



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，_____的输出是 token 序列，_____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_____型文法，它可以被_____接受。
3. 算符优先分析过程每次规约的是_____。
4. 向输入文法插入动作符号后得到的文法是_____文法，这个文法推导所产生的终结符号串称为_____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_____和_____。

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`，

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

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

三、单选题（每题 3 分，共 21 分）

1. 已知语言 $L = \{a^n b c^m \mid n \geq 0, m > 0\}$ ，下列文法中 D 产生的语言不等于 L 。

- A. $Z ::= AbC \quad A ::= Aa \mid \epsilon \quad C ::= Cc \mid c$
 B. $Z ::= AC \quad A ::= aA \mid b \quad C ::= Cc \mid c$
 C. $Z ::= bC \mid aZ \quad C ::= cC \mid c$
 D. $Z ::= aZc \mid B \quad B ::= b$

2. 对于文法 $G[S]$:

$S ::= \underline{ic}SE \mid a$

$E ::= \underline{e}S \mid \epsilon$

下列符号串中能证明该文法有二义性的是 B。

- A. icicita B. icicicaea C. icicaeaea D. icaeaea

3. 对于算符优先文法 $G[Z]$:

$Z ::= E?E:E$

$E ::= T \mid E + T$

$T ::= F \mid -F$

$F ::= i \mid (Z)$

其中出现的运算符的优先关系，下列正确的有 B。

- ① ?等于: ② ~~+~~等于+ ③ (小于- ④ (小于: ⑤ -大于+ ⑥ ?大于+
 A. ~~①②④~~ B. ①③⑤ C. ~~②③⑥~~ D. ④⑤⑥

4. 对于文法 $G[I]$:

$I ::= \underline{P}BBB \mid D$

P $::= 0b$

$B ::= 0 \mid 1$

D $::= 0 \mid 1 \mid 2 \mid 3 \mid 4 \mid 5 \mid 6 \mid 7$

下列说法正确的是 C。

- A. 它是一个算符优先文法
 B. 它是一个 LL(1)文法
 C. 它是一个 2 型文法
 D. 它的句子集合是无限的

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. $\underline{L(a)}L(a^*) = \underline{L(a^*)}$

B. $\underline{L(a|b)} = \underline{L(a)L(b)}$

C. $\underline{L(b(ab)^*)} = \underline{L((ba)^*b)}$

D. $\underline{L(ab|c)} = \underline{L(a)L(b|c)}$

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

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

$a b c d - e / * +$

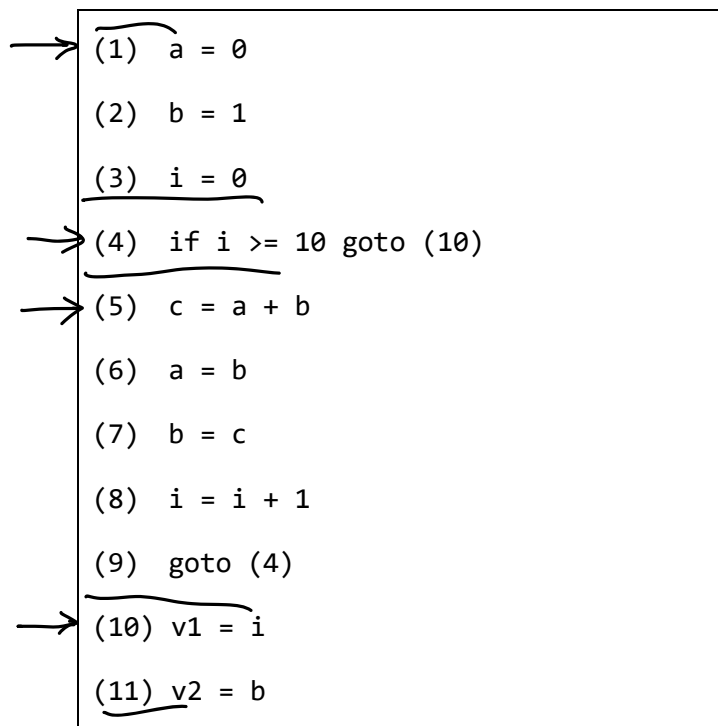
B. $b c d - e / * a +$

$+ * \text{ (crossed out) } /$

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。



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)

