

得分

一. 填空题 (每空 2 分, 共 20 分)

1. 不同的编译程序关于数据空间的存储分配策略可能不同, 但大部分编译中采用的方案有两种: 静态存储分配方案和动态存储分配方案, 而后者又分为 (1) 和 (2)。
2. 规范规约是最 (3) 规约。
3. 编译程序的工作过程一般划分为 5 个阶段: 词法分析、(4)、语义分析与中间代码生成, 代码优化及 (5)。另外还有 (6) 和出错处理。
4. 表达式  $x+y*z/(a+b)$  的后缀式为 (7)。
5. 文法符号的属性有综合属性和 (8)。
6. 假设二维数组按行存放, 而且每个元素占用一个存储单元, 则数组  $a[1..15, 1..20]$  某个元素  $a[i, j]$  的地址计算公式为 (9)。
7. 局部优化是局限于一个 (10) 范围内的一种优化。

得分

二. 选择题 (1-6 为单选题, 7-8 为多选题, 每问 2 分, 共 20 分)

1. 一个上下文无关文法  $G$  包括四个组成部分: 一组终结符, 一组非终结符, 一个 ( ), 以及一组 ( )。  
A. 字符串      B. 产生式      C. 开始符号      D. 文法
2. 程序的基本块是指 ( )。  
A. 一个子程序      B. 一个仅有一个入口和一个出口的语句  
C. 一个没有嵌套的程序段      D. 一组顺序执行的程序段, 仅有一个入口和一个出口
3. 高级语言编译程序常用的语法分析方法中, 递归下降分析法属于 ( ) 分析方法。  
A. 自左向右      B. 自顶向下      C. 自底向上      D. 自右向左
4. 在通常的语法分析方法中, ( ) 特别适用于表达式的分析。  
A. 算符优先分析法      B. LR 分析法

- C. 递归下降分析法                      D. LL(1) 分析法
5. 经过编译所得到的目标程序是 ( )。
- A. 四元式序列                      B. 间接三元式序列
- C. 二元式序列                      D. 机器语言程序或汇编语言程序
6. 一个文法所描述的语言是 ( )；描述一个语言的文法是 ( )。
- A. 唯一的                      B. 不唯一的                      C. 可能唯一，也可能不唯一
7. 如果在文法  $G$  中存在一个句子，当其满足下列条件 ( ) 之一时，则称该文法是二义文法。
- A. 其最左推导和最右推导相同                      B. 该句子有两个不同的最左推导
- C. 该句子有两个不同的最右推导                      D. 该句子有两棵不同的语法树
- E. 该句子对应的语法树唯一
8. 下面 ( ) 语法制导翻译中，采用拉链—回填技术。
- A. 赋值语句                      B. 布尔表达式的计算                      C. 条件语句                      D. 循环语句

得分

### 三. 解答题 (共 60 分)

1. (共 15 分) 已知文法  $G[E]$ :

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

$$T \rightarrow * | +$$

- (1) 将文法  $G$  改造成 LL(1) 文法; (5 分)
  - (2) 构造文法  $G$  中每个非终结符的 FIRST 集合及 FOLLOW 集合; (5 分)
  - (3) 构造 LL(1) 分析表。(5 分)
2. (共 12 分) 给定文法  $G[S]$ :  $S \rightarrow S(S) | \epsilon$
- (1) 给出句子  $((00)00)$  的规范推导过程; (4 分)

(2) 指出每步推导所得句型的句柄; (4分)

(3) 画出该句子的语法推导树。(4分)

3. (共 8 分) 在一个移入-规约分析过程中采用以下的语法制导翻译模式, 在按一个产生式规约时, 立即执行括号中的动作。

**A→aB     {print “0”; }**

```
A→c    {print "1"; }
```

```
B→Ab    {print "2"; }
```

(1) 当分析器的输入为 `aacbb` 时, 打印的字符串是什么? (3分)

(2) 写出分析过程。(5分)

4. (10分) 翻译循环语句 `while (a<b) do if (c>d) then x:=y+z`。要求：给出加注释的分析树及四元式序列。

参考以下部分翻译模式：

```
(1)  S → if E then M S1           {backpatch(E.truelist, M.quad);
                                     S.nextlist := merge(E.falselist, S1.nextlist)}
```

[illegible]

(3)  $S \rightarrow A$  {S.nextlist:=makelist()}

(4)  $L \rightarrow S$   $\{L.nextlist := S.nextlist\}$

(5)  $M \rightarrow \varepsilon$   $\{M.\text{quad} := \text{nextquad}\}$

(6)  $E \rightarrow id_1 \text{ relop } id_2$        $\{E.truelist := makelist(nextquad);$

e.falselist:=makelist(nextquad+1);

emit('j'relop.op,' ,id1.place',id2.place','0');

emit('j,-,0')}

(7)  $S \rightarrow L := E$  {emit(=,E.place,-,L.place)}

(8)  $E \rightarrow E_1 + E_2$  {E.place:=newtemp;

emit(+,E1.place,E2.place,E.place,)}

5. (共 15 分) 设有表格构造文法  $G[S]$ :

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

$T \rightarrow T, S | S$

(1) 计算文法  $G[S]$  的 FIRSTVT 集和 LASTVT 集。(5 分)

(2) 构造  $G[S]$  的优先关系表, 并判断  $G[S]$  是否为算符优先文法。(5 分)

(3) 计算  $G[S]$  的优先函数。(5 分)

得分

二. 单项选择题 (每题 2 分, 共 10 分)

1. 设有文法  $G[]$ :  $I \rightarrow I1 | I0 | Ia | Ic | a | b | c$

下列符号串中是该文法句子的有 ( )。

① ab0

② a0c01

③ aaa

④ bc10

可选项有:

- A. ①                      B. ②③④                      C. ③④                      D. ①②③④

2. 程序的基本块是指 ( )。

- A. 一个子程序                      B. 一个仅有一个入口和一个出口的语句  
C. 一个没有嵌套的程序段      D. 一组顺序执行的程序段, 仅有一个入口和一个出口

3. 高级语言编译程序常用的语法分析方法中, 递归下降分析法属于 ( ) 分析方法。

- A. 自左向右      B. 自顶向下      C. 自底向上      D. 自右向左

4. 经过编译所得到的目标程序是 ( )。

- A. 四元式序列                      B. 间接三元式序列  
C. 二元式序列                      D. 机器语言程序或汇编语言程序

5. 运行阶段的存储组织与管理的目的是 ( )。

- ① 提高编译程序的运行速度                      ② 节省编译程序的存储空间  
③ 提高目标程序的运行速度                      ④ 为运行阶段的存储分配做准备

可选项有:

- A. ①②                      B. ②③                      C. ③④                      D. ④②

得分

2. (10分) 已知文法  $G[S]$ :

$S \rightarrow aBc \mid bAB$

$A \rightarrow aAb \mid b$

$B \rightarrow b \mid \varepsilon$

(4) 构造其 LL(1) 分析表;

