

A 卷

2019–2020 学年第 1 学期
(2019 秋季)

《编译原理与技术》
期末考试卷

班级_____学号_____

姓名_____成绩_____

2020 年 1 月 2 日



《编译原理与技术》
期末考试卷

注意事项：1. 所有答案请直接写在题目中，另附纸无效。
2. 交卷时请以班为单位交卷。

题号	一	二	三	四					总分
				1	2	3	4	5	
成绩									
阅卷人 签字									
任课教师签字									

- 题目：
- 一、填空题.....(9 分)
- 二、判断题.....(10 分)
- 三、单选题.....(21 分)
- 四、综合题
1.(10 分)
2.(6 分)
3.(14 分)
4.(16 分)
5.(14 分)

一、填空题（每空 1 分，共 9 分）

1. 在编译过程的五个阶段中，词法分析的输出是 token 序列，语法分析的输出是 抽象语法树。
2. 根据乔姆斯基对文法的分类，正则文法是 3 型文法，它可以被 有限自动机 接受。
3. 算符优先分析过程每次规约的是 取右最左。
4. 向输入文法插入动作符号后得到的文法是 翻译 文法，这个文法推导所产生的终结符号串称为 动作 序列。
5. 根据能否在编译阶段确定所需数据空间大小，可将运行时的存储分配方式分为 静态存储分配 和 动态。

二、判断题（每题 1 分，共 10 分）

1. 整个编译过程中只对源代码做 一次 从头到尾扫描的编译器，就是“一遍扫描的编译器”。
☒
2. 文法 G 所描述的语言，就是文法 G 的终结符集合 V_t 的闭包 V_t^* 。
☒
3. NFA 的接受状态可以多于一个，但 DFA 只能有一个。
☒
4. 算符优先分析过程中，栈顶运算符优先级小于栈外输入运算符时，执行入栈操作；栈顶运算符优先级大于栈外输入运算符时，执行出栈规约操作；其他情况说明遇到了错误。
☒
5. 属性翻译文法中综合属性的求值是 自下向上 的；而继承属性的求值是 自上向下 的。
☒
6. First 集合可以包含 ϵ ，Follow 集中不可以包含 ϵ 。
☒
7. 规范句型的 活前缀 不一定是唯一的。
☒
8. LL(1) 文法和 SLR(1) 文法都 一定 是无二义性的。
☒
9. 与 机器 有关的优化一般是在中间代码上进行的。
☒
10. 对于右侧的代码块：语句 `return j+1` 等价于 `return 1`，
语句 `return j-1` 等价于 `return i-1`；因此可以在优化时应用复写传播改为：
☒

```
int fun(int i) {
    int j = i;
    if (j == 0) return j+1;
    else return j-1;
}
```

```
if (i == 0) return 1;
else return i-1;
```


5. 记正则表达式 r 定义的语言为 $L(r)$ ，下列选项中正确的是 C。

A. $L(a)L(a^*) = L(a^+)$ a^+

B. $L(a|b) = L(a)L(b)$ \cup

C. $L(b(ab)^*) = L((ba)^*b)$ $b a b a b a b$

D. $L(ab|c) = L(a)L(b|c)$ $L(ab) \cup L(c)$

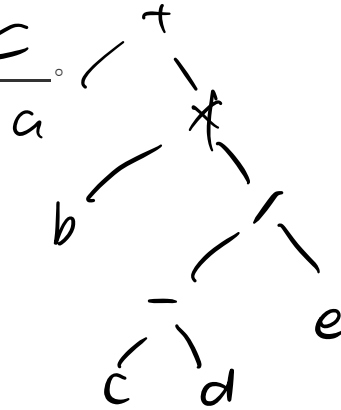
6. 算术表达式 $a + b * (c - d) / e$ 的后缀表示为 C。

