

总复习

赵银亮

2024

¹论域{^{1.1}中心概念{符号, 字母表, 符号串, 语言, 问题}, ^{1.2}图灵机模型{状态, 当前状态, 输入串, 输入符号, 状态转换规则, 初始状态, 接受状态, 程序, TM, UTM}, ^{1.3}Chomsky体系{形式文法, 形式语言}, ^{1.4}编译过程{词法分析, 语法分析, 语义分析, 代码优化, 代码生成, 编译遍}, ^{1.5}编译程序{源程序, 源语言, 目标程序, 目标语言, 中间语言, 前端, 后端}, ^{1.6}主文法, ^{1.7}主符号系统{Lisp表, 命名惯例}},

²正则语言{²¹语言识别器{DFA^[1.2], NFA^[1.2], RE^[1.1]{语言表达式, 原子语言, 语言上运算, 语言模式化}}, ²²判定性质{扩展转移函数^[211-2], 模式匹配^[213]}, ²³等价性质^[21]{NFA~DFA, DFA~DFA, RE~NFA}, ²⁴封闭性},

³CFL{³¹识别器{CFG^[1.3]{变元, 终结符, 开始符号, 产生式规则, 候选式}, PDA^[1.2]{栈符号, 栈开始符号, 转移函数}}, ³²判定性质{推导^[311]{句型, 句子, 直接推导, 最左推导, 最右推导}, 归约^[311]{直接归约, 可归约串, 确定性归约}, 语法树^[311]{产物, 短语, 直接短语, 句柄}, 移动^[312]{瞬时描述, N移动, P移动}}, ³³等价性质^[31]{CFG化简^[311, 321]{去除无用符号, 消除 ϵ -产生式, 消除单位产生式}, PDA_N~PDA_F^[312, 324], ^UPDA~CFG}, ³⁴歧义性{来源{优先级, 结合性, 悬空else}, 消除歧义性, 固有歧义性}},

⁴词法分析{词法记号{一符一种, 全体一种}, σ -DFA^[121-2]{ \mathcal{L} , ψ , 事实优先级, 前缀最大化}, 扫描框架{预处理{有效串, 分隔, 判定框架{ σ -DFA 状态转换表, 识别流程}}, 附加处理{缓冲区控制, 超前搜索}, 后处理{关键字识别, 值转换}}, 扫描器, 扫描器生成器{描述文件}},

⁵LL(1)分析{⁵¹LL(1)文法^[311]{LL(1)文法性质, 首符集, FOLLOW集, 消除左递归, 消除回溯}, ⁵²LL(1)分析器{LL(1)框架, 递归下降分析程序, 预测分析表},

⁶SLR(1)分析{⁶¹最左归约{符号栈, 移进, 归约, sr冲突, ss冲突, 非确定性归约}, ⁶²规范归约{句柄, 活前缀, 有效项目, LR(0)规范簇^[32]}, ⁶³规范归约模拟器{itemNFA, itemDFA^[311,324], SLR(1)文法^[624], 冲突消解}, ⁶⁴SLR(1)分析器{LR(0)分析表^[632], SLR(1)分析表^[632], SLR(1)框架^[632]}, ^ULR(1)分析, ^ULALR分析, ^U生成器},

⁷语法制导语义分析{⁷¹分析框架, ⁷²属性文法^[312]{属性, 属性名, 属性方程, 综合属性, 继承属性, 属性求值, 桩变量, S-属性文法, L-属性文法}, ⁷³LL(1)制导分析框架^[72], ⁷⁴SLR(1)制导分析框架^[72], ⁷⁵函数符号表{表头{声明宿主, 宽度, 参数个数, 参数表, 返回类型, 层数, 代码}, 登记项{名, 类型, 类型特有域}, 操作{newtab(), bind(), lookup(), update[]}}, ⁷⁶声明的语义分析^[74,75]{局部名声明, 形参声明}, ^{U77}类型检查, ⁷⁸中间语言{三地址指令, 四元式, ^U抽象语法树, ^U后缀式}, ⁷⁹中间代码生成^[74]{名字引用, 表达式, 语句}},

⁸运行时环境{⁸¹内存映像{活动树, 栈帧{参数区, 链接区, 局部区}, 栈快照, 内存快照}, ⁸²参数传递{形参{简单变量, 数组原型, 函数原型, ^U数组, ^U函数签名}, 机制{传值, ^U传地址, ^U得结果, ^U传名}}, ^{U83}函数可执行代码^[812]{指令系统, 指令模板, 调用序列, 返回序列, 序言, 尾声, 寻址序列}}, ⁸⁴非局部名寻址{声明宿主, 引用宿主, 作用域, 访问链, 函数闭包},