(5) 判断符号串 **baabbb** 是否为该文法的句子 (写出含有符号栈、输入串和规则的分析过程)。

3. (10分) 已知文法  $G$  为:

$E \rightarrow E+T | T$

$T \rightarrow T * P | P$

$P \rightarrow i$

(1) 构造该文法的优先关系表（不考虑语句括号#），并指出此文法是否为算符优先文法。

(2) 构造文法 G 的优先函数表。

4. （8分）在一个移入-规约分析过程中采用以下的语法制导翻译模式，在按一个产生式规约时，立即执行括号中的动作。

$S \rightarrow bAb \quad \{\text{print "1"}\}$

$A \rightarrow (B \quad \{\text{print "2"}\}$

$A \rightarrow a \quad \{\text{print "3"}\}$

$B \rightarrow Aa \quad \{\text{print "4"}\}$

(3) 当输入序列为  $b(((aa)a)a)b$  时，打印的字符串是什么？

(4) 写出移入-规约分析过程。

5. （12分）翻译循环语句  $\text{while } (x > y) \text{ do if } (a = b) \text{ then } x := 2 * y + a$ 。要求：给出加注释的分析树、三地址码序列及相应的四元式序列。

参考以下部分翻译模式：

(1)  $S \rightarrow \text{if } E \text{ then } M \ S_1 \quad \{\text{backpatch}(E.\text{truelist}, M.\text{quad});$   
 $S.\text{nextlist} := \text{merge}(E.\text{falselist}, S_1.\text{nextlist})\}$

(2)  $S \rightarrow \text{while } M_1 \ E \ \text{do } M_2 \ S_1 \quad \{\text{backpatch}(S_1.\text{nextlist}, M_1.\text{quad});$   
 $\text{backpatch}(E.\text{truelist}, M_2.\text{quad});$   
 $S.\text{nextlist} := E.\text{falselist}$   
 $\text{emit}('j, -, ', M_1.\text{quad})\}$

- (3)  $S \rightarrow A$  {S.nextlist:=makelist()}
- (4)  $L \rightarrow S$  {L.nextlist:=S.nextlist}
- (5)  $M \rightarrow \epsilon$  {M.quad:=nextquad}
- (6)  $E \rightarrow id_1 \text{ relop } id_2$  {E.truelist:=makelist(nextquad);  
e.falselist:=makelist(nextquad+1);  
emit('j' relop.op, 'id1.place', 'id2.place', '0');  
emit('j,-,0')}
- (7)  $S \rightarrow L := E$  {emit(:=,E.place,-,L.place)}
- (8)  $E \rightarrow E_1 + E_2$  {E.place:=newtemp;  
emit(+,E1.place,E2.place,E.place,)}

6. (8 分) Generate assembly code for the following sequence assuming that x,y and z are in memory locations(noticing only two registers R1 and R2).

S=0

I=0

L1: if x>y goto L2

Z=s+a[i]

I=i+1

Goto L1

L2:

7. (6 分) Give out the all basic blocks of the following program fragment and construct the relevant flow graph(DAG).

read C

A=0

B=1

L4: A=A+B

if B>C goto L2

B=B+1

goto L4

L2: write A

8. (8 分) Translate the assignment statement  $b[i]=b*c-b*d$  into

- (1) A syntax tree.
- (2) Three address instructions.

答案:

- (1) 栈式动态存储分配
- (2) 堆式动态存储分配
- (3) 左
- (4) 语法分析
- (5) 目标代码生成
- (6) 表格管理
- (7)  $xyz*ab++$
- (8) 继承属性
- (9)  $a+(i-1)*20+j-1$
- (10) 基本块

一、 选择题 (每问 2 分, 共 20 分)

1.C B    2.D    3.B    4.A    5.D    6.A,C



7.BCD,选对一个得1分且不超过满分,选错一个扣一分,扣完为止。

8.BCD,选对一个得1分且不超过满分,选错一个扣一分,扣完为止。

## 二、解答题

1. (1) 文法存在左递归,消除左递归后的文法为:

$E \rightarrow (E)E' | iE'$  (2分)

$E' \rightarrow TEE' | \varepsilon$  (2分)

$T \rightarrow * | +$  (1分)

(2) (5分)没考虑#扣0.5分,其它错或少写一个扣0.5分

$FIRST(E) = \{(, i\}$   $FIRST(E') = \{*, +, \varepsilon\}$   $FIRST(T) = \{*, +\}$

$FOLLOW(E) = \{), *, +, \#\}$   $FOLLOW(E') = \{), *, +, \#\}$   $FOLLOW(T) = \{(, i\}$

(3) 每错一个扣0.5分,全错或不写不得分,扣完为止,共5分

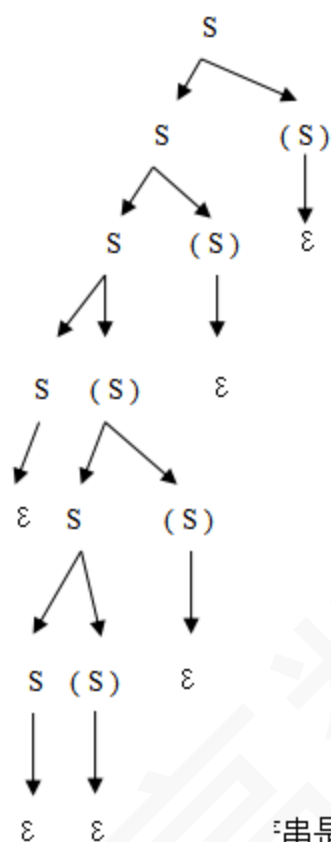
	(	)	i	*	+	#
E	$E \rightarrow (E)E'$		$E \rightarrow iE'$			
E'		$E' \rightarrow \varepsilon$		$E' \rightarrow TEE'$ $E' \rightarrow \varepsilon$	$E' \rightarrow TEE'$ $E' \rightarrow \varepsilon$	$E' \rightarrow \varepsilon$
T				$T \rightarrow *$	$T \rightarrow +$	

2. (1) 规范推导过程如下。写错推导符号扣0.5分,错写或少写一步推导扣0.5分,扣完为止,最左推导扣2分,共4分。

$S \Rightarrow \underline{S(S)} \Rightarrow S(\underline{\varepsilon}) \Rightarrow \underline{S(S)}0 \Rightarrow S(\underline{\varepsilon})0 \Rightarrow \underline{S(S)}00 \Rightarrow S(\underline{S(S)})00 \Rightarrow S(S(\underline{\varepsilon}))00 \Rightarrow S(\underline{S(S)})000$   
 $\Rightarrow S(S(\underline{\varepsilon})0)00 \Rightarrow S((\underline{\varepsilon})0)00 \Rightarrow \underline{\varepsilon}(00)00$

