) > -$$

$$T* \quad \text{LASTVT}(T) > *$$

$$E) \quad \text{LASTVT}(E) > )$$

	-	*	(	)	i	#
-	>	<	<	>	<	>
*	>	>	<	>	<	>
(	<	<	<	=	<	
)	>	>		>		>
i	>	>		>		>
#	<	<	<		<	=

六、分析题（本题共 1 道小题，总计 20 分）

考虑文法  $G[S]$ :

(0)  $S' \rightarrow S$

(1)  $S \rightarrow aBc$

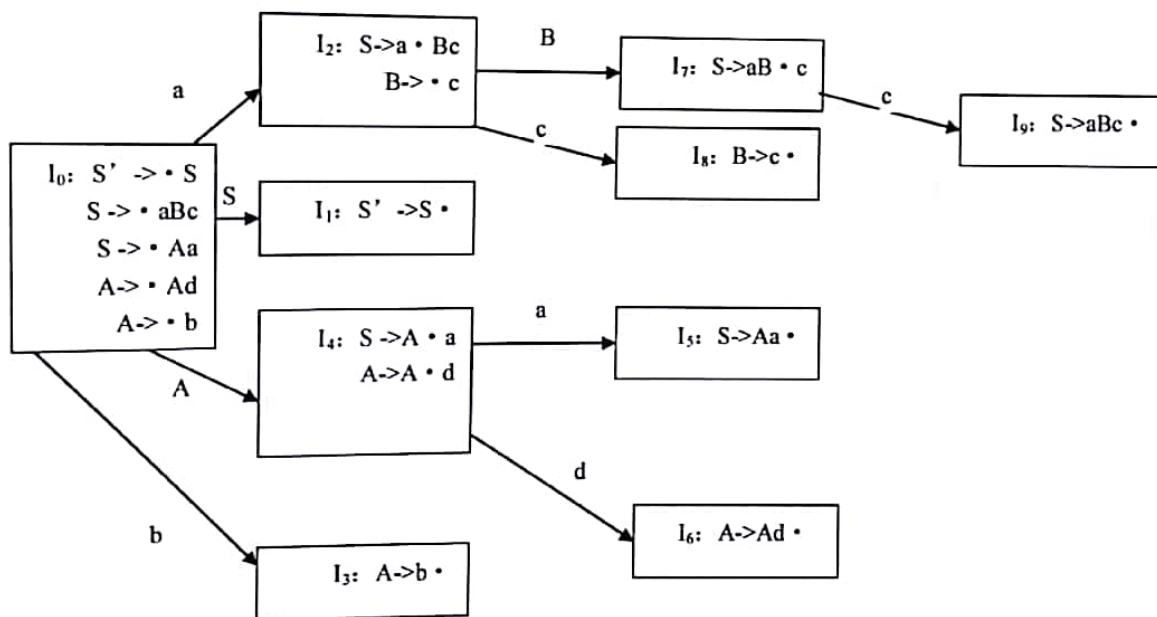
(2)  $S \rightarrow Aa$

(3)  $A \rightarrow Ad$

(4)  $A \rightarrow b$

(5)  $B \rightarrow c$

画出识别活前缀的 DFA，构造文法的 LR(0)分析表。



	a	b	c	d	#	S	A	B
0	S2	S3				1	4	
1					acc			
2			S8					7
3	R4	R4	R4	R4	R4			
4	S5			S6				
5	R2	R2	R2	R2	R2			
6	R3	R3	R3	R3	R3			
7			S9					
8	R5	R5	R5	R5	R5			
9	R1	R1	R1	R1	R1			