关于主文法

- ▷ $P \rightarrow \check{D} \check{S}$
- ▷ $\check{D} \rightarrow \varepsilon \mid \check{D} D;$
- ▷ $D \rightarrow T d \mid T d[\check{I}] \mid T d(\check{A})\{\check{D} \check{S}\}$
- ▷ $T \rightarrow \text{int} \mid \text{float} \mid \text{void}$
- ▷ $\check{I} \rightarrow i \mid \check{I}, i$
- ▷ $\check{A} \rightarrow \varepsilon \mid \check{A} A;$
- ▷ $A \rightarrow T d \mid T d[] \mid \cancel{T d[\check{I}]} \mid \cancel{T d(\check{T})} \mid T d()$
- ~~▷ $\check{T} \rightarrow \varepsilon \mid \check{T} T;$~~
- ▷ 对形参更改：有数组原型和函数原型，无数组和函数签名。



关于主文法 (续)

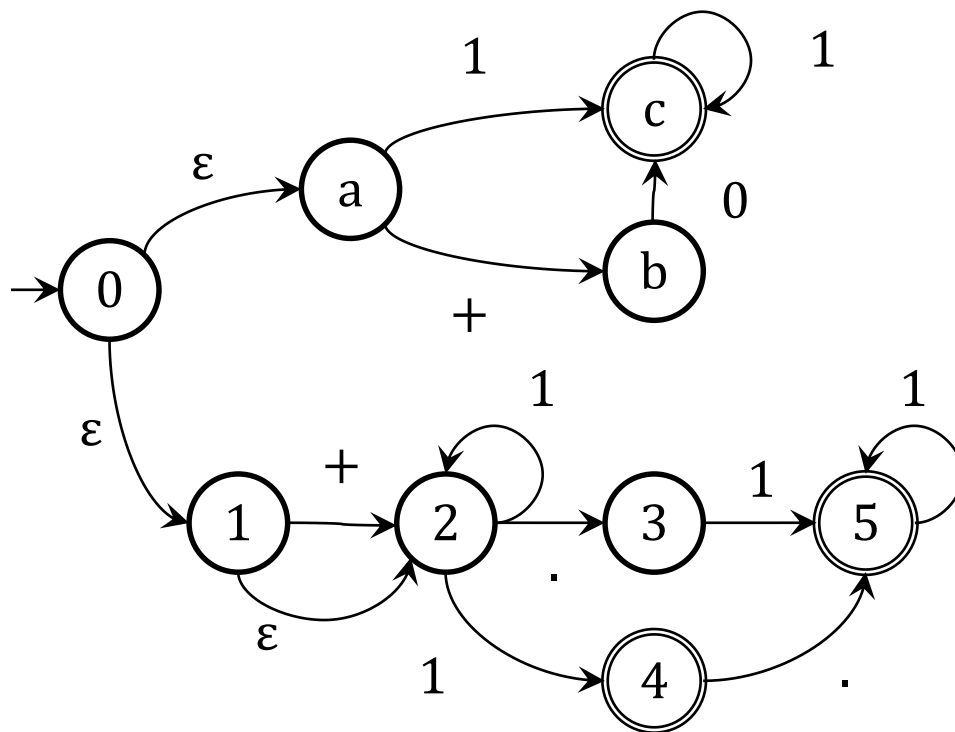
- ▶ $\check{S} \rightarrow S \mid \check{S}; S$
- ▶ $S \rightarrow d = E \mid d[\check{E}] = E \mid \text{if } (B) S \mid \text{if } (B) S \text{ else } S \mid \text{while } (B) S \mid \text{return } E \mid \{\check{S}\} \mid d(\check{R})$
- ▶ $E \rightarrow i \mid f \mid d \mid d[\check{E}] \mid E \circ E \mid u E \mid (E) \mid d(\check{R})$
- ▶ $\check{E} \rightarrow E \mid \check{E}, E$
- ▶ $B \rightarrow B \wedge B \mid B \vee B \mid ! B \mid (B) \mid E \text{ r } E \mid E$
- ▶ $\check{R} \rightarrow \varepsilon \mid \check{R} R,$
- ▶ $R \rightarrow E \mid d[] \mid d()$
- ▶ 对算术运算符即有全集一种即 \circ 也有一符一种即 $+ - * /$, 对关系运算符类似。
- ▶ 也有 $\&\&$ 和 $||$