(2) (1) 中加下划线的部分是句柄,标识如(1)。每少写一个句柄扣0.5分,扣完为止,共4分。

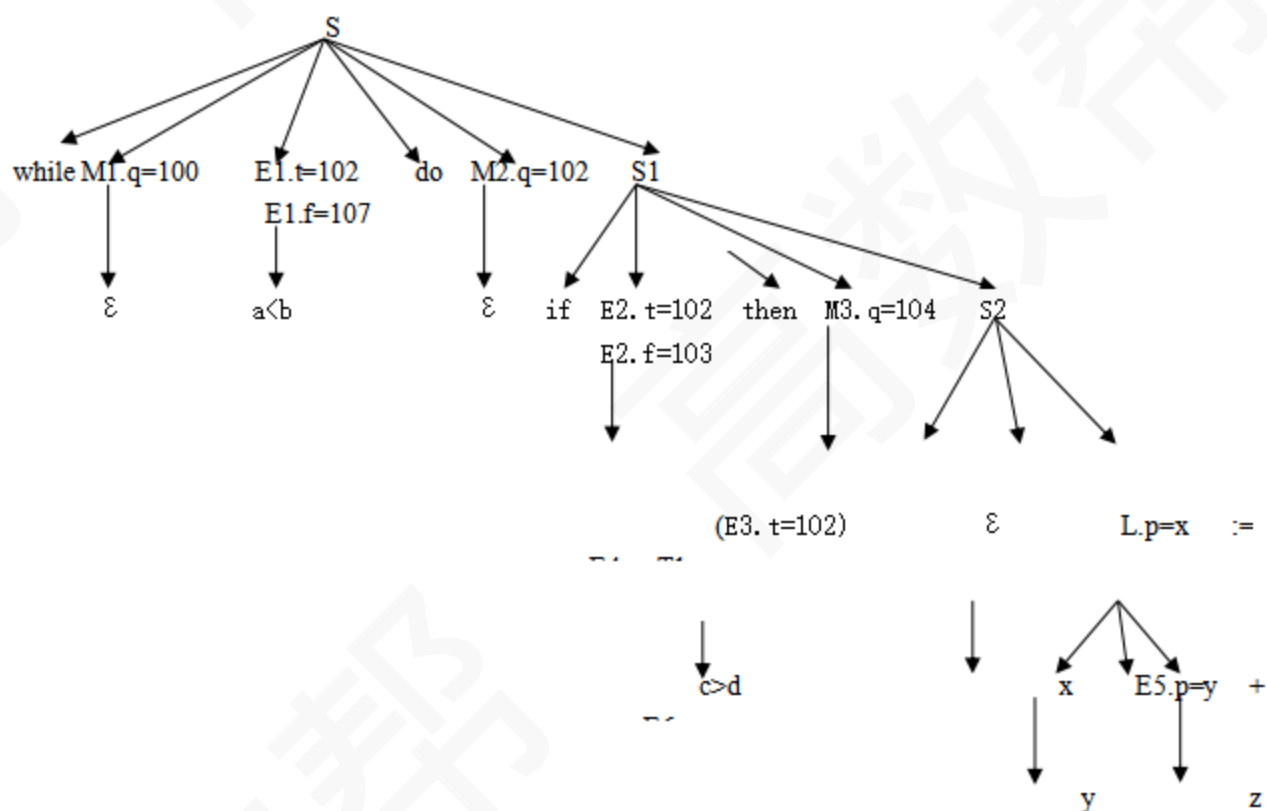
(3) 每少写步扣0.5分,扣完为止,共4分。



串是：12020（错一个扣 0.5 分，共 3 分）

(2) 归约过程中错一步扣 0.5 分，扣完为止。(共 5 分)

4. (1) 每少写一步扣 0.5 分, 扣完为止, 共 5 分。



(2) 少写一个四元式扣 0.5 分，全错或不写不得分，回填错误扣 0.5 分，共 5 分。

四元式序列为：

100( $j < a, b, 102$ )

101( $j, \_, \_, 107$ )

102( $j > c, d, 104$ )

103( $(j, \_, \_, 106)$ )

104( $+, y, z, T1$ )

105( $:=, T1, \_, x$ )

106( $j, \_, \_, 100$ )

5. (1) 少写一个扣 1 分，全错或不写不得分，共 5 分。

FIRSTVT(S)={a, ^, {}

FIRSTVT(T)={, a, ^, {}

LASTVT(S)={ a, ^, , }

LASTVT(T)={ a, ^, , , }

(2) 优先表如下。每错一个扣 0.5 分，全错或不写不得分，扣完为止，共 3 分

文法 G[S] 没有两个非终结符相邻的情况，且其优先表中任一对终结符之间最多满足  $\leq$ 、 $>$ 、 $=$  三种关系中的一种，因此是 G[S] 算符优先文法。(2 分)

可以不考虑终结符“#”。

	a	^	(	)	,	#
A				$>$	$>$	$>$
^				$>$	$>$	$>$
(	$\leq$	$\leq$	$\leq$	$=$	$\leq$	
)				$>$	$>$	
,	$\leq$	$\leq$	$\leq$	$>$	$>$	$>$
#	$\leq$	$\leq$	$\leq$		$\leq$	$=$

或者

(3) 优先函数。可以不考虑终结符“#”。每错一个扣 0.5 分，全错或不写不得分，扣完为止，共 5 分。

	a	^	(	)	,	#
f	6	6	2	6	6	2
g	7	7	7	2	5	2

或者

	a	^	(	)	,
f	4	4	2	4	4
g	5	5	5	2	3

### 三、 填空题（每空 2 分，共 20 分）

- 1 目标程序 (target code)      语法分析 (syntax analyzer)      代码优化器 (code optimizer)  
代码产生器 (code generator)      符号表管理 (symbol table manager)
- 2 继承属性 (inherited attribute)
- 3 局部优化 (local optimization)
- 4 四元式 (quadruple)
- 5 E      + \* ( ) id

### 四、 单项选择题（每题 2 分，共 10 分）

1.B 2.D 3.B 4.D 5.C

### 五、 解答题（共 70 分）

1. (1)  $L(G) = \{0^m 1^n \mid m \geq 1\}$  共 2 分， $\geq$ 写成 $>$ 扣 1 分  
(2)  $S \Rightarrow 0S1 \Rightarrow 00S11 \Rightarrow 000111$ ，共 3 分， $\Rightarrow$ 写成 $\rightarrow$ 扣 1 分  
(3) 共 3 分，错处扣 0.5 分，扣完为止
2. (1) 空白表格也可以填写“错误”字样，共 4 分，错一个扣 0.5 分，扣完为止

	a	b	c	S(#)
S	$S \rightarrow aBc$	$S \rightarrow bAB$		
A	$A \rightarrow aAb$	$A \rightarrow b$		
B		$B \rightarrow b$	$B \rightarrow \varepsilon$	$B \rightarrow \varepsilon$

