

诚信考试，诚信做人。

姓名: _____

学号: _____

订正: _____

班级: _____

专业: _____

学年: _____

广东工业大学考试试卷 (A)

2021 -- 2022 学年度第 2 学期

课程名称: 编译原理 学分 _____ 试卷满分 100 分

考试形式: 闭卷 (开卷或闭卷)

题号	一	二	三	四	五	六	七	八	九	十	总分
评卷得分											
评卷签名											
复核得分											
复核签名											

一 (6 分) 构造一个 2 型文法 G, 使其语言为

$$L(G)=\{ a^n c^m b^n \mid n, m \geq 1 \text{ 且为 } n \text{ 偶数, } m \text{ 为奇数} \}$$

二 (12 分) 已知文法 G[S]:

$$S \rightarrow TS|T$$

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

- (1) 写出句型 (()) () 的最左推导;
- (2) 画出句型 (()) () 的语法树;
- (3) 写出句型 (()) () 的全部短语、直接短语和句柄。

三 (6 分) 已知文法 G[S] 为:

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

$$T \rightarrow T, S | S$$

求消除文法 G[S] 中的左递归后的文法 G'[S]。

四 (10 分) 已知字母表 $\Sigma = \{a, b, c\}$, 试求在该字母表上的仅包括一个 b 的所有串的集合相对应的正规式, 并画出 NFA。

五 (10 分) 已知文法 $G(S)$

$$S \rightarrow a \mid \wedge \mid (T) \quad T \rightarrow ST' \quad T' \rightarrow , ST' \mid \epsilon$$

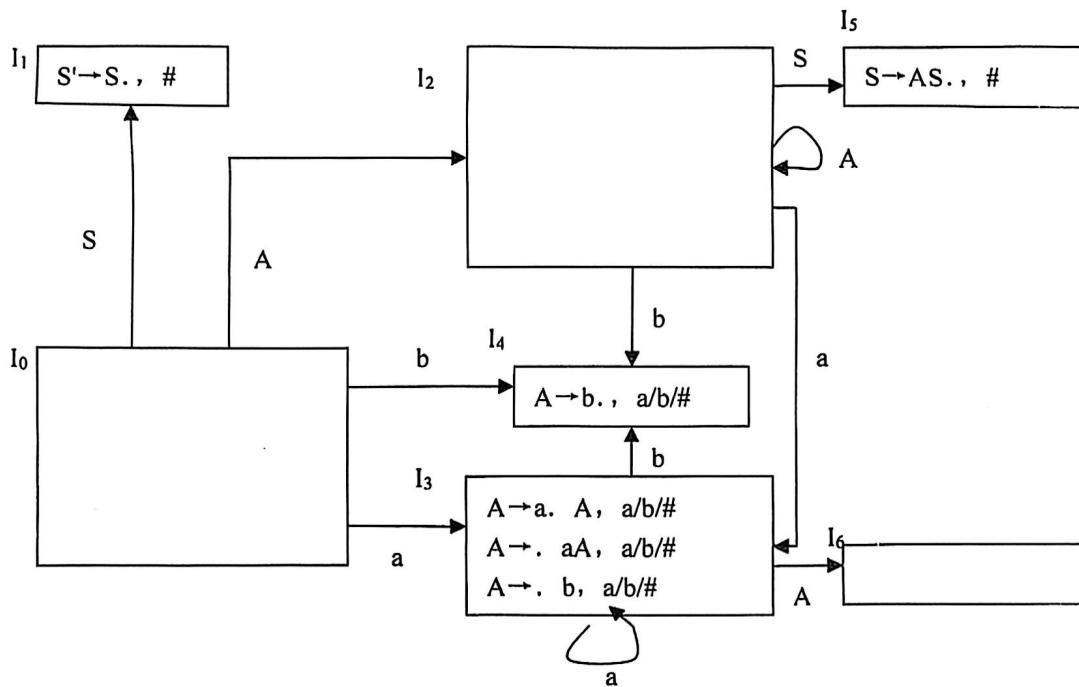
(1) 填空完成下表中的 select 集,并判断该文法是否为 LL(1) 的文法。