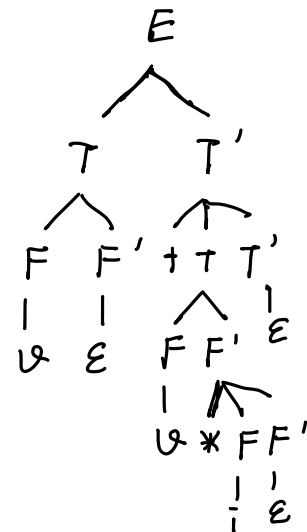
四、综合题（共 60 分）

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）

1) $E \Rightarrow TT'$
 $\Rightarrow FF'T'$
 $\Rightarrow vFT'$
 $\Rightarrow vT'$
 $\Rightarrow v + TT'$
 $\Rightarrow v + FF'T'$
 $\Rightarrow v + vF'T'$
 $\Rightarrow v + v * FF'T'$
 $\Rightarrow v + v * iF'T'$
 $\Rightarrow v + v * iT'$
 $\Rightarrow v + v * i$



13) 短语: $v, \varepsilon, v\varepsilon, i, *i\varepsilon$
 $v * i\varepsilon, +v * i\varepsilon\varepsilon,$
 $v\varepsilon + v * i\varepsilon\varepsilon$

短语 $\varepsilon =$

$v, \varepsilon, i, *i$

$v * i, +v * i, v + v * i$

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)

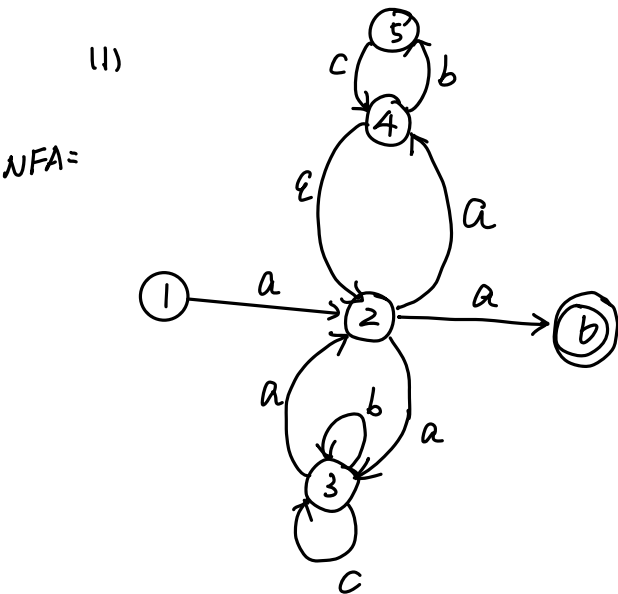
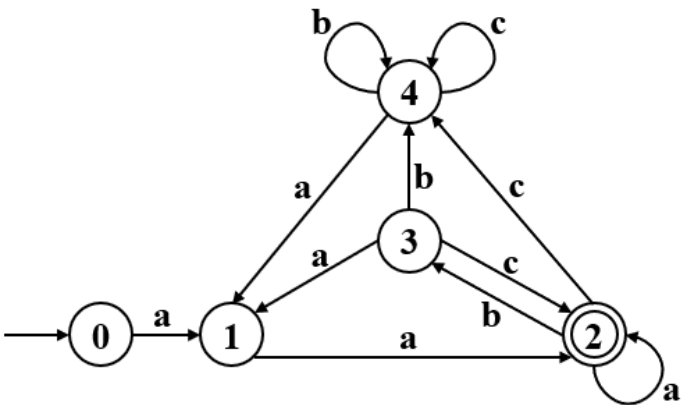
$$\begin{aligned} (1) \quad & p(1) = 4 \times 4 = 16 \\ & p(2) = 4 \\ & p(3) = 1 \\ & RC = -4 \times (1 \times 16 + -1 \times 4 + 0 \times 1) \\ & = -4 \times (16 - 4) \\ & = -48 \end{aligned}$$

$$\begin{aligned} (2) \quad & ADDR = x - 48 + 4 \times (2 \times 16 + 2 \times 4 + 2 \times 1) \\ & = x - 48 + 4 \times (32 + 8 + 2) \\ & = x - 48 + 4 \times (42) \\ & = x - 48 + 168 \\ & = x + 120 \end{aligned}$$

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 M1。

(1) 求 M1。(12 分)

(2) 右图所示的 DFA 已经最小化，记为 M2。判断 M1 和 M2 是否等价，并证明你的结论。(2 分)



I	I_a	I_b	I_c	
0	1	\emptyset	\emptyset	I
1	2	\emptyset	\emptyset	II
<u>2</u>	2	3	4	III
3	1	4	5	IV
4	1	4	4	V
5	2	3	4	

	I	I_a	I_b	I_c
0	{1}	{2}	\varnothing	\varnothing
1	{2}	{2, 3, 4, 6}	\varnothing	\varnothing
<u>2</u>	{2, 3, 4, 6}	{2, 3, 4, 6}	{5, 3}	{3}
3	{3, 5}	{2}	{3}	{3, 4, 2}
4	{3}	{2}	{3}	{3}
5	{3, 4, 2}	{2, 3, 4, 6}	{3, 5}	{3}