A. $a b c d - * e / +$

B. $b c d - e / * a +$

C. $a b c d - e / * +$

D. $c d - b * e / a +$



7. 如下是某函数的中间代码表示。

```

(1)  a = 0
(2)  b = 1
(3)  i = 0
(4)  if i >= 10 goto (10)
(5)  c = a + b
(6)  a = b
(7)  b = c
(8)  i = i + 1
(9)  goto (4)
(10) v1 = i
(11) v2 = b
    
```

下列中间代码序列中，D 是一个基本块。

A. (1) (2) (3) (4)

B. (4) (5) (6) (7) (8) (9)

C. (5) (6) (7) (8)

D. (10) (11)

四、综合题（共 60 分）

1. （10 分）有文法 $G[E]$:

$$E ::= TT'$$

$$T ::= FF'$$

$$T' ::= \varepsilon \mid +TT'$$

$$F ::= v \mid i$$

$$F' ::= \varepsilon \mid *FF'$$

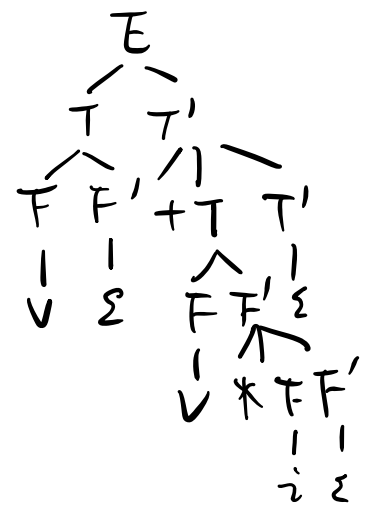
对于句型 $v + v * i$:

(1) 写出它的最左推导过程。(3 分)

(2) 画出它的语法树。(3 分)

(3) 写出它的所有短语、简单短语和句柄 (4 分)

$$\begin{aligned} E &\Rightarrow TT' \Rightarrow FF'T' \Rightarrow vF'T' \Rightarrow vT' \\ &\Rightarrow v+TT' \Rightarrow v+FF'T' \Rightarrow v+vF'T' \\ &\Rightarrow v+v*FF'T' \Rightarrow v+v*iF'T' \Rightarrow v+v*i\varepsilon T' \\ &\Rightarrow v+v*i\varepsilon \end{aligned}$$



短语: $v + v * i$; v ; $+ v * i$; $v * i$; $* i$; i ; ε ;

简单短语: v ; ε ; i

句柄: v

2. (6分) n 维数组某元素绝对地址 $ADDR$ 的计算公式为

$$ADDR = LOC + RC + E \sum_{i=1}^n V(i)P(i), \quad RC = -E \sum_{i=1}^n L(i)P(i)$$

其中 LOC 是分配给数组的连续内存空间的开始地址； RC 是相对不变部分； E 是数组中每个元素占用的内存大小。 $V(i)$ 是指定元素第 i 维下标的值； $P(i)$ 是

$$P(i) = \begin{cases} 1 & \text{当 } i = n \\ \prod_{j=i+1}^n [U(j) - L(j) + 1] & \text{当 } 1 \leq i < n \end{cases}$$

$U(i)$ 是数组第 i 维下标的最大值； $L(i)$ 是数组第 i 维下标的最小值。

假设 $E = 4$ 且程序中声明了数组 $arr(1:5, -1:2, 0:3)$ 。

(1) 计算数组 arr 的 RC 值。(3分)

