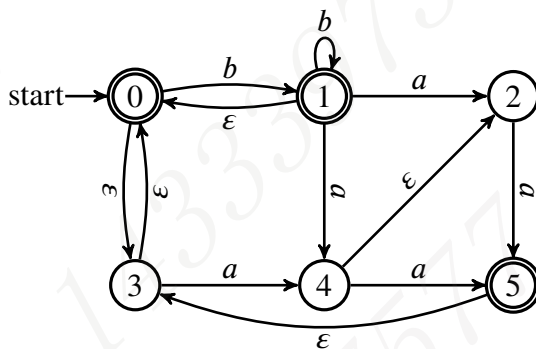


武汉大学计算机学院
2020 - 2021 学年第二学期 2018 级
《编译原理》期末考试试卷 (A)

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 “baabbbaa” 的过程;
- (2) 设用子集构造法求出的与 NFA N 等价的 DFA M 有 4 个状态 A, B, C 和 D , 其中 $A = \epsilon\text{-closure}(\{0\})$, 状态转换函数 $\text{Dtrans}(A, a) = B, \text{Dtrans}(A, b) = C$ 试求与状态 A, B, C 和 D 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) 求正则表达式 r , 使得 $L(r) = L(N)$.

二、设嵌套序偶列表 $G(S)$ 定义如下:

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

其中: ‘id’, ‘(’, ‘)’, 和 ‘,’ 为终结符, S 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 $G(S)$ 中的左递归和左公因子;
- (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 $G(S)$ 如题二所示：

(10 分, 5+5)

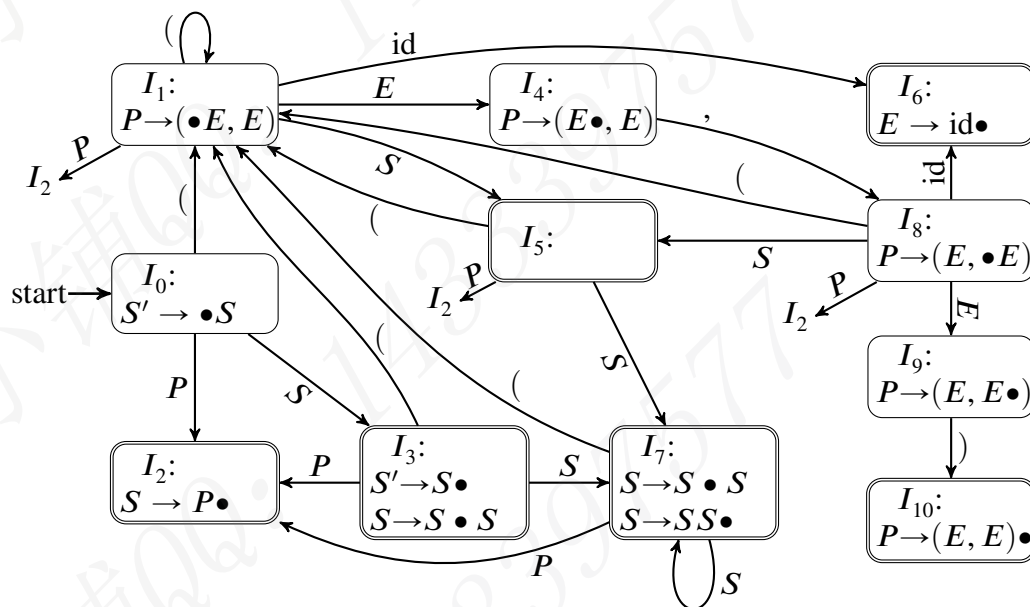
- (1) 试对语句 “(id,id)(id,id)(id,id)” 画出两棵不同的语法树，从而说明该文法为二义文法；
- (2) 试设计一个与文法 $G(S)$ 等价的无二义的文法，使得序偶链接运算 (SS) 为左结合。

四、设题二文法 $G(S)$ 的拓广文法 $G(S')$ 如下所示：

(20 分, 5+5+5+5)

$$\begin{aligned}
 S' &\rightarrow S & (0) \\
 S &\rightarrow SS & (1) \\
 &| P & (2) \\
 P &\rightarrow (E, E) & (3) \\
 E &\rightarrow S & (4) \\
 &| id & (5)
 \end{aligned}$$