(2) 共 6 分，其中判断“baabbb 是该文法句子”为 2 分，其他错一个扣 0.5 分，扣完为止

符号栈	输入串	规则
-----	-----	----

\$S	baabbb\$	
\$BAb	baabbb\$	$S \rightarrow bAB$
\$BA	aabbb\$	
\$BbAa	aabbb\$	$A \rightarrow aAb$
\$BbA	abbb\$	
\$BbbAa	abbb\$	$A \rightarrow aAb$
\$BbbA	bbb\$	
\$Bbbb	bbb\$	$A \rightarrow b$
\$Bbb	bb\$	
\$Bb	b\$	
\$b	\$	$B \rightarrow \varepsilon$
\$	\$	success

3.(1) 共 6 分，其中判断“该文法为算符优先文法”为 2 分，其他错一个扣 0.5 分，扣完为止

	+	*	i
+	>	<	<
*	>	>	<
i	>	>	

(2) 共 4 分，错一个扣 0.5 分，扣完为止

	+	*	i
f	2	4	4
g	1	3	5

4. (1) 34242421，共 4 分，错一个扣 0.5 分

(2) 共 4 分，错一个扣 0.5 分，扣完为止

stack	Input string	action
\$	b(((aa)a)a)b\$	
\$b	(((aa)a)a)b\$	shift
\$b(	((aa)a)a)b\$abbb\$	shift
\$b((	(aa)a)a)b\$bbb\$	shift

\$b(((	aa)a)a)b\$bb\$	shift
\$b(((a	a)a)a)b\$\$	shift
\$b(((A	a)a)a)b\$	reduce, $A \rightarrow a$
\$b(((Aa	)a)a)b\$	shift
\$ b(((Aa)	a)a)b\$	shift
\$ b(((B	a)a)b\$	reduce, $B \rightarrow Aa$ )
\$b((A	a)a)b\$	reduce, $A \rightarrow (B$
\$b((Aa	)a)b\$	shift
\$b((Aa)	a)b\$	shift
\$b ((B	a)b\$	reduce, $B \rightarrow Aa$ )
\$b(A	a)b\$	reduce, $A \rightarrow (B$
\$b(Aa	)b\$	shift
\$b(Aa)	b\$	shift
\$b(B	b\$	reduce, $B \rightarrow Aa$ )
\$bA	b\$	reduce, $A \rightarrow (B$
\$bAb	\$	shift
\$\$	\$	reduce, $S \rightarrow bAb$
\$s\$		accept

5. 共 12 分，其中带注释的分析树、三地址码序列和四元式序列分别为 4 分，错一个序列扣 0.5 分，而错某点（某项）少于或等于 5 个扣 0.5 分

带注释语法树(略)

三地址码序列

M1: if (x>y) goto M2

goto M4

M2: if (a=b) goto M3

goto M1

M3: t1=2\*y

t2=t1+a

x=t2

四元式序列

100 (j>, x,y,102)

101 (j,-,-,108)

102 (j=,a,b,104)

103 (j,-,-,100)

104 (\*,2,y,t1)

105 (+,t1,a,t2)

106 (=,t2,-,x)

goto M1

107 (j,-,100)

M4:

108 (-,-,-)

6. 共 8 分，错一个扣 0.5 分，扣完为止

LD R1,0

ST S,R1

ST I,R1

L1: LD R1,X

SUB R1,R1,Y (OR SUB R1,Y)

BGTZ R1,L2

LD R2,a(R1)

ADD R2,R2,S (OR ADD R2,S)

ST Z,R2

LD R1,I (从这开始，下面的语句中的 R1 也可以全部变成 R2)

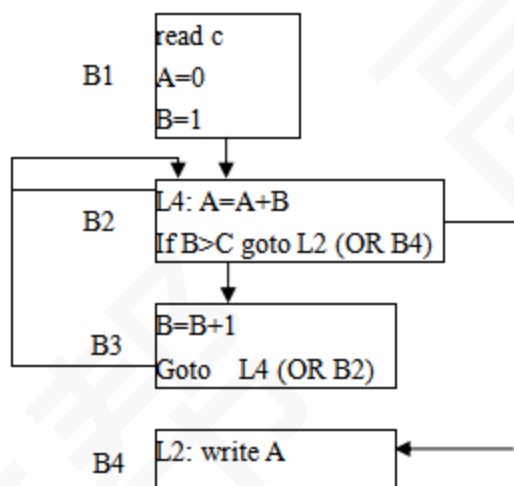
ADD R1,R1,1 (OR INC R1)

ST I,R1

BR L1

L2:

7. 共 6 分，基本块划分和流图各为 3 分，错一处扣 1 分，扣完为止



8. (1) 共 4 分, 错一项扣 1 分, 扣完为止

(2) 共 4 分, 错一项扣 1 分, 扣完为止

$t1=b*c$

$t2=b*d$

$t3=t1-t2$

$t4=i+1$  (or  $t4=i$ )

$b[t4]=t3$

#### 一、判断题:

1. 一个上下文无关文法的开始符, 可以是终结符或非终结符。

( )

2. 一个句型的直接短语是唯一的。

( )

3. 已经证明文法的二义性是可判定的。

( )

4. 每个基本块可用一个 DAG 表示。

( )

5. 每个过程的活动记录的体积在编译时可静态确定。

( )

6. 2 型文法一定是 3 型文法。

( )

7. 一个句型一定是句子。

( )

8. 算符优先分析法每次都是对句柄进行归约。

( )



- 9.采用三元式实现三地址代码时，不利于对中间代码进行优化。  
( )
- 10.编译过程中，语法分析器的任务是分析单词是怎样构成的。  
( )
- 11.一个优先表一定存在相应的优先函数。  
( )
- 12.目标代码生成时，应考虑如何充分利用计算机的寄存器的问題。  
( )
- 13.递归下降分析法是一种自下而上分析法。  
( )
- 14.并不是每个文法都能改写成 LL(1)文法。  
( )
- 15.每个基本块只有一个入口和一个出口。  
( )
- 16.一个 LL(1)文法一定是无二义的。  
( )
- 17.逆波兰法表示的表达式亦称前缀式。  
( )
- 18.目标代码生成时，应考虑如何充分利用计算机的寄存器的问題。  
( )
- 19.正规文法产生的语言都可以用上下文无关文法来描述。  
( )
- 20.一个优先表一定存在相应的优先函数。  
( )
- 21.3 型文法一定是 2 型文法。  
( )
- 22.如果一个文法存在某个句子对应两棵不同的语法树，则文法是二义性的。  
( )

