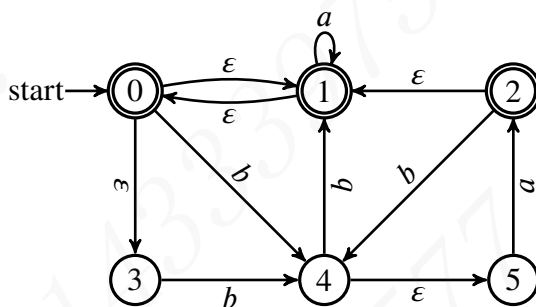


武汉大学计算机学院
2019 - 2020 学年第一学期 2017 级
《编译原理》期末考试试卷 (A)

学号: _____ 姓名: _____ 专业: _____ 成绩: _____

(注: ①考试时间为 120 分钟; ②所有的解答必须写在答题纸上, 并注明题号。)

一、 设 NFA N 的状态转换图如下所示: (25 分, 每小题 5 分)



- (1) 试写出 NFA N 接受字符串 “abaabb” 的一个过程;
- (2) 设用子集构造法求出的与 NFA N 等价的 DFA M 有 3 个状态 A , B 和 C , 其中 $A = \epsilon\text{-closure}(\{0\})$, $\text{Dtrans}(A, b) = B$. 试求与状态 A , B 和 C 所对应的 NFA N 的状态集, 并画出 DFA M 的状态转换图;
- (3) 求 DFA M 的最小状态自动机;
- (4) 试用自然语言描述 NFA N 所接受的语言;
- (5) 求正规表达式 r , 使得 $L(r) = L(N)$.

二、 设函数调用表达式文法 $G(E)$ 定义如下:

$$\begin{aligned} E &\rightarrow E(A) \mid (E) \mid \text{id} \\ A &\rightarrow A, A \mid E \end{aligned}$$

其中: ‘id’, ‘,’ , ‘(’ 和 ‘)’ 为终结符, E 是文法开始符号. (25 分, 每小题 5 分)

- (1) 试写出语句 “id(id, id)” 的一个最左推导;
- (2) 试消除文法 $G(E)$ 中的左递归;
- (3) 试对消除左递归后的文法所有非终结符求 First 集和 Follow 集;
- (4) 试对消除左递归后的文法构造 LL(1) 分析表, 从而说明消除左递归后的文法不是 LL(1) 文法;
- (5) 试利用你的分析表写出语句 “id(id)” 的一个正确的分析过程.

三、 设文法 $G(E)$ 如题二所示：

(10 分, 5+5)

- (1) 试对语句 “id(id, id, id)” 画出两棵不同的语法树，从而说明该文法为二义文法；
- (2) 试设计一个与文法 $G(E)$ 等价的无二义的文法，使得参数列表 (A, A) 为左结合。

四、 设题二文法 $G(E)$ 的拓广文法 $G(E')$ 如下所示：

(20 分, 5+5+5+5)