规则	Select 集
$S \rightarrow a$	
$S \rightarrow \wedge$	
$S \rightarrow (T)$	
$T \rightarrow ST'$	
$T' \rightarrow , ST'$	
$T' \rightarrow \epsilon$	

(2) 构造非终结符 T 的递归下降分析子程序 $\text{ParseT1}()$ 。

六 (12 分) 已知拓广文法 $G[S']$: $S' \rightarrow S \quad S \rightarrow AS \mid \epsilon \quad A \rightarrow aA \mid b$

(1) 试补齐下图以 LR(1) 项目集为状态的识别活前缀的有穷自动机;



(2) 试判断文法是否是 LR(1) 文法, 并说明理由。

七 (12 分) 已知文法 $G[S]$: $S' \rightarrow S \quad S \rightarrow 0S0 \mid 1S1 \mid a$

为 S 引入长度属性 $S.\text{len}$, 试为每个产生式增加语义规则, 使属性文法按照语法制导翻译的方法计算, 并输出句子的长度。例如: 输入 0100a0010, 语法制导翻译的结果是



五 (10 分) 已知文法 $G(S)$

$$S \rightarrow a \mid \wedge \mid (T) \quad T \rightarrow ST' \quad T' \rightarrow , ST' \mid \epsilon$$

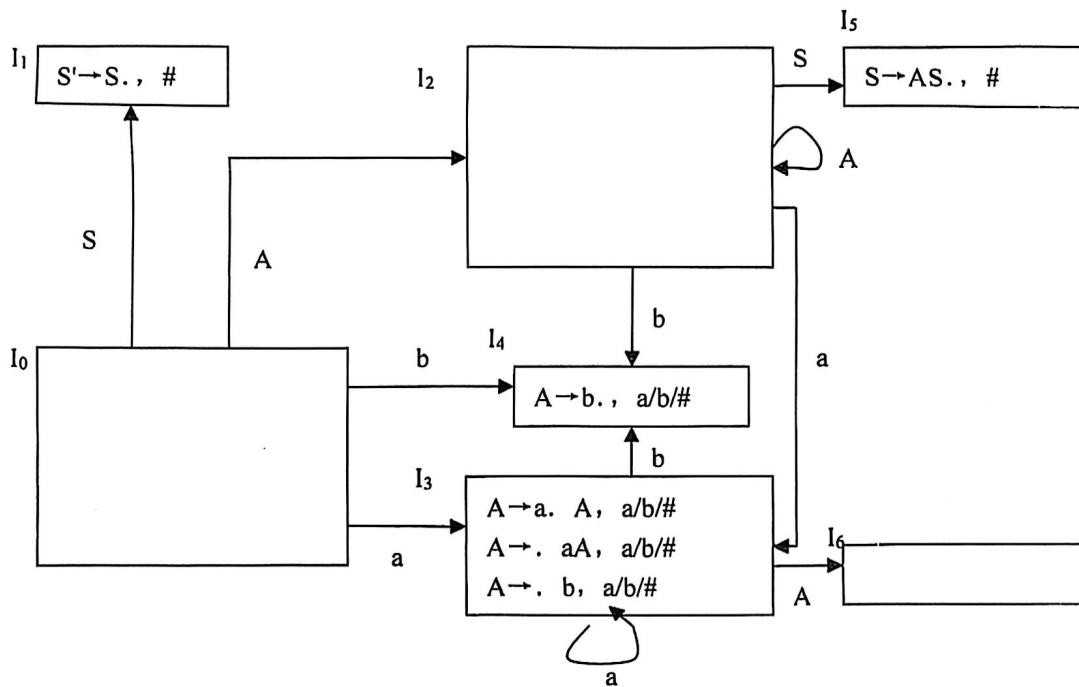
(1) 填空完成下表中的 select 集,并判断该文法是否为 LL(1) 的文法。