## 二、填空题:

- 1.( )称为规范推导。
- 2.编译过程可分为 ( ), ( ), ( ), ( )和 ( )五个阶段。
- 3.如果一个文法存在某个句子对应两棵不同的语法树,则称这个文法是 ( )。
- 4.从功能上说,程序语言的语句大体可分为 ( )语句和 ( )语句两大类。
- 5.语法分析器的输入是 ( ),其输出是 ( )。
- 6.扫描器的任务是从 ( )中识别出一个个 ( )。
- 7.符号表中的信息栏中登记了每个名字的有关的性质,如 ( )等等。
- 8.一个过程相应的 **DISPLAY**表的内容为( )。
- 9.一个句型的最左直接短语称为句型的 ( )。
- 10.常用的两种动态存储分配办法是 ( )动态分配和 ( )动态分配。
- 11.一个名字的属性包括( )和( )。
- 12.常用的参数传递方式有 ( ), ( )和 ( )。
- 13.根据优化所涉及的程序范围,可将优化分成为 ( ), ( )和 ( )三个级别。
- 14.语法分析的方法大致可分为两类,一类是 ( )分析法,另一类是 ( )分析法。
- 15.预测分析程序是使用一张 ( )和一个 ( )进行联合控制的。
- 16.常用的参数传递方式有 ( ), ( )和 ( )。
- 17.一张转换图只包含有限个状态,其中有一个被认为是 ( )态;而且实际上至少要有一个 ( )态。
- 18.根据优化所涉及的程序范围,可将优化分成为 ( ), ( )和 ( )三个级别。
- 19.语法分析是依据语言的 ( )规则进行。中间代码产生是依据语言的 ( )规则进行的。
- 20.一个句型的最左直接短语称为句型的 ( )。
- 21.一个文法 **G**,若它的预测分析表 **M**不含多重定义,则该文法是 ( )文法。
- 22.对于数据空间的存储分配, **FORTRAN**采用 ( )策略, **PASCAL**采用 ( )策略。

- 23.如果一个文法存在某个句子对应两棵不同的语法树， 则称这个文法是( )。
- 24.最右推导亦称为( )，由此得到的句型称为( )句型。
- 25.语法分析的方法大致可分为两类，一类是( )分析法，另一类是( )分析法。
- 26.对于文法  $G$ ，仅含终结符号的句型称为( )。
- 27.所谓自上而下分析法是指( )。
- 28.语法分析器的输入是( )，其输出是( )。
- 29.局限于基本块范围的优化称( )。
- 30.预测分析程序是使用一张( )和一个( )进行联合控制的。
- 31.2 型文法又称为( )文法；3 型文法又称为( )文法。
- 32.每条指令的执行代价定义为( )。
- 33.算符优先分析法每次都是对( )进行归约。

### 三、名词解释题：

- 1.局部优化
- 2.二义性文法
- 3.DISPLAY 表
- 4.词法分析器
- 5.最左推导
- 6.语法
- 7.文法
- 8.基本块
- 9.语法制导翻译
- 10.短语
- 11.待用信息
- 12.规范句型
- 13.扫描器
- 14.超前搜索
- 15.句柄
- 16.语法制导翻译

17.规范句型

18.素短语

19.语法

20.待用信息

21.语义

#### 四、简答题:

1.写一个文法  $G$ , 使其语言为 不以 0 开头的偶数集。

2.已知文法  $G(S)$ 及相应翻译方案

$S \rightarrow aAb \quad \{\text{print "1"}\}$

$S \rightarrow a \quad \{\text{print "2"}\}$

$A \rightarrow AS \quad \{\text{print "3"}\}$

$A \rightarrow c \quad \{\text{print "4"}\}$

输入  $acab$ , 输出是什么?

3. 已知文法  $G(S)$

$S \rightarrow bAa$

$A \rightarrow (B \mid a$

$B \rightarrow Aa)$

写出句子  $b(aa)b$  的规范归约过程。

4. 考虑下面的程序:

...

procedure  $p(x, y, z);$

begin

$y := x + y;$

$z := z * z;$

end

begin

$A := 2;$

$B := A * 2;$

$P(A, A, B);$

Print A, B

end.

试问，若参数传递的方式分别采用传地址和传值时，程序执行后输出 A, B 的值是什么？

5.文法  $G(S)$

$S \rightarrow dAB$

$A \rightarrow aA \mid a$

$B \rightarrow Bb \mid \varepsilon$

描述的语言是什么？

6.证明文法  $G(S)$

$S \rightarrow SaS \mid \varepsilon$

是二义性的。

7.已知文法  $G(S)$

$S \rightarrow BA$

$A \rightarrow BS \mid d$

$B \rightarrow aA \mid bS \mid c$

的预测分析表如下

	a	b	c	d	#
S	$S \rightarrow BA$	$S \rightarrow BA$	$S \rightarrow BA$		
A	$A \rightarrow BS$	$A \rightarrow BS$	$A \rightarrow BS$	$A \rightarrow d$	
B	$B \rightarrow aA$	$B \rightarrow bS$	$B \rightarrow c$		

给出句子 adccd 的分析过程。

8.写一个文法  $G$ , 使其语言为  $L(G) = \{a^l b^m c^l a^m b^n \mid l \geq 0, m \geq 1, n \geq 2\}$

9.已知文法  $G(S)$ :

$S \rightarrow a \mid (T)$

$T \rightarrow T, S \mid S$

的优先关系表如下:

关系	a	(	)	,
a			>	>
(	<	<	=	<
)			>	>
,	<	<	>	>

请计算出该优先关系表所对应的优先函数表。

10.何谓优化？按所涉及的程序范围可分为哪几级优化？

11.目标代码有哪几种形式？生成目标代码时通常应考虑哪几个问题？

12.一字母表 $\Sigma=\{a, b\}$ ，试写出 $\Sigma$ 上所有以 a 为首的字组成的正规集相对应的正规式。

13.基本的优化方法有哪几种？

14.写一个文法 G, 使其语言为  $L(G)=\{ab^nc^n \mid n \geq 0\}$

15.考虑下面的程序：

```

...
procedure p(x, y, z);
begin
    y:=y+z;
    z:=y*z+x
end;
begin
    a:=2;
    b:=3;
    p(a+b, b, a);
    print a
end.

```

试问，若参数传递的方式分别采用传地址和传值时，程序执行后输出 a 的值是什么？

16.写出表达式  $a + b * (c - d) / e$  的逆波兰式和三元序列。

17.证明文法 G(A)

$A \rightarrow AA \mid (A) \mid \varepsilon$

是二义性的。

18. 令  $\Sigma = \{a, b\}$ , 则正规式  $a^*b|b^*a$  表示的正规集是什么?

19. 何谓 DISPLAY 表? 其作用是什么?

20. 考虑下面的程序:

```
...  
procedure p(x, y, z);  
begin  
  y:=y+2;  
  z:=z+x;  
end  
begin  
  a:=5;  
  b:=2;  
  p(a+b, a-b, a);  
  print a  
end.
```

试问, 若参数传递的方式分别采用传地址和传值时, 程序执行后输出  $a$  的值是什么?

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

22. 写出表达式  $a := (b+c) * e + (b+c) / f$  的逆波兰式和三元序列。

23. 一个文法  $G$  别是 LL(1) 文法的充要条件是什么?

24. 已知文法  $G[S]$

$$S \rightarrow S * aF \mid aF \mid * aF$$
$$F \rightarrow + aF \mid + a$$

消除文法左递归和提公共左因子。

25. 符号表的作用是什么? 符号表查找和整理技术有哪几种?