NFA转DFA

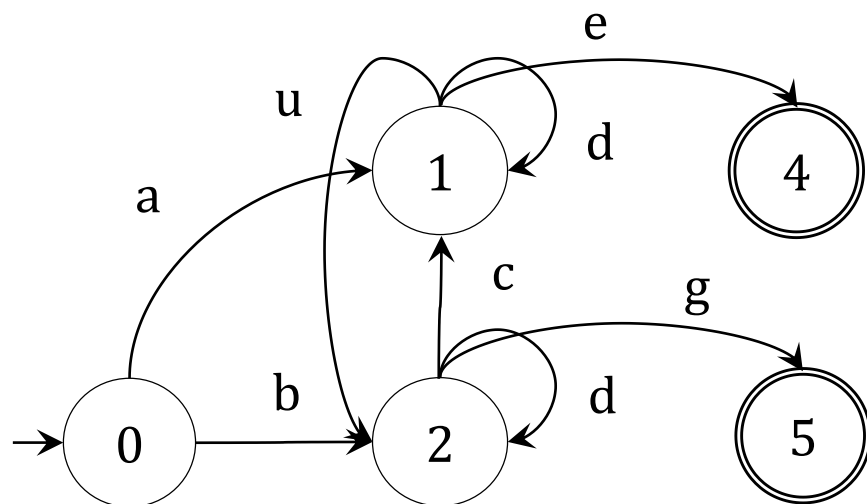
已知NFA如图所示，试完成以下3个小题。

- (1) 写出与该NFA等价的正则表达式；
- (2) 将该NFA等价地转换为DFA；
- (3) 判断所得DFA是否为最小，给出理由。





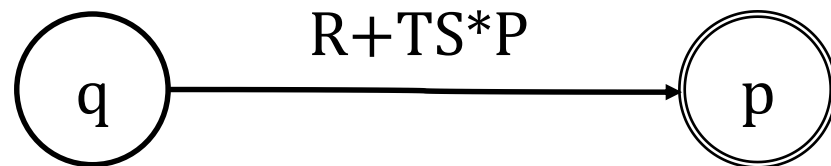
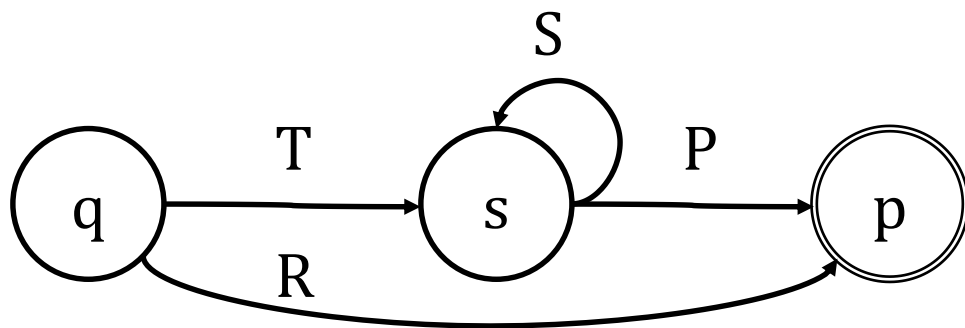
将DFA转为RE



$$\begin{aligned} & a(d+ud^*c)^*e \\ & +b(d+cd^*u)^*e \\ & +a(d+ud^*c)^*ud^*g \\ & +b(d+cd^*u)^*cd^*e \end{aligned}$$