规则	Select 集
$S \rightarrow a$	
$S \rightarrow \wedge$	
$S \rightarrow (T)$	
$T \rightarrow ST'$	
$T' \rightarrow , ST'$	
$T' \rightarrow \epsilon$	

(2) 构造非终结符 T 的递归下降分析子程序 $\text{ParseT1}()$ 。

六 (12 分) 已知拓广文法 $G[S']$: $S' \rightarrow S \quad S \rightarrow AS \mid \epsilon \quad A \rightarrow aA \mid b$

(1) 试补齐下图以 LR(1) 项目集为状态的识别活前缀的有穷自动机;



(2) 试判断文法是否是 LR(1) 文法, 并说明理由。

七 (12 分) 已知文法 $G[S]$: $S' \rightarrow S \quad S \rightarrow 0S0 \mid 1S1 \mid a$

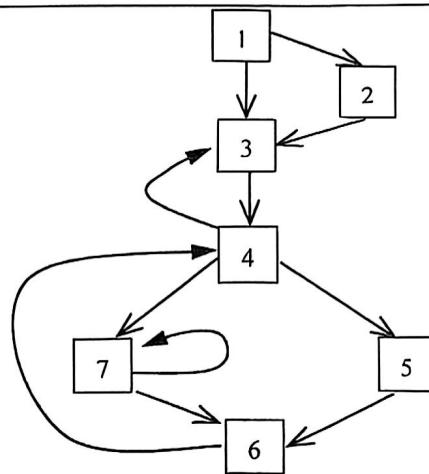
为 S 引入长度属性 $S.\text{len}$, 试为每个产生式增加语义规则, 使属性文法按照语法制导翻译的方法计算, 并输出句子的长度。例如: 输入 0100a0010, 语法制导翻译的结果是



九 (12 分)

1. 已知流图如右图所示：

- (1) 求出流图中结点 2~7 的支配结点集 $D(n)$ ；
- (2) 求出流图中的回边；
- (3) 求出流图中各个回边对应的循环。



十 (6 分) 下面是 PL0 语言程序的表达式部分代码生成片段 (部分代码用省略号代替)，功能是生成加法和减法指令，其中 $gendo(a, b, c)$ 为生成指令的函数，表达式的 EBNF 描述为 $<\text{表达式}> ::= [+|-]<\text{项}> \{ (+|-)<\text{项}> \}$ 。请参考附表的“类 P-code 虚拟机的部分指令系统”进行填空。

附表：类 P-code 虚拟机的部分指令系统

OPR 0 0	过程调用结束后, 返回调用点并退栈
OPR 0 1	栈顶元素取反
OPR 0 2	次栈顶与栈顶相加, 退两个栈元素, 结果值进栈
OPR 0 3	次栈顶减去栈顶, 退两个栈元素, 结果值进栈
OPR 0 4	次栈顶乘以栈顶, 退两个栈元素, 结果值进栈
OPR 0 5	次栈顶除以栈顶, 退两个栈元素, 结果值进栈
OPR 0 6	栈顶元素的奇偶判断, 结果值在栈顶

```
int expression(bool* fsys, int* ptx, int lev)
{
    enum symbol addop; /* 用于保存正负号 */
    bool nxtlev[symnum];
    if (sym == plus || sym == minus)
    {
        addop = sym;
        .....
        if (addop == minus)
```



```
{  
    gendo(_____(1)_____);  
}  
}  
.....  
  
while (sym == plus || sym == minus)  
{  
    addop = sym;  
    .....  
    if (addop == plus)  
    {  
        gendo(_____(2)_____);  
    }  
    else  
    {  
        gendo(_____(3)_____);  
    }  
    .....
```