## 五、计算题:

1. 设文法  $G(S)$ :

$$S \rightarrow \wedge \mid a \mid (T)$$
$$T \rightarrow T, S \mid S$$

- (1) 消除左递归;
- (2) 构造相应的 FIRST 和 FOLLOW 集合;
- (3) 构造预测分析表

2. 语句 if E then S

- (1) 改写文法, 使之适合语法制导翻译;
- (2) 写出改写后产生式的语义动作。

3. 设文法  $G(S)$ :

$S \rightarrow (T) \mid a$

$T \rightarrow T+S \mid S$

- (1) 计算 FIRSTVT 和 LASTVT;
- (2) 构造优先关系表。

4. 设某语言的 for 语句的形式为

for  $i := E^{(1)}$  to  $E^{(2)}$  do S

其语义解释为

$i := E^{(1)}$

LIMIT :=  $E^{(2)}$

again: if  $i \leq$  LIMIT then

Begin

S;

$i := i + 1$

goto again

End;

- (1) 写出适合语法制导翻译的产生式;
- (2) 写出每个产生式对应的语义动作。

5. 把语句

while  $a < 10$  do

if  $c > 0$  then  $a := a + 1$

else  $a := a * 3 - 1$ ;

翻译成四元式序列。



6.设有基本块

$D := A - C$

$E := A * C$

$F := D * E$

$S := 2$

$T := A - C$

$Q := A * C$

$G := 2 * S$

$J := T * Q$

$K := G * 5$

$L := K + J$

$M := L$

假设基本块出口时只有 **M** 还被引用，请写出优化后的四元序列。

7.已知文法  $G(S)$

$S \rightarrow a \mid \wedge \mid (T)$

$T \rightarrow T, S \mid S$

(1) 给出句子  $(a, (a, a))$  的最左推导；

(2) 给出句型  $((T, S), a)$  的短语、直接短语、句柄。

8.对于 C 语言 **do S while E** 语句

(1) 改写文法，使之适合语法制导翻译；

(2) 写出改写后产生式的语义动作。

9.已知文法  $G(S)$

$S \rightarrow aAcBe$

$A \rightarrow Ab \mid b$

$B \rightarrow d$

(1) 给出句子  $abbcd e$  的最左推导及画出语法树；

(2) 给出句型  $aAbcd e$  的短语、素短语。

10.设文法  $G(S)$ :

$S \rightarrow (T) \mid aS \mid a$

$T \rightarrow T, S \mid S$

- (1) 消除左递归和提公共左因子;
- (2) 构造相应的 FIRST 和 FOLLOW 集合;
- (3) 构造预测分析表。

11. 把语句

if  $X > 0 \vee Y < 0$

then while  $X > 0$  do  $X := A * 3$

else  $Y := B + 3$ ;

翻译成四元式序列。

12. 已知文法  $G(S)$

$E \rightarrow E + T \mid T$

$T \rightarrow T * F \mid F$

$F \rightarrow (E) \mid i$

- (1) 给出句型  $(i+i)*i+i$  的最左推导及画出语法树;
- (2) 给出句型  $(E+T)*i+F$  的短语, 素短语和最左素短语。

13. 设文法  $G(S)$ :

$S \rightarrow T \mid S \vee T$

$T \rightarrow U \mid T \wedge U$

$U \rightarrow i \mid \neg U$

- (1) 计算 FIRSTVT 和 LASTVT;
- (2) 构造优先关系表。

## 参考答案

### 一、是非题:

- 1.×    2.×    3.×    4.√    5.√    6.×    7.×    8.×    9.√    10.×    11.×  
12.√    13.×    14.√    15.√    16.√    17.×    18.√    19.√    20.×    21.√    22.√

### 二、填空题:

- 1.(最右推导)
- 2.(词法分析), (语法分析), (中间代码生成), (代码优化), (目标代码生成)
- 3.(二义性的)
- 4.(执行性), (说明性)
- 5.(单词符号), (语法单位)。
- 6.(源程序), (单词符号)
- 7.(类型、种属、所占单元大小、地址)
- 8.(现行活动记录地址和所有外层最新活动记录的地址)
- 9.(句柄)
- 10.(栈式), (堆式)
- 11.(类型), (作用域)
- 12.(传地址), (传值), (传名)
- 13.(局部优化), (循环优化), (全局优化)
- 14.(自上而下), (自下而上)
- 15.(分析表), (符号栈)
- 16.(传地址), (传值), (传名)
- 17.(初), (终)
- 18.(局部优化), (循环优化), (全局优化)
- 19.(语法), (语义)
- 20.(句柄)
- 21.(LL(1) 文法)
- 22.(静态), (动态)
- 23.(二义性文法)
- 24.(规范推导), (规范)
- 25.(自上而下), (自下而上)
- 26.(句子)
- 27.(从开始符号出发, 向下推导, 推出句子)

- 28.(单词符号), (语法单位)
- 29.(局部优化)
- 30.(分析表), (符号栈)
- 31.(上下文无关文法), (正规)
- 32.(指令访问主存次数加 1)
- 33.(最左素短语)

### 三、名词解释题:

- 1.局部优化——局限于基本块范围的优化称。
- 2.二义性文法——如果一个文法存在某个句子对应两棵不同的语法树, 则称这个文法是二义性文法。
- 3.DISPLAY 表——过程的嵌套层次显示表, 记录该过程的各外层过程的最新活动记录的起始地址。
- 4.词法分析器——执行词法分析的程序。
- 5.最左推导——任何一步  $\alpha \Rightarrow \beta$  都是对  $\alpha$  中的最右非终结符替换。
- 6.语法——一组规则, 用它可形成和产生一组合式的程序。
- 7.文法——描述语言的语法结构的形式规则。
- 8.基本块——指程序中一顺序执行的语句序列, 其中只有一个入口和一个出口, 入口就是其中的第一个语句, 出口就是其中的最后一个语句。
- 9.语法制导翻译——在语法分析过程中, 根据每个产生式所对应的语义子程序进行翻译的办法叫做语法制导翻译。
- 10.短语——令  $G$  是一个文法,  $S$  划文法的开始符号, 假定  $\alpha\beta\delta$  是文法  $G$  的一个句型, 如果有  $S \xRightarrow{*} \alpha A \delta$  且  $A \xRightarrow{+} \beta$ , 则称  $\beta$  是句型  $\alpha\beta\delta$  相对非终结符  $A$  的短语。
- 11.待用信息——如果在一个基本块中, 四元式  $i$  对  $A$  定值, 四元式  $j$  要引用  $A$  值, 而从  $i$  到  $j$  之间没有  $A$  的其它定值, 则称  $j$  是四元式  $i$  的变量  $A$  的待用信息。
- 12.规范句型——由规范推导所得到的句型。
- 13.扫描器——执行词法分析的程序。
- 14.超前搜索——在词法分析过程中, 有时为了确定词性, 需超前扫描若干个字符。
- 15.句柄——一个句型的最左直接短语。
- 16.语法制导翻译——在语法分析过程中, 根据每个产生式所对应的语义程序进行翻译的方法叫做语法制导翻译。
- 17.规范句型——由规范推导所得到的句型。
- 18.素短语——素短语是指这样一个短语, 至少含有一个终结符, 并且, 除它自身外不再含任何更