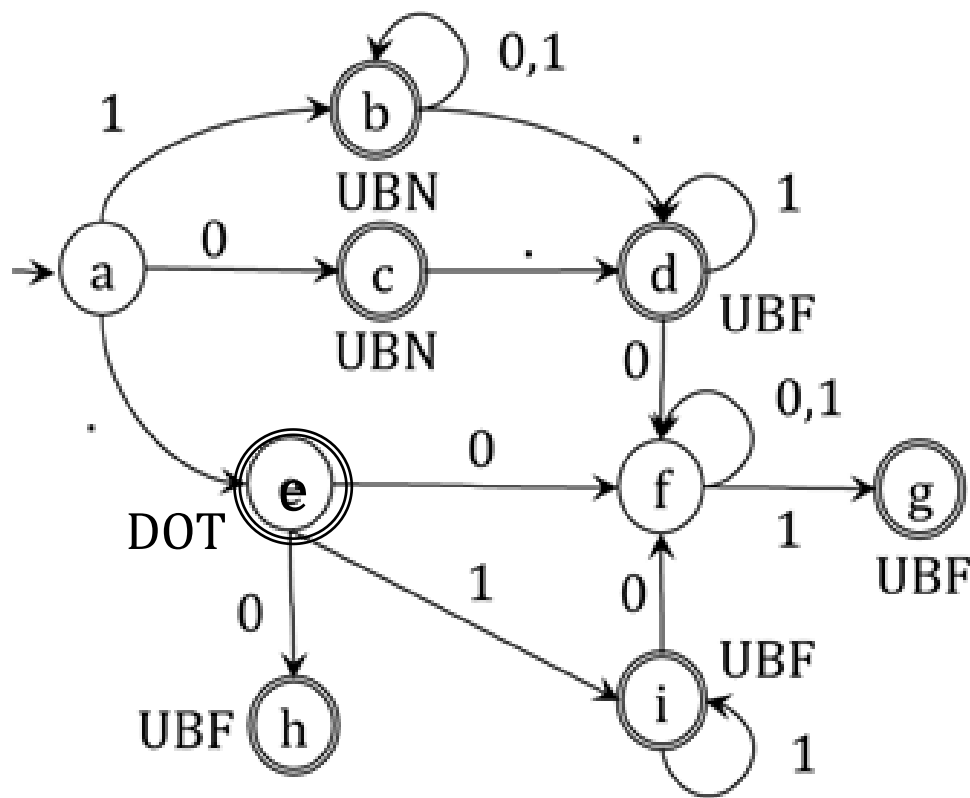
填表法/划分法

反证如状态4和5
等价即得结论





多语言联合DFA



- 对于 σ -DFA给定输入写出输出（词法记号用二元组表示）
- 输入1..10.1..01
- 输出(UBF, 1.)(UBF, .1)(UBF, 0.1)(DOT, _)(UBF, .01)



(1) 消除文法中的无用符号

$S \rightarrow AB|\varepsilon \quad A \rightarrow BC|a \quad C \rightarrow b$

▶ 消去无产出变元得 $S \rightarrow \varepsilon \quad A \rightarrow a \quad C \rightarrow b$

▶ 消去不可达变元得最终结果 $S \rightarrow \varepsilon$

(2) 消除文法中的 ε -产生式 $S \rightarrow AD|b \quad A \rightarrow aA|\varepsilon \quad D \rightarrow b|\varepsilon$

▶ 最终结果是 $S \rightarrow AD|D|A|b \quad A \rightarrow aA|a \quad D \rightarrow b$

▶ 另解：把D代入S中得 $S \rightarrow Ab|A|b \quad A \rightarrow aA|a$

▶ 另解：继续代入，把A代入S中得 $S \rightarrow aAb|aA|b \quad A \rightarrow aA|a$

(3) 消除文法中的单位产生式 $E \rightarrow T|iT \quad T \rightarrow F|Ti \quad F \rightarrow i|(E)$

▶ 结果是 $E \rightarrow i|(E)|Ti \quad |iT \quad T \rightarrow i|(E)|Ti \quad F \rightarrow i|(E)$

▶ 注：同一变元得候选式次序随意写；F产生式也可省略；



(4) 消除文法中的左递归 $\check{A} \rightarrow A | \check{A}A$; $A \rightarrow Td | Td[\check{I}]$ $\check{I} \rightarrow i | \check{I}, i$

- ▶ 最终结果是 $\check{A} \rightarrow A | A; \check{A}$ $A \rightarrow Td | Td[\check{I}]$ $\check{I} \rightarrow i | i, \check{I}$
- ▶ 另一结果 $\check{A} \rightarrow A\check{A}'$ $\check{A}' \rightarrow ; A\check{A}' | \varepsilon$ $A \rightarrow Td | Td[\check{I}]$ $\check{I} \rightarrow i\check{I}'$ $\check{I}' \rightarrow , \check{I}' | \varepsilon$
- ▶ 另一结果 $\check{A} \rightarrow A | A\check{A}'$ $\check{A}' \rightarrow ; A\check{A}'$. . .
- ▶ 注: \check{A} 和 \check{I} 的产生式都有左递归, 都要消除;



计算首符集和FOLLOW集

- ▶ 给定文法，写出每个变元的首符集和FOLLOW集
（注意明确计算方法，计算过程不要有遗漏）
- ▶ 检查该文法是不是满足LL(1)文法的条件
 - 同一变元的各个候选式的首符集两两不相交
 - 每个变元的首符集和FOLLOW集不相交（注意前提是首符集包含 ϵ ）
 - 注意：应该说明检查了什么，而不是简单地是或否。



预测分析表

- CFG (V, T, P, S) 是 LL(1) 文法，它的预测分析表 M 满足，对于任意 $(A, \gamma) \in P$,
- (1) 如果 $a \in \text{FIRST}(\gamma)$ 那么 $M[A, a] = (A, \gamma)$ 且
 - (2) 如果 $\epsilon \in \text{FIRST}(\gamma)$ 且 $a \in \text{FOLLOW}(A)$ 那么 $M[A, a] = (A, \gamma)$



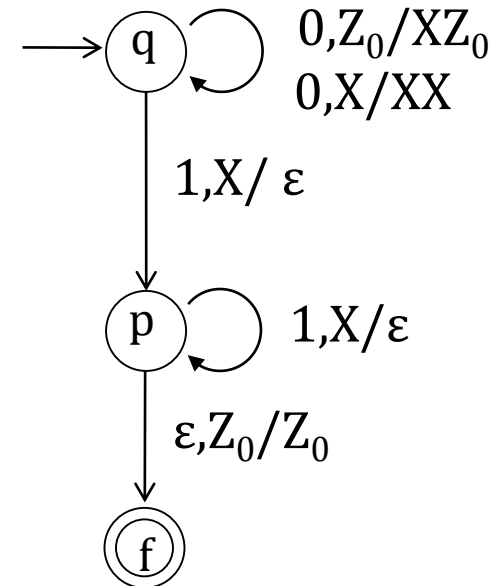
CFG判定性质

- ▶ 语法树
- ▶ 推导
- ▶ 最左推导
- ▶ 最右推导
- ▶ 规范规约
- ▶ 相关概念术语：产物、根、内节点、双亲结点、孩子结点、子树、句型、句子、短语、直接短语、句柄
- ▶ 语法树与文法对应关系
- ▶ 文法歧义性



