

## 2011 级《编译原理》期末考试试题 (A 卷答案)

考试时间: 2013 年 1 月 10 日

班级 \_\_\_\_\_ 学号 \_\_\_\_\_ 姓名 \_\_\_\_\_

- ✧ 请将答案写在答题纸上, 写明题号, 不必抄题, 字迹工整、清晰;
- ✧ 请在答题纸和试题纸上都写上你的班级, 学号和姓名, 交卷时请将试题纸、答题纸和草纸一并交上来。

### 一、填空题 (4 小题, 每题 5 分, 共 20 分)

1. 设有文法如下, 则句型 (adSdA) 的句柄为 a ; 简单短语有: a, SdA ;

除以上简单短语外, 剩余的短语有: (adSdA), adSdA 。

$S \rightarrow a \mid b \mid (A)$

$A \rightarrow S d A \mid S$

2. 请依据所给文法, 将下面的递归下降程序补充完整。

$S \rightarrow a S \mid X b \mid \varepsilon$

$X \rightarrow b a$

$S ()$

```
{ switch (ch) of
    'a' : match(a); S(); break;
    'b' : X(); match(b); break;
    '#' : skip 或 {}; break;
    default : error();
}
```

3. 在 C 语言程序中, 局部变量 int x 存放的位置是 栈区; 语句  $p = \text{malloc}(\text{sizeof}(\text{int}) * 10)$  申请得到的空间位于 堆区; 全局变量 int sum 存放的位置是 静态区; 静态局部变量 static int sx 的存放位置是 静态区, 该程序生成四元式中间代码时, 产生的临时变量存放的位置是 栈区。

4. 设语义分析中当前层数为 L, 偏移量为 off, 试写出以下程序段中各标记点处的层数和偏移情况。注意: 约定 char, int, float 型分别分配 1, 1, 2 个存储单元。

(L, off)

```
typedef struct { int number;
                char name[10]; } example;
```

① (L, off) example p;

float x;

void main(② (L+1, 0) )

{ int x;

{ ③ (L+1, 1) float x; ..... }

```

    { ④(L+1, 1)int x; .....}
float  y⑤(L+1,5);
.....
}

```

二、计算题（4 小题，每题 5 分，共 20 分）

1. 等价的 DFA 如下

	0	1
{0, 1, 2}	{1, 2}	{1, 2, 3}
{1, 2}	{1, 2}	{1, 2, 3}
{1, 2, 3}	{1, 2, 4}	{1, 2, 3, 4}
{1, 2, 4}*	{1, 2}	{1, 2, 3}
{1, 2, 3, 4}*	{1, 2, 4}	{1, 2, 3, 4}