(2) 若对于数组 arr 有 $LOC = x$ ，求数组元素 $arr(2, 2, 2)$ 的绝对地址。(3分)

$$1) RC = - \sum_{i=1}^n L(i) P(i) E$$

$$P(i) = \begin{cases} 1, & i=3 \\ 4, & i=2 \\ 16, & i=1 \end{cases}$$

$$RC = - \sum_{i=1}^n (16 - 4 + 0) \times 4 \\ = - 48$$

$$2) ADDR = LOC + \sum_{i=1}^n V(i) P(i) E + RC$$

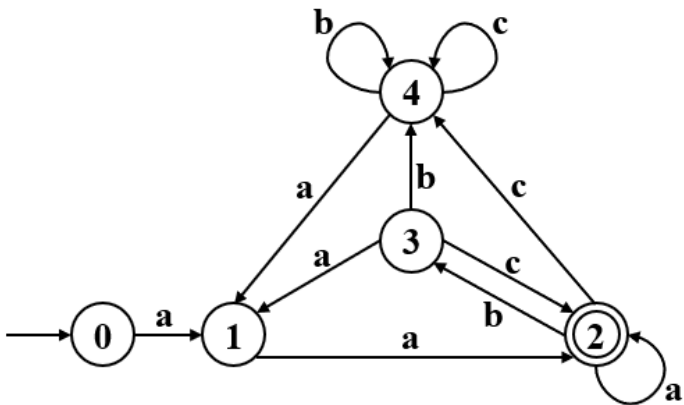
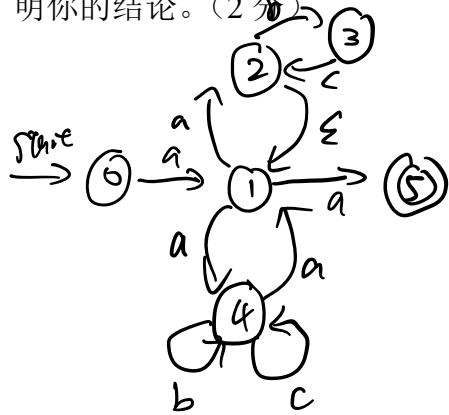
$$= x + 2 \times 21 \times 4 - 48$$

$$= x + 120$$

3. (14 分) 记正则表达式 $a(a(bc)^*|a(b|c)^*a)^*a$ 对应的最小化的 DFA 为 M_1 。

(1) 求 M_1 。(12 分)

(2) 右图所示的 DFA 已经最小化，记为 M_2 。判断 M_1 和 M_2 是否等价，并证明你的结论。(2 分)



(2) $a a b c$ M_2 可接受
 M_1 不可接受
不等价

11) $Z\text{-closure}(\{0,7\}) = \{0,7\}$

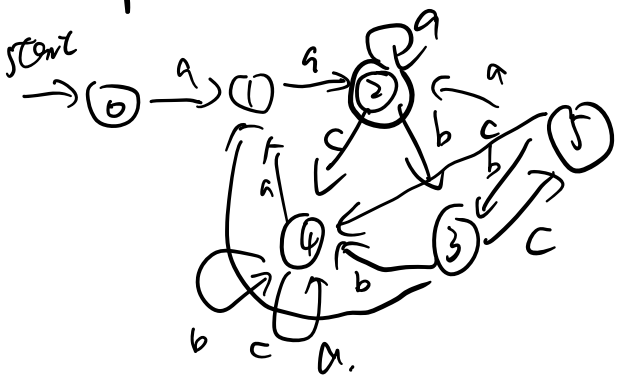
Z	Z_a	Z_b	Z_c
$\{0,7\}$	$\{1,7\}$	\emptyset	\emptyset
$\{1,7\}$	$\{1,2,4,5\}$	\emptyset	\emptyset

$\sqrt{\{1,2,4,5\}}$	$\{1,2,4,5\}$	$\{3,4\}$	$\{4,7\}$
$\{3,4\}$	$\{1,7\}$	$\{4,7\}$	$\{1,2,4,7\}$
$\{4,7\}$	$\{1,7\}$	$\{4,7\}$	$\{4,7\}$
$\{1,2,4,7\}$	$\{1,2,4,5\}$	$\{3,4,7\}$	$\{4,7\}$

转移	a	b	c
0	1	-	-
1	2	-	-
2	2	3	4
3	1	4	5
4	1	4	4
5	2	3	4

	a	b	c
0	1	-	-
1	2	-	-
3	1	4	5
4	1	4	4
5	2	3	4
2	2	3	4

已取此



4. （16 分）给定对于文法G[E]:

$E \rightarrow TE \mid \varepsilon$ $T \rightarrow aFP$ $P \rightarrow bF \mid \varepsilon$ $F \rightarrow i;F \mid ;$

用 # 表示输入串的结尾。

（1）求各候选式的 FIRST 集，以及各非终结符的 FIRST 集和 FOLLOW 集。请直接以集合的形式填写下面的表格，空集合写 Φ 。（6 分）

候选式	FIRST	候选式	FIRST
ε	{ ε }	bF	{ b }
TE	{ a }	$i;F$	{ i }
aFP	{ a }	$;$	{ $;$ }

非终结符	FIRST	FOLLOW
E	{ ε, a }	{ $\#$ }
T	{ a }	{ $\#, a$ }
P	{ b, ε }	{ $\#, a$ }
F	{ $i, ;$ }	{ $\#, a, b$ }

（2）此文法是否适合使用自顶向下的方法分析？说明理由。（2 分）

是

没有左递归、右递归

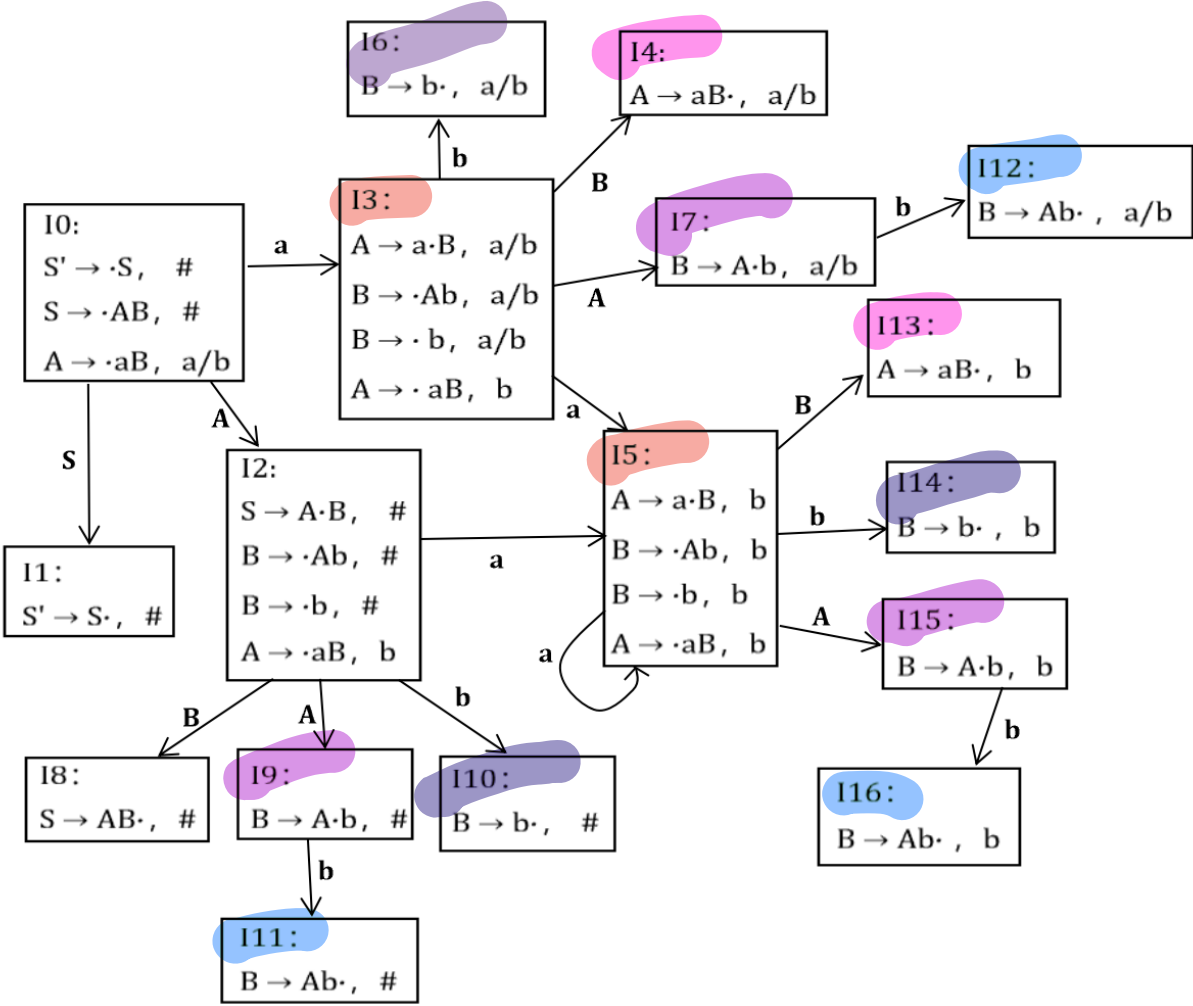
（3）尝试构造该文法的 LL(1) 分析表。请直接填写下面的表格，如果有表项被多次填入，在该处标记 \times 。（8 分）