- ▶ 瞬时描述 $ID(<状态>, <剩余串>, <栈内容>)$
- ▶ 移动: 0到多步直接移动, 其中直接移动 定义为,
 $ID(q, ax, X\alpha) \vdash ID(p, x, \gamma\alpha)$ 当且仅当 $(p, \gamma) \in \delta(q, a, X)$
- ▶ 判定性质: $ID(<初始状态>, <输入串>, Z_0) \vdash_P ID(<接受状态>, \varepsilon, \eta)$ 或 $ID(<初始状态>, w, Z_0) \vdash_N ID(<状态>, \varepsilon, \varepsilon)$
- ▶ 例, $(q, 000111, Z_0) \vdash (q, 00111, XZ_0) \vdash (q, 0111, XXZ_0) \vdash (q, 111, XXXZ_0) \vdash (p, 11, XXZ_0) \vdash (p, 1, XZ_0) \vdash (p, \varepsilon, Z_0) \vdash (f, \varepsilon, Z_0)$

$$\begin{aligned}\delta(q, 0, Z_0) &= \{(q, XZ_0)\} \\ \delta(q, 0, X) &= \{(q, XX)\} \\ \delta(q, 1, X) &= \{(p, \varepsilon)\} \\ \delta(p, 1, X) &= \{(p, \varepsilon)\} \\ \delta(p, \varepsilon, Z_0) &= \{(f, Z_0)\}\end{aligned}$$





➤ 例子中的移动序列为:

✓ $(q, 000111, Z_0) \vdash (q, 00111, XZ_0) \vdash (q, 0111, XXZ_0)$
 $\vdash (q, 111, XXXZ_0) \vdash (p, 11, XXZ_0) \vdash (p, 1, XZ_0)$
 $\vdash (p, \varepsilon, Z_0) \vdash (f, \varepsilon, Z_0)$

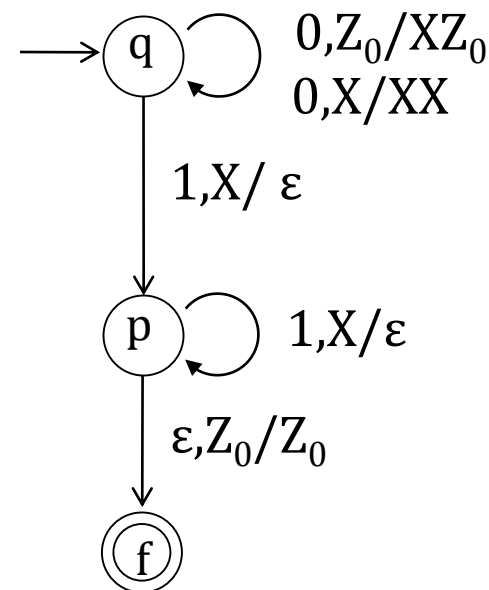
$$\delta(q, 0, Z_0) = \{(q, XZ_0)\}$$

$$\delta(q, 0, X) = \{(q, XX)\}$$

$$\delta(q, 1, X) = \{(p, \varepsilon)\}$$

$$\delta(p, 1, X) = \{(p, \varepsilon)\}$$

$$\delta(p, \varepsilon, Z_0) = \{(f, Z_0)\}$$



- 因此, $(q, 000111, Z_0) \vdash^* (f, \varepsilon, Z_0)$ 。
- 输入0001111会怎样?



itemDFA

- ▶ 规范规约模拟器
- ▶ 给定文法构建itemDFA（也称识别活前缀DFA）：
初始状态为 $\omega[S' \rightarrow .S]$ ；转移函数为 $v(q, X)=p$ ，如果
 $p = \omega \cup [A \rightarrow \rho.X\eta] \in q \bullet [A \rightarrow \rho X.\eta]$
 ω 含义为： $\omega[A \rightarrow \rho.N\eta] = \{[A \rightarrow \rho.N\eta]\} \cup \omega\{[N \rightarrow .\gamma] \mid (N, \gamma) \in P\}$ ，其中 ρ 和 η 为任意文法符号串。
注意不关心是否标注接受状态，即含有完全项目的状态。
- ▶ 概念术语：右句型、句柄、活前缀、有效项目、增广文法文法项目、初始项目、完全项目、移进项目
- ▶ SLR(1)文法，默认冲突消解规则。
- ▶ SLR(1)分析表。