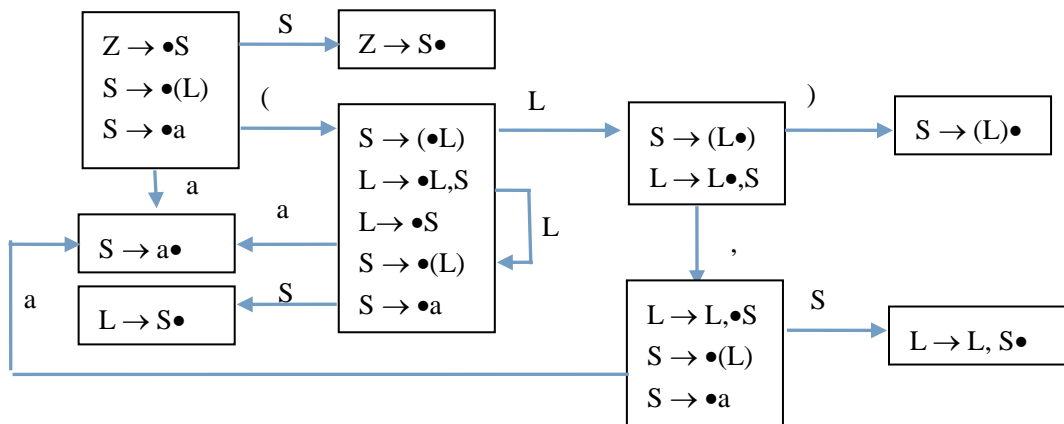
2. 最小 DFA 如下。

$\{\{0,1,2\}, \{1,2\}\}$  、  $\{1,2,3\}$ 、  $\{1,2,4\}$ 、  $\{1,2,3,4\}$

3. 为下面文法构造 LR(0) 自动机。

$S \rightarrow ( L ) \mid a$

$L \rightarrow L , S \mid S$



4. 为下面文法构造简单优先矩阵。

$$S \rightarrow ( L ) \mid a$$

$$L \rightarrow L , S \mid S$$

	#	S	L	(	)	a	,
#		<		<		<	
S	>				>		>
L					=		=
(		<	=,<	<		<	
)	>				>		>
a	a				>		>
,		=		<		<	

三、简答题（4 小题，每题 5 分，共 20 分）

1.  $\Sigma = \{a, b\}$ ，写出识别语言“每个 a 后面至少有一个 b 的所有串”的正则表达式。

$$(ab \mid b)^*$$

2. 写出识别语言  $L(G) = \{a^n cb^n \mid n \geq 1\}$  的上下文无关文法。

$$S \rightarrow aSb \mid c$$

3. 下面文法是否是二义性文法？

$$S \rightarrow a S b \mid S b \mid b$$

是二义性文法。句子 abbb 存在两棵语法树。

4. 过程活动记录中一般包含哪些内容？

动态链指针、返回地址、返回值、层数、活动记录空间大小、寄存器状态、形参变量、局部变量、临时变量。

四、问答题（4 小题，每题 10 分，共 40 分）

1. 判断下面文法是否是 LL(1) 文法？说明理由。若不是，将其变换成 LL(1) 文法，并给出 LL(1) 分析表。

$$S \rightarrow (S)S \mid (L)S \mid \epsilon$$

$$L \rightarrow i \mid L, i$$

因产生式存在左公共前缀和左递归，因此不是 LL(1) 文法。

变换后文法如下：

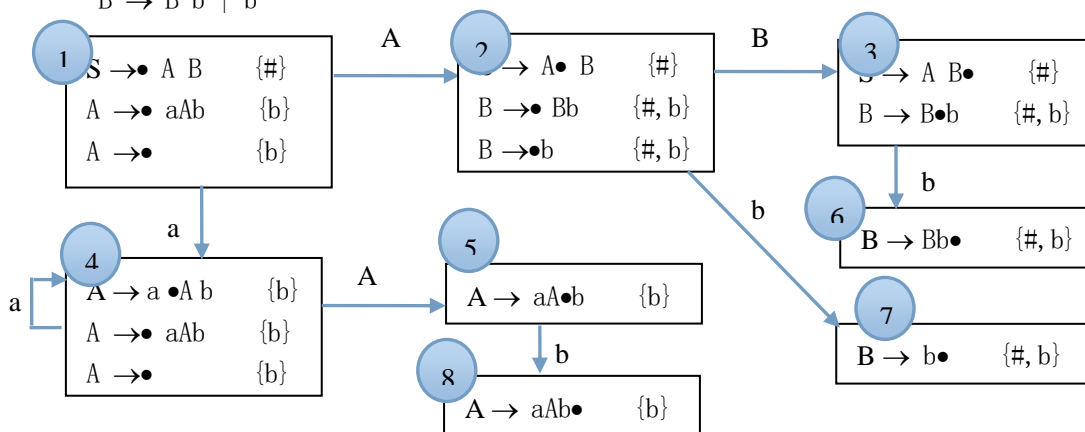
$S \rightarrow ( S' \quad [1] \mid \varepsilon \quad [2]$   
 $S' \rightarrow S ) S \quad [3] \mid L S \quad [4]$   
 $L \rightarrow i L' \quad [5]$   
 $L' \rightarrow , i L' \quad [6] \mid \varepsilon \quad [7]$

LL(1)分析表如下:

	(	)	,	i	#
S	1	2			2
S'	3	3		4	
L	5				
L'		7	6		

2. 画出下面文法的 LR(1) 自动机, 并判断该文法是否是 LR (1) 文法? 说明理由。若是, 则构造 LR (1) 分析表。

$S \rightarrow A B$   
 $A \rightarrow a A b \mid \varepsilon$   
 $B \rightarrow B b \mid b$



不存在 S-R, R-R 冲突, 是 LR(1) 文法。

LR (1) 分析表如下: (因状态编号不同, 此答案不唯一)

	a	b	#	A	B
1	S4	R3		2	
2		S7			3
3		S6	Accept		
4	S4	R3		5	
5		S8			
6		R4	R4		
7		R5	R5		
8		R2			

3. 驻留法实现全局顺序符号表，给出扫描下述程序后的符号表内容。（每个函数的局部数据区的起始偏移为 initOff，每个 int 类型数据占 1 个存储单元，每类名字应包括关键属性。）

```
void main()
{
    int a=1;
    int b=1;
    {
        int a=2;
        {
            int b=2;
            printf("a = %d, b = %d\n",a, b);
        }
        {
            int a = 3;
            printf("a = %d, b = %d\n",a, b);
        }
    }
    printf("a = %d, b = %d\n", a, b);
}
```

名字	局部单位编号	层数	偏移	
main	0	0		
a	1	1	0	
b	1	1	1	
a	2	1	2	
b	3	1	3	
#				
a	3	1	3	
#				
#				
#				

4. 已知如下程序段:

```
a = 1;
while (a <=10) do
{   if (a != b) A[a] = A[a] + 2;
    a = a + 1;
}
```

(1) 写出上述程序段的四元式中间代码;

```
(Assign, 1, -, a)
(While, -, -, -)
(LE, a, 10, t1)
(Do, -, -, -)
(NE, a, b, t2)
(THEN, t2, -, -)
(AADD, A, a, t3)
(AADD, A, a, t4)
(ADDI, t4, 2, t5)
(Assign, t5, 1, t3)
(ENDIF, -, -, -)
(ADDI, a, 1, t6)
(ASSIGN, t6, 1, t7)
(ENDWHILE, -, -, -)
```

(2) 将(1)中生成的中间代码划分成基本块。

基本块划分如下:

```
B1: (Assign, 1, -, a)
B2: (While, -, -, -)
    (LE, a, 10, t1)
    (Do, -, -, -)
B3: (NE, a, b, t2)
    (THEN, t2, -, -)
B4: (AADD, A, a, t3)
    (AADD, A, a, t4)
    (ADDI, t4, 2, t5)
    (Assign, t5, 1, t3)
B5: (ENDIF, -, -, -)
    (ADDI, a, 1, t6)
(ASSIGN, t6, 1, t7)
(ENDWHILE, -, -, -)
```