	a	b	i	;	#
E	$E \rightarrow TE$				$E \rightarrow \varepsilon$
T	$T \rightarrow aFP$				
P	$P \rightarrow \varepsilon$	$P \rightarrow bF$			$P \rightarrow \varepsilon$
F			$F \rightarrow i;F$	$F \rightarrow ;$	

5. (14 分) 文法 $G[S]$ 的拓广文法 $G[S']$ 如下:

- (0) $S' \rightarrow S$
- (1) $S \rightarrow AB$
- (2) $A \rightarrow aB$
- (3) $B \rightarrow Ab$
- (4) $B \rightarrow b$

该文法的 LR(1) 项目集如下图所示。



- (1) 指出对该 LR(1) 项目规范族合并同心集时, 哪些项目集会被合并。(3 分)
 - (2) 构造 LALR(1) 分析表。请直接填写下面的表格。(8 分)
 - (3) 利用 LALR(1) 分析表, 分析输入串 aabbabb。请直接填写下面的表格。(3 分)
- 注意: 合并同心集后的编号取原项目集的最小值, 比如将 I_3 、 I_4 与 I_6 合并, 得到的新项目集为 I_3 , 因此得到的新项目族的编号可能是不连续的。填写分析表和推导过程时, 可能不会用到表格的所有行。

(1) $z_6 z_7 z_4 ; z_{r11} z_{r12} z_{r16}$
 $z_3 z_5 ; z_4 z_5 ; z_7 z_9 z_{15}$

(2) LALR(1) 分析表:

状态	ACTION			GOTO		
	a	b	#	S	A	B
0	S3			1	2	
1			accept			
2	S3	S6			7	8
3	S2	S6			7	4
4	r2	r2				
6	r4	r4	r4			
7		r1				
8			r1			
11	r3	r3	r3			

(3) 分析输入串 aabbabb 的过程：

步骤	状态栈（栈底在左）	已识别符号	待输入串
0	0	#	aabbabb#
1	0a3	#a	abbabb#
2	0a3a3	#aa	bbabb#
3	0a3a3b6	#aab	abb#
4	0a3a3B4	#aaB	abb#
5	0a3A7	#aA	abb#
6	0a3A7b1	#aAb	abb#
7	0a3B4	#aB	abb#
8	0A2	#A	abb#
9	0A2a3	#Aa	bb#
10	0A2a3b6	#Aab	b#
11	0A2a3B4	#AaB	#
12	0A2A7	#AA	b#
13	0A2A7b1	#AAb	#
14	0A2B8	#AB	#
15	0S1	#S	#
16			
17			
18			
19			
20			
21			