小的素短语。

19.语法-----是组规则，用它可形成和产生一个合式的程序。\_

20.待用信息-----如果在一个基本块中，四元式  $i$  对  $A$  定值，四元式  $j$  要引用  $A$  值，而从  $i$  到  $j$  之间没有  $A$  的其它定值，则称  $j$  是四元式  $i$  的变量  $A$  的待用信息。

21.语义-----定义程序的意义的一组规则。

#### 四、简答题:

1.所求文法是  $G[S]$ :

$S \rightarrow AB \mid BA \mid 0$

$A \rightarrow AD \mid C$

$B \rightarrow 2 \mid 4 \mid 6 \mid 8$

$C \rightarrow 1 \mid 3 \mid 5 \mid 7 \mid 9 \mid B$

$D \rightarrow 0 \mid C$

2.输出是 4231

3.句子  $b(aa)b$  的规范归约过程:

步骤	符号栈	输入串	动作
0	#	$b(aa)b$ #	预备
1	#b	$(aa)b$ #	移进
2	#b(	$aa)b\#$	移进
3	#b(a	$a)b\#$	移进
4	#b(A	$a)b\#$	归约
5	#b(Ma	$)b\#$	移进
6	#b(Ma)	$b\#$	移进
7	#b(B	$b\#$	归约
8	#bA	$b\#$	归约
9	#bAb	$\#$	移进
10	#S	$\#$	接受

4.传地址  $A=6, B=16$

传值  $A=2, B=4$

5. $L(G)=\{da^n b^m \mid n>0, m \geq 0\}$

6.证明:

因为文法  $G[S]$  存在句子  $aa$  有两个不同的最左推导, 所以文法  $G[S]$  是二义性的。

$$S \Rightarrow SaS \Rightarrow SaSaS \Rightarrow aSaS \Rightarrow aaS \Rightarrow aa$$
$$S \Rightarrow SaS \Rightarrow aS \Rightarrow aSaS \Rightarrow aaS \Rightarrow aa$$

7. 句子  $adccd$  的分析过程:

步骤	符号栈	输入串	产生式
0	#S	adccd#	
1	#AB	adccd#	$S \rightarrow BA$
2	#AAa	adccd#	$B \rightarrow aA$
3	#AA	dccd#	
4	#Ad	dccd#	$A \rightarrow d$
5	#A	ccd#	
6	#SB	ccd#	$A \rightarrow BS$
7	#Sc	ccd#	$B \rightarrow c$
8	#S	cd#	
9	#AB	cd#	$B \rightarrow c$
10	#Ac	d#	
11	#A	d#	
12	#d	d#	$A \rightarrow d$
13	#	#	

8. 所求文法是  $G[S]$ :

$$S \rightarrow AB$$
$$A \rightarrow aAc \mid D$$
$$D \rightarrow bD \mid b$$
$$B \rightarrow aBb \mid aabb$$

9.

函数	a	(	)	,
f	4	2	4	4
g	5	5	2	3

10. 优化: 对程序进行各种等价变换, 使得从变换后的程序出发, 能产生更有效的目标代码。

三种级别: 局部优化、循环优化、全局优化

11.目标代码通常采用三种形式：机器语言，汇编语言，待装配机器语言模块。

应着重考虑的问题：

- (1)如何使生成的目标代码较短；
- (2)如何充分利用寄存器，以减少访问内存次数；
- (3)如何充分利用指令系统的特点。

12.正规式  $a(a|b)^*$ 。

13.删除多余运算，代码外提，强度削弱，变换循环控制条件，合并已知量，复写传播和删除无用赋值。

14.文法  $G[S]$ :

$S \rightarrow aB | a$

$B \rightarrow bc | bBc$

15.传值  $a=2$

传地址  $a=15$

16.逆波兰式:  $abcd-*e/+$

三元序列:	op	arg1	arg2
(1)	-	c	d
(2)	*	b	(1)
(3)	/	(2)	e
(4)	+	a	(3)

17.证明:

因为文法  $G[S]$ 存在句子 0 有两个不同的最左推导，所以文法  $G[S]$ 是二义性的。

$A \Rightarrow AA \Rightarrow (A)A \Rightarrow ()A \Rightarrow ()$

$A \Rightarrow AA \Rightarrow A \Rightarrow (A) \Rightarrow ()$

18.  $(a^*b|b^*a) = \{a, b, ab, ba, aab, bba, \dots\}$

19.Display 表: 嵌套层次显示表

由于过程嵌套允许内层过程引用外层过程定义的数据，因此，当一个过程运行时必须跟踪它的所有外层过程的最新活动记录起始地址，display 表就是用于登记每个外层过程的最新活动记录起始地址。

20.传地址  $a=12$

传值  $a=5$

21.所求文法是  $G[S]$ :

$S \rightarrow AC$

$A \rightarrow aaAbb \mid ab$

$C \rightarrow ccC \mid cc$

22.逆波兰式  $abc+e*bc+f/+:=$

三元序列	op	arg1	arg2
(1)	+	b	c
(2)	*	(1)	e
(3)	+	b	c
(4)	/	(3)	f
(5)	+	(2)	(4)
(6)	:=	a	(5)

23.一个文法  $G$  是  $LL(1)$  文法的充要条件是:

(1)  $FIRST(\alpha) \cap FIRST(\beta) = \Phi$

(2) 如果  $\beta \Rightarrow^* \epsilon$ ,  $FIRST(\alpha) \cap FOLLOW(A) = \Phi$

24.消除左递归

$S \rightarrow aFS' \mid *aFS'$

$S' \rightarrow *aFS' \mid \epsilon$

$F \rightarrow +aF \mid +a$

提公共左因子,文法  $G'(S)$

$S \rightarrow aFS' \mid *aFS'$

$S' \rightarrow *aFS' \mid \epsilon$

$F \rightarrow +aF'$

$F' \rightarrow F \mid \epsilon$

25.作用: 登记源程序中出现的各种名字及其信息, 以及了解各阶段的进展状况。

主要技术: 线性表, 对折查找, 杂凑技术。

## 五、计算题:

1.