写itemDFA (构造识别活前缀的DFA)

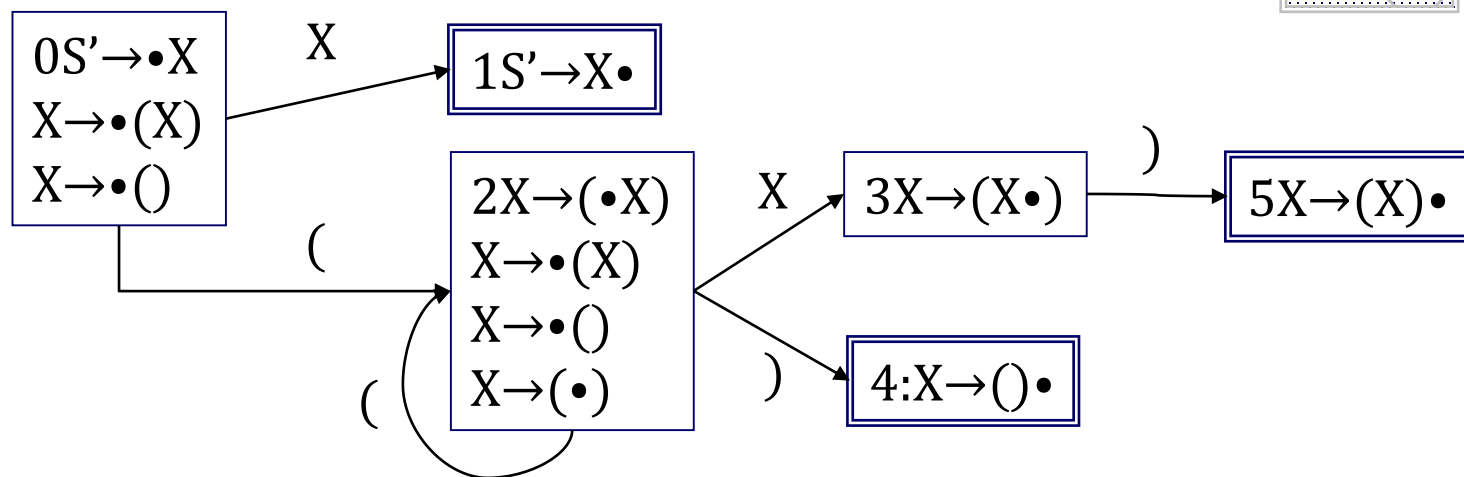
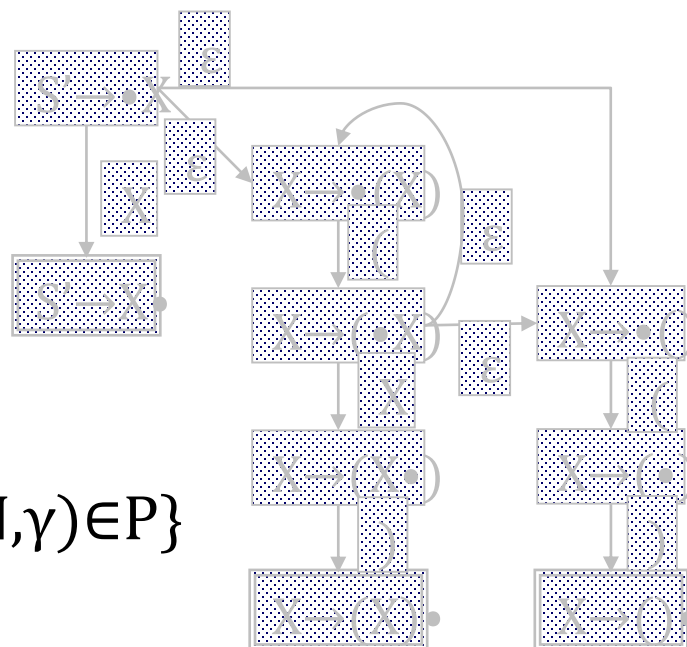
① ~~构建itemNFA并转itemDFA;~~

② 直接写出itemDFA;

➤ 初始状态为 $\omega[S' \rightarrow \cdot S]$; 转移函数为 $v(q, X) = p$, 如果

$$p = \omega \cup [A \rightarrow \rho \cdot X \eta] \in q \bullet [A \rightarrow \rho X \eta]$$

➤ $\omega[A \rightarrow \rho \cdot N \eta] = \{[A \rightarrow \rho \cdot N \eta]\} \cup \omega\{[X \rightarrow \cdot \gamma] \mid (N, \gamma) \in P\}$