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	$\{\varepsilon\}$	bF	$\{b\}$
TE	$\{a\}$	$i;F$	$\{i\}$
aFP	$\{a\}$	$;$	$\{;\}$

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

（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

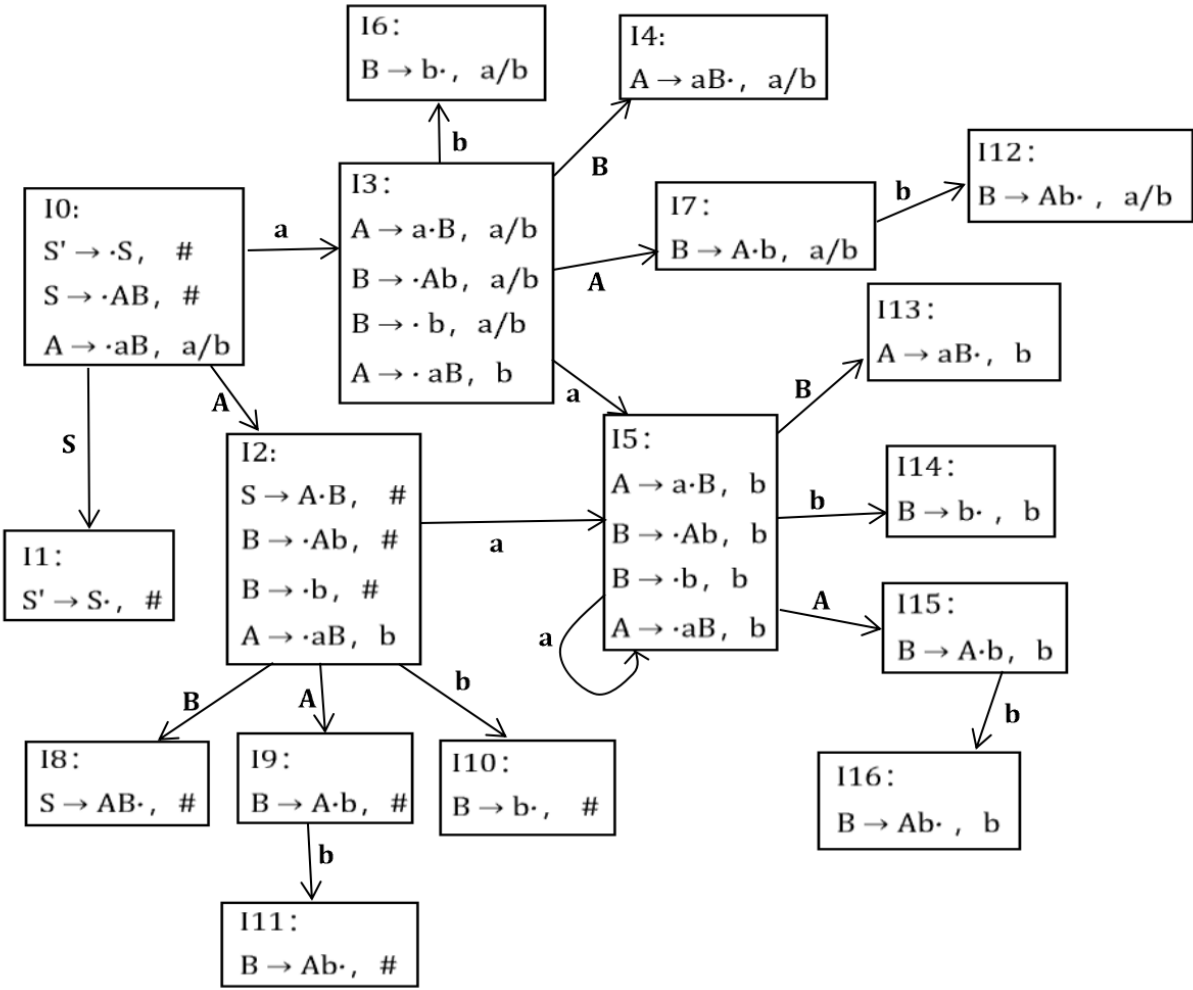
（3）尝试构造该文法的 LL(1) 分析表。请直接填写下面的表格，如果有表项被多次填入，在该处标记 \times 。（8 分）

	a	b	i	;	#
E					
T					
P					
F					

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)

(2) LALR(1) 分析表:

状态	ACTION			GOTO		
	a	b	#	S	A	B

(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			
2			
3			
4			
5			
6			
7			
8			
9			
10			
11			
12			
13			
14			
15			
16			
17			
18			
19			
20			
21			