(1)消除左递, 文法变为  $G'[S]$ :

$S \rightarrow \wedge \mid a \mid (T)'$

$T \rightarrow ST' \mid S$

$T' \rightarrow ,ST' \mid \epsilon$

此文法无左公共左因子。

(2)构造相应的  $FIRST$  和  $FOLLOW$  集合:



$FIRST(S) = \{a, \wedge, ()\}$ ,  $FOLLOW(S) = \{\#, \dots\}$

$FIRST(T) = \{a, \wedge, ()\}$ ,  $FOLLOW(T) = \{\}$

$FIRST(T') = \{., \varepsilon\}$ ,  $FOLLOW(T') = \{\}$

(3)构造预测分析表:

	a	$\wedge$	(	)	,	#
S	$S \rightarrow a$	$S \rightarrow \wedge$	$S \rightarrow (T)'$			
T	$T \rightarrow ST'$	$T \rightarrow ST'$	$T \rightarrow ST'$			
T'				$T' \rightarrow \varepsilon$	$T' \rightarrow ,ST'$	

2. (1)

$C \rightarrow \text{if } E \text{ then}$

$S \rightarrow CS^{(1)}$

(2)

$C \rightarrow \text{if } E \text{ then } \{BACK(E.TC, NXQ); C.chain := E.FC\}$

$S \rightarrow CS^{(1)} \quad \{S.chain := MERG(C.Chain, S^{(1)}.Chain)\}$

3. (1)  $FIRSTVT(S) = \{a, ()\}$

$FIRSTVT(T) = \{+, a, ()\}$

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

$LASTVT(T) = \{+, a, )\}$

(2)

	a	+	(	)
a		>		>
+	<	>	<	>
(	<	<	<	=
)		>	>	>

4. (1)  $F \rightarrow \text{for } i := E^{(1)} \text{ to } E^{(2)} \text{ do}$

$S \rightarrow FS^{(1)}$

(2)  $F \rightarrow \text{for } i := E^{(1)} \text{ to } E^{(2)} \text{ do}$

$\{GEN(=, E^{(1)}.place, \_, entry(i));$

$F.place := entry(i);$

$LIMIT := Newtemp;$

$GEN(=, E^{(2)}.place, \_, LIMIT);$

```

Q:=NXQ;
F.QUAD:=q;
GEN(j≤, entry(i), LIMIT, q+2)
F.chain:=NXQ;
GEN(j, _, _, 0)}
S→FS(1)
{BACKPATCH(S(1).chain, NXQ);
  GEN(+, F.place, 1, F.place);
  GEN(j, _, _, F.QUAD);
  S.chain:=F.chain
}

```

5. (1) (j<, a, '10', (3))
- (2) (j, \_, \_, (12))
- (3) (j>, c, '0', (5))
- (4) (j, \_, \_, (8))
- (5) (+, a, '1', T1))
- (6) (:=, T1, \_, a)
- (7) (j, \_, \_, (1))
- (8) (\*, a, '13', T2)
- (9) (-, T2, '1', T3)
- (10) (:=, T3, \_, a)
- (11) (j, \_, \_, (1))

#### 6. 优化后的四元序列

```

D:=A-C
E:=A*C
F:=D*E
M:=F+20

```

#### 7. 最左推导

$S \Rightarrow (T) \Rightarrow (T, S) \Rightarrow (S, S) \Rightarrow (a, S) \Rightarrow (a, (T)) \Rightarrow (a, (T, S)) \Rightarrow (a, (S, S)) \Rightarrow (a, (a, S)) \Rightarrow (a, (a, a))$

短语

$((T, S), a)$

$(T, S), a$

(T,S)

T,S

a

直接短语

T,S

a

句柄

T,S

8.(1)

$S \rightarrow \text{do } M_1 \text{ } S_1 \text{ while } M_2 \text{ } E$

$M \rightarrow \varepsilon$

(2)

$M \rightarrow \varepsilon$  {M.quad=nestquad;}

$S \rightarrow \text{do } M_1 \text{ } S_1 \text{ while } M_2 \text{ } E$  {backpatch( $s_1$ .nextlist,  $M_2$ .quad);  
backpatch( $E$ .truelist,  $M_1$ .quad);  
 $S$ .nextlist= $E$ .falelist;  
}

9.(1)  $S \Rightarrow aAcBe \Rightarrow AAbcBe \Rightarrow abbcBe \Rightarrow abbcde$

(2) 短语: aAbcde, Ab, d

素短语: Ab, d

10.(1)  $S \rightarrow (L) \mid aS'$

$S' \rightarrow S \mid \varepsilon$

$L \rightarrow SL'$

$L' \rightarrow ,SL' \mid \varepsilon$

(2)  $\text{FIRST}(S) = \{a, ( \}$   $\text{FIRST}(S') = \{a, (, \varepsilon \}$

$\text{FIRST}(L) = \{a, ( \}$   $\text{FIRST}(L') = \{,, \varepsilon \}$

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

$\text{FOLLOW}(L) = \{ ) \}$   $\text{FOLLOW}(L') = \{ ) \}$

(3)

	(	)	a	,	#
S	$S \rightarrow (L)$		$S \rightarrow aS'$		
S'	$S' \rightarrow S$	$S' \rightarrow \varepsilon$	$S' \rightarrow S$	$S' \rightarrow \varepsilon$	$S' \rightarrow \varepsilon$
L	$L \rightarrow SL'$		$L \rightarrow SL'$	$L' \rightarrow ,SL'$	
L'		$L' \rightarrow \varepsilon$			

11.(1)  $(j > X, 0, (5))$

(2)  $(j, \_, \_, (3))$

(3)  $(j < Y, 0, (5))$

(4)  $(j, \_, \_, (11))$

(5)  $(j > 0, X, 0, (7))$

(6)  $(j, \_, \_, (7))$

(7)  $(*, A, 3, T_1)$

(8)  $(:=, T_1, \_, N)$

(9)  $(j, \_, \_, (5))$

(10)  $(j, \_, \_, (13))$

(11)  $(+, B, 3, T_2)$

(12)  $(:=, T_2, \_, Y)$

12.(1)  $E \Rightarrow E+T \Rightarrow T+T \Rightarrow T * F+T \Rightarrow F * F+T \Rightarrow (E) * F+T \Rightarrow (E+T) * F+T \Rightarrow (T+T) * F+T$

$\Rightarrow (F+T) * F+T \Rightarrow (i+T) * F+T \Rightarrow (i+F) * F+T \Rightarrow (i+i) * F+T \Rightarrow (i+i) * i+T$

$\Rightarrow (i+i) * i+F \Rightarrow (i+i) * i+i$

(2) 短语  $i, F, E+T, (E+T), (E+T) * i, (E+T) * i+F$

素短语  $i, E+T$

最左素短语  $E+T$

13.(1)  $FIRSTVT(S) = \{ \vee, \wedge, i, - \}$

$FIRSTVT(T) = \{ \wedge, i, - \}$

$FIRSTVT(U) = \{ i, - \}$

$LASTVT(S) = \{ \vee, \wedge, i, - \}$

$LASTVT(T) = \{ \wedge, i, - \}$

$LASTVT(U) = \{ i, - \}$

(2)

	i	$\vee$	$\wedge$	-
s		.>	.>	
$\vee$	<.	.>	<.	<.
$\wedge$	<.	.>	.>	<.
-	<.	.>	.>	<.