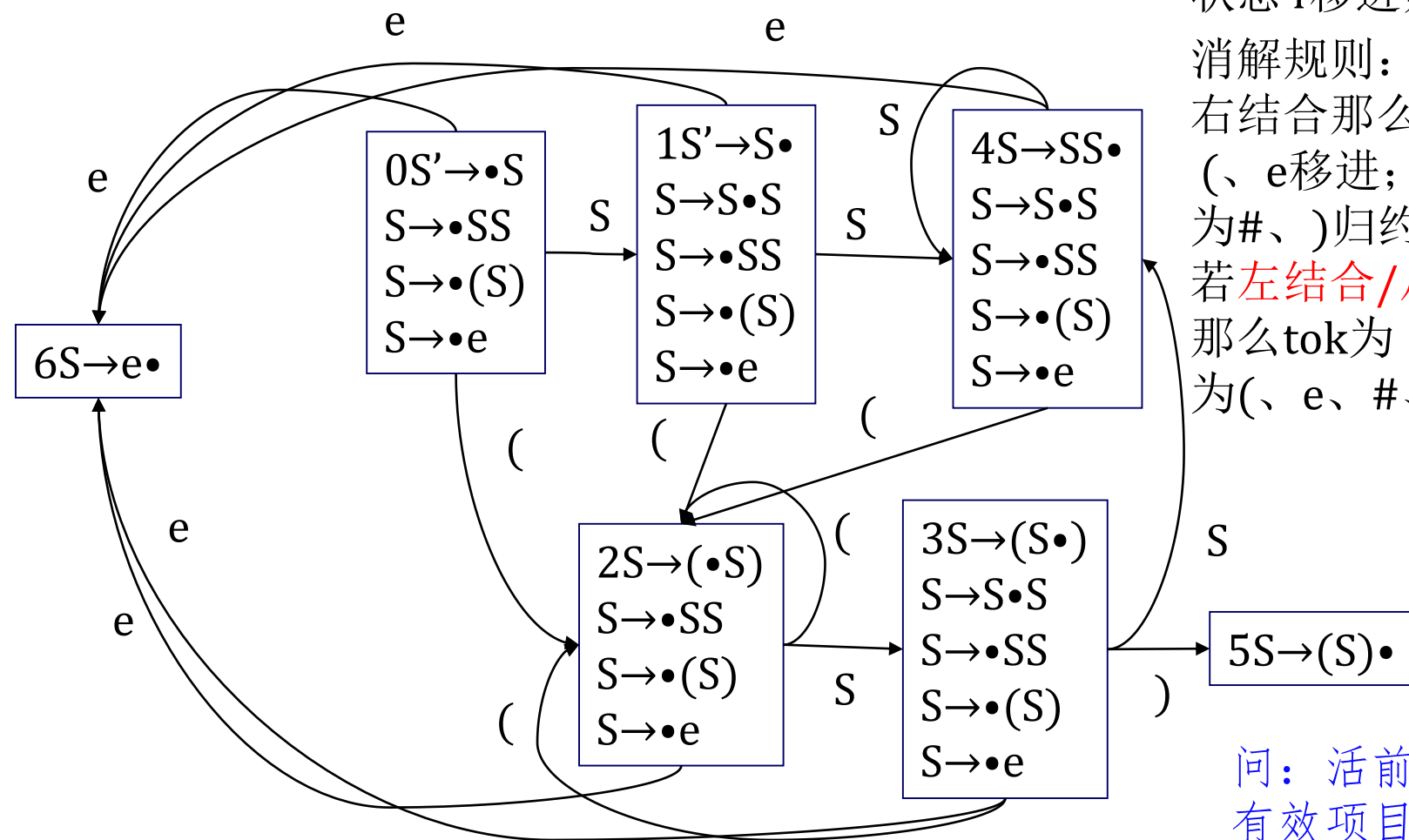
CFG G: $X \rightarrow (X) \quad X \rightarrow ()$

或G增广文法: $S' \rightarrow X \quad X \rightarrow (X) \quad X \rightarrow ()$



冲突消解

- $FOLLOW(S) = \{ \#,), (, e \}$ $FIRST(S) = \{ (, e \}$
- tok为(则移进, 为#或)则用 ϵ 候选式进行归约



状态1移进归约冲突
消解规则: tok为
(、e移进; #归约。
状态4移进归约冲突
消解规则: 若
右结合那么tok为
(、e移进;
为#、)归约。
若左结合/从左往右
那么tok为
为(、e、#、)归约

问: 活前缀((的
有效项目?2所示



► 例 对于下列程序试完成：

(1) 写出对程序声明中g()函数声明的语义分析结果。提示：不涉及中间代码生成；分析结果可采用自然语言描述，或者采用符号表表示均可。

(2) 分别写出第6至7行语句（属于g函数体部分）和第8行（属于最外层函数体）的中间语言代码。假定FLO类型宽度为8。

```
int x;  
float a[3, 6];  
int foo(int x){x=x+3; return x};  
int g(int r(); int y; float b[];){  
    int c[10];  
    if(y<1^b[2])y=r(5,) else y=a[1,2];  
    return y};  
g(foo(), 0, a[],)
```