文法 $G(S')$ 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) 试问 “(E,E,” 是活前缀吗，为什么？
- (3) 试构造该文法的 SLR 分析表，并对分析表中的移进/归约和归约/归约冲突选择正确的移进或归约动作，使得文法 $G(S)$ 的所有语句能被正确地分析且运算的优先级与结合次序与题三所规定的一致；
- (4) 试利用你的分析表写出 “(id,id,id)” 的分析过程，从而说明它不是文法 $G(S)$ 的语句。

- 五、若把题二文法 $G(S)$ 的 id 看成有向图的顶点，所生成的序偶看成有向图的边，如： $(a,b)(c,d)$ 表示有向图有四个顶点 a 、 b 、 c 和 d 及两条边 (a,b) 和 (c,d) ； $((a,b)(c,d),(e,f))$ 可把 $(a,b)(c,d)$ 和 (e,f) 看成两个子图 G_1 和 G_2 ， G_1 第一个顶点 a 到 G_2 的第一个顶点 e 有有向边。现需将文法 $G(S)$ 所生成的语句转换成绘图软件 Grapviz 输入文件格式。如

序号	嵌套序偶列表	Graphviz 输入
1	$(a,b)(c,d)$	$a \rightarrow b \quad c \rightarrow d$
2	$((a,b)(c,d),e)$	$\text{subgraph } \{a \rightarrow b \quad c \rightarrow d\} \quad a \rightarrow e$
3	$((a,b)(c,d),(e,f))$	$\text{subgraph } \{a \rightarrow b \quad c \rightarrow d\}$ $\text{subgraph } \{e \rightarrow f\} \quad a \rightarrow e$
4	$((((a,b)(c,d),(e,f)),g,h))$	$\text{subgraph } \{$ $\text{subgraph } \{a \rightarrow b \quad c \rightarrow d\}$ $\text{subgraph } \{e \rightarrow f\} \quad a \rightarrow e$ $\}$ $\text{subgraph } \{g \rightarrow h\} \quad a \rightarrow g$

为此设计为非终结符 S, E 及 P 设计综合属性 $graph$ 和 $node$ ， $graph$ 的值为该非终结符所表示的语法成分所对应的 Graphviz 语句； $node$ 的值为其表示的语法成分第一个顶点。如 S 的表示的语法成分是 $(a,b)(a,c)$ 时， $S.graph = "a \rightarrow b \quad a \rightarrow c"$ ， $S.node = "a"$ 。 E 同时有取值为布尔量的属性 is_id ，当 E 是由 id 生成时，其取值为 $true$ ，否则取值为 $false$ 。属性 $id.lexeme$ 取值为终结符 id 所对应的字符串：

- (1) 试写出文法 $G(S)$ 翻译成 Graphviz 绘图语言的语法制导定义；
- (2) 试语句 “ $((((a,b),(c,d)(e,f))(g,h),(i,m)))$ ” 转换后的 Graphviz 语句。

- 六、设有如下 Pascal 程序片段： (5 分)

```
while not(a < b and (c < d or not (e < f))) do
begin
  x := x + 1;
  if not (x = 0) and not (x = 2) then continue;
  else x := x + 2;
end;
```

其对应的三地址码如下所示

```
L1: [   ] (a < b) goto L__ | [   ] (x = 2) goto L__
    [   ] (c < d) goto L__ | L0: t1 := x + 2
    [   ] (e < f) goto L__ |    x := t1
L2: t0 := x + 1           |    goto L1
    x := t0               | L3:
    [   ] (x = 0) goto L__ |
```

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

七、 设有如下 C 语言程序:

(5 分)

```
1 #include <stdio.h>
2 #include <stdlib.h>
3
4 void alloc(char *s)
5 {
6     s = (char *) malloc(20);
7 }
8
9 int main(void)
10 {
11     char *s;
12     alloc(s);
13     sprintf(s, "%s", "be away from covid!");
14     printf("%s\n", s);
15     return 0;
16 }
```

该程序在 Intel x86/Linux 下用 gcc 编译没有任何警告, 但运行编译后的程序报错如下:

Segmentation fault (core dumped)

试分析产生运行错误的原因.