$$E' \rightarrow E \quad (0)$$

$$E \rightarrow E(A) \quad (1)$$

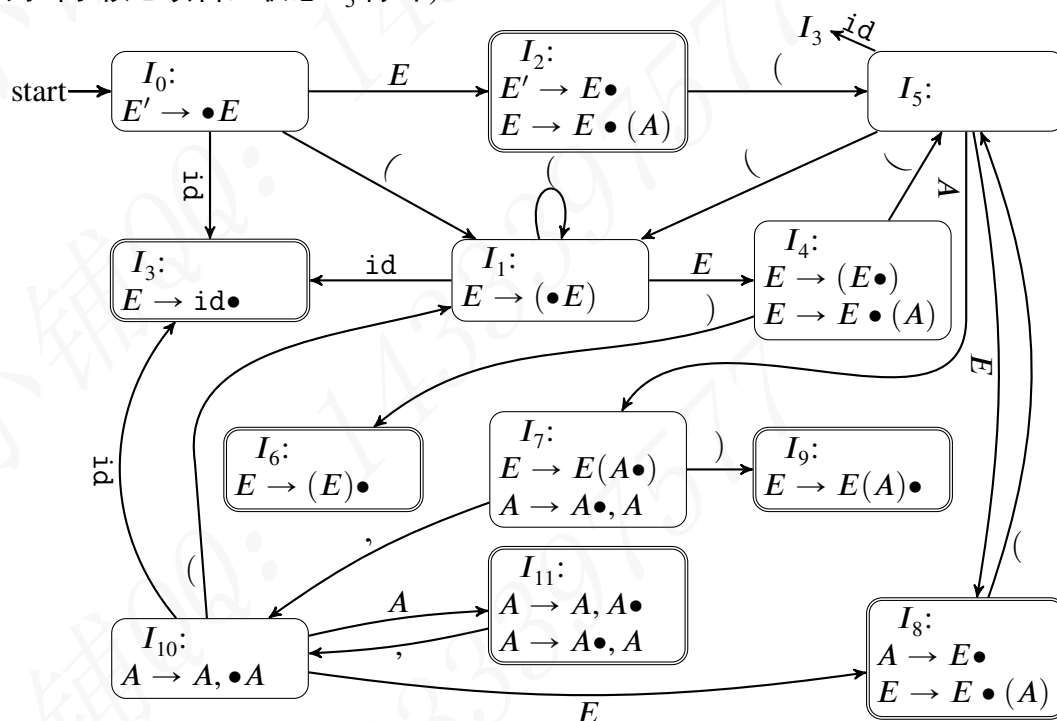
$$| (E) \quad (2)$$

$$| id \quad (3)$$

$$A \rightarrow A, A \quad (4)$$

$$| E \quad (5)$$

文法 $G(E')$ 的识别活前缀 LR(0) 项目自动机 M 如下图所示 (注意每个状态仅列出了核心项目，状态 I_5 除外)：



- (1) 试求状态 I_5 所对应的 LR(0) 项目集；
- (2) 试求终结符号串集合 $\{w \mid E(A, w) \in L(M) \wedge w \in \{id, ,, (,)\}^*\}$ (即活前缀 “ $E(A,$ ” 加上后缀 w 能形成句柄) 所对应的正则表达式；
- (3) 试构造该文法的 SLR 分析表，并对分析表中的移进/归约冲突选择正确的移进或归约动作，使得文法 $G(E)$ 的所有语句能被正确地分析且运算的结合次序与题三所规定的一致；
- (4) 试利用你的分析表写出语句 “id(id)” 的分析过程。

- 五、现需对题二文法 $G(E)$ 所生成的函数调用表达式柯里化 (currying), 即所有的函数只允许有一个参数, 且函数调用 $f(x)$ 记为 $f\ x$. 若 f 是二元函数, 柯里化把 f 看成高阶函数, 即 $f\ x$ 的返回值是函数, 这样 $f(x,y)$ 的柯里化为 $(f\ x)\ y$, 简记为 $f\ x\ y$ (柯里化的函数调用为左结合). 函数调用表达式的最简柯里化 (括号最少) 用例如下: (10 分, 5+5)

序号	表达式	最简柯里化
1	$f(a,b,c,d)$	$f\ a\ b\ c\ d$
2	$(f(a,b))(c,d)$	$f\ a\ b\ c\ d$
3	$((f(a))(b))(c,d)$	$f\ a\ b\ c\ d$
4	$(f(a(b)))(c,d)$	$f\ (a\ b)\ c\ d$
5	$f(a(c,d),e(g,h))$	$f\ (a\ c\ d)\ (e\ g\ h)$

为此设计以下综合属性: $E.kind$ 和 $A.kind$, 其取值为选项 OPEN 或 CLOSE. 当 E 所表示的语法成分是 id 或 $E(A)$ 时, $E.kind$ 的取值分别是 CLOSE 或 OPEN. 当 A 所表示的语法成分是 A, A 或 E 时, $A.kind$ 的取值分别是 CLOSE 或 $E.kind$; $E.curry$ 和 $A.curry$ 取值为其表示的语法成分最简柯里化表达式 (字符串); 终结符 id 有属性 $id.lexeme$, 其取值为 id 所对应的字符串.

- (1) 试设计最简柯里化 SDD;
- (2) 试求 “ $(a(b(c(d,e),f),g))(h(i,j))$ ” 的最简柯里化.

- 六、设有如下 Pascal 程序片段: (5 分)

```
while a > b and not (c > d) do
begin
  x := x + 1;
  if e > f and g > h or not (i > k) then continue;
  else x := y + 2;
end;
```

其对应的三地址码如下所示

```
L1: [ ] (a > b) goto L__ | L0: [ ] (i > k) goto L__
      [ ] (c > d) goto L__ |      t0 := y + 2
      x := x + 1           |      x := t0
      [ ] (e > f) goto L__ |      goto L__
      [ ] (g > h) goto L__ | L2:
```

试为其中空白 “__” 填上正确的标号编号, 并为空白 “[]” 填上 if 或 ifnot.

第七题见下页!

七、设有如下 C 语言程序：

(5 分)

```
1 #include <stdio.h>
2 #include <string.h>
3 #include <stdlib.h>
4
5 char **foo(char **src, int n)
6 {
7     int i = 0;
8     char **dest;
9     dest = (char **) malloc(sizeof(char *) * n);
10    while(i < n) {
11        dest[i] = strdup(src[i]);
12        i++;
13    }
14    return dest;
15 }
16
17 int main()
18 {
19     char s[][20] = {"Hello", "Class 2017"};
20     char **dest;
21     dest = foo((char **)s, 2);
22     for(int i = 0; i < 2; i++)
23         printf("%s\n", dest[i]);
24     return 0;
25 }
```

该程序希望拷贝一个字符串数组，并输出拷贝后的数组。在 Intel x86/Linux 下用 gcc 编译没有任何警告，但运行编译后的程序报“Segmentation fault”。但若把程序的第 19 行修改为：

```
19 char *s[] = {"Hello", "Class 2017"};
```

则程序工作正常，即在输出以下字符串后正常退出：

```
Hello
Class 2017
```

试问上述程序产生运行错误的原因。