例

```
▶ int g(int r(); int y; float b[];){
    int c[10];
    if(y<1^b[2])y=r(5,) else y=a[1,2]; return y}
```

```
@table:(outer:NULL width:164 rtype:INT argc:0 arglist:NIL level:0 code:...
entry:(name:x type:INT offset:4)
entry:(name:a type:ARRAY base:148 etype:FLO dims:2 dim[0]:3 dim[1]:6)
entry:(name:foo type:FUNC offset:156 mytab:foo@table)
entry:(name:g type:FUNC offset:164 mytab: g@table))

foo@table:(outer:@table width:4 argc:1 arglist:(x) rtype:INT level:1
code:... entry:(name:x type:INT offset:4))

g@table:(outer:@table width:56 argc:3 arglist:(r y b) rtype:INT level:1)
code:... entry:(name:r type:FUNPTT offset:8) entry:(name:y type:INT
offset:12) entry:(name:b type:ARRPTT offset: 16) entry:(name:c
type:ARRAY base:56 etype:INT dims:1 dim[0]:10))
```



也可用自然语言描述

- ▶ 共3个符号表，顶层为无名函数的，它的下层为foo和g的符号表。0层符号表没有形数，返回值任意类型，有x、b、foo和g共4个登记项，其中第一个的类型为INT，第二个为ARRAY，其余均为FUNC，它们的偏移量依次为4、148、156、164，所以顶层函数宽度为164。第二、三登记项都有指向自己符号表的指针域。第二个登记项还有维数2维长3、6元素类型FLO。
- ▶ ...



函数的中间代码生成

- $\text{if}(y < 1 \wedge b[2]) y = r(5,) \text{ else } y = a[1, 2]; \text{ return } y;$
- ```
[t1=1; IF y<t1 THEN l1 ELSE l2;
LABEL l1; t2=2; t3=t2*8; t4=b[t3];
IF t4!=0 THEN l3 ELSE l4;
LABEL l3; t5=5; PAR t5; t6=CALL r,1; y=t6; GOTO l5;
LABEL l2; LABEL l4; t7=1; t8=t7*6; t9=2; t10=t8+t9;
t11=t10*8; y=a[t11]; RETURN y; LABEL l5]
```
- $g(\text{foo}(), 0, a[], )$
- $[t1=0; \text{PAR } a; \text{PAR } t1; \text{PAR } \text{foo}; t2=\text{CALL } g, 3]$



# 运行时环境（栈快照）

```
int x;
int y;
void q(int s(); int x;){
 int y;
 if(x>3) y=s(x*5)
 else y=s(x+10);
 print y}
int p(){
 int r(int x;){
 int z;
 z=x+y;
 return z};
 q(r(), x,)}
x=27;
y=21;
p()
```

- ▶ 该程序涉及函数作为参数、函数声明的嵌套、局部和非局部名引用等语法结构。
- ▶ 写出对程序声明进行语义分析所得结果。
- ▶ 设栈底单元地址**500**，不考虑临时变量，在源程序上进行模拟执行。写出执行到第7行**print**语句时的栈快照。

```
int x;
int y;
void q(int s(); int x;){
 int y;
 if(x>3) y=s(x*5)
 else y=s(x+10);
 print y}
int p(){
 int r(int x;){
 int z;
 z=x+y;
 return z};
 q(r(), x,)}
x=27;
y=21;
p()
```

|     |              |
|-----|--------------|
| 500 | <访问链>:0      |
| 499 | <控制链>:0      |
| 498 | <返址>         |
| 497 | x:27         |
| 496 | y:21         |
| 495 | q[1]         |
| 494 | q[0]         |
| 493 | p[1]         |
| 492 | p[0]         |
| 491 | <访问链>:499    |
| 490 | <控制链>:499    |
| 489 | <返址>         |
| 488 | r[1]:r@label |
| 487 | r[0]:490     |
| 486 | <参数2>:27     |
| 485 | <参数1>:480    |
| 484 | <访问链>:499    |
| 483 | <控制链>:490    |
| 482 | <返址>         |
| 481 | s[1]:r@label |
| 480 | s[0]:490     |
| 479 | x:27         |
| 478 | y:156        |

|     |           |
|-----|-----------|
| 477 | <参数1>:135 |
| 476 | <访问链>:490 |
| 475 | <控制链>:483 |
| 474 | <返址>      |
| 473 | x:135     |
| 472 | z:156     |



# 后续工作

- ▶ 答疑：线上即时答疑；线下临近安排届时通知

预祝大家考试取得好成绩！