

誠朴雄偉
勵學敦行

编译原理习题课（一）





作业等第



- 作业共分A, B, C, D四个等第
 - A: 答题正确, 书面美观, 过程完整
 - B: 答题基本正确, 有必要的过程
 - C: 答题正确率一般, 书面较差, 过程不完整
 - D: 未在规定时间内提交
-
- 原则上A, B皆为满分, 第一次作业提交即为满分
 - (实际上所有作业大部分都是满分)



课外学习资料



- 龙书核心作者所授课程 (STU, CMU)
 - STU-CS143, CS243, CS343 逐步深入
- CS143 - Compilers (Instructor: Alex Aiken)
 - Compiler基础知识, 完整介绍了COOL语言的实现过程
 - COOL语言: 面向教学的编程语言, 同时兼有OO特性
 - Lecture: [课程官网](#) (2021版)
 - Video: [B站](#)有部分熟肉, [edX](#)有完整生肉



课外学习资料



■ CS243 - Advanced Compilers

- 编译器的结构都十分类似，Fortran开始结构比重变化：
- 工作重心：从解析器（Parser）到优化（Optimization）
- Lecture-21: Program Analysis and Optimization
- Lecture-06: (Ullman) : Advanced Compiling Techniques
- 缺少视频，21版讲义过于简单，06版适用于自学

■ CS343 - Advanced Topics in Compilers

- 领域经典论文，以不同Topic为章节
- 选用论文截止2014年



第三章



1. 试描述该正则表达式定义的语言：

$$(a|b)^*a(a|b)(a|b)$$

a和b组成的字符串

倒数第三位一定是a

最后两位是a和b组成的字符串

由a和b组成的字符串，且倒数第三个字符为a。

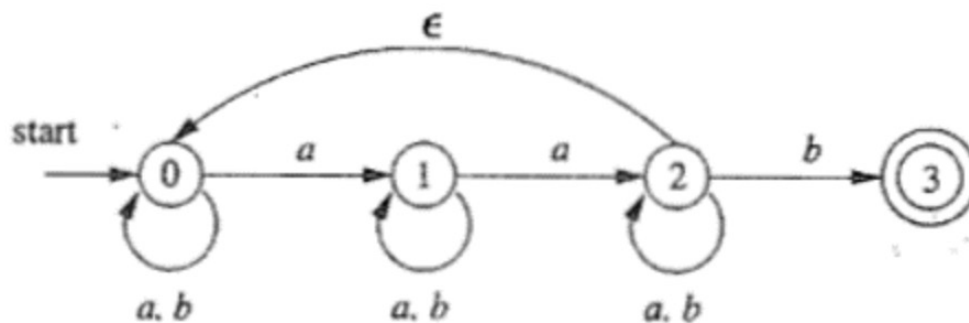
- 解不唯一，回答尽量简洁
- 书图3-5



第三章



2. 找出下图NFA中所有标号为aabb的路径，这个NFA接受aabb吗？



标号为aabb的路径为：00000, 00111, 01111, 01222, 012220, 01223, 012000, 012200。

接受aabb，如路径01223。



第三章



3. 将下图中的NFA转化为DFA:

(书算法3.20) 首先构造子集:

A: $\epsilon\text{-closure}(0) = \{0\}$

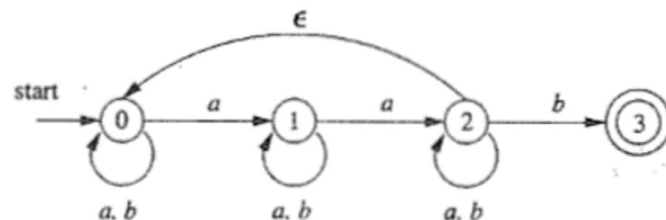
B: $D\text{tran}[A,a] = \epsilon\text{-closure}(\text{move}(A,a)) = \epsilon\text{-closure}(0,1) = \{0,1\}$

C: $D\text{tran}[B,a] = \epsilon\text{-closure}(\text{move}(B,a)) = \epsilon\text{-closure}(0,1,2) = \{0,1,2\}$

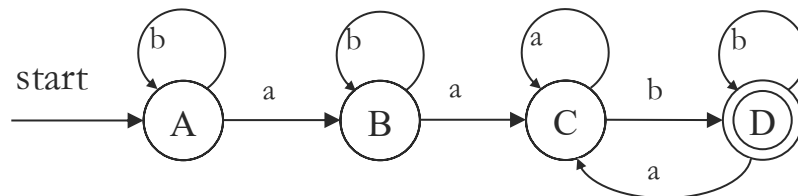
D: $D\text{tran}[C,b] = \epsilon\text{-closure}(\text{move}(C,b)) = \epsilon\text{-closure}(0,1,2,3) = \{0,1,2,3\}$

构造状态转换表:

NFA状态	DFA状态	Input-a	Input-b
0	A	B	A
0, 1	B	C	B
0, 1, 2	C	C	D
0, 1, 2, 3	D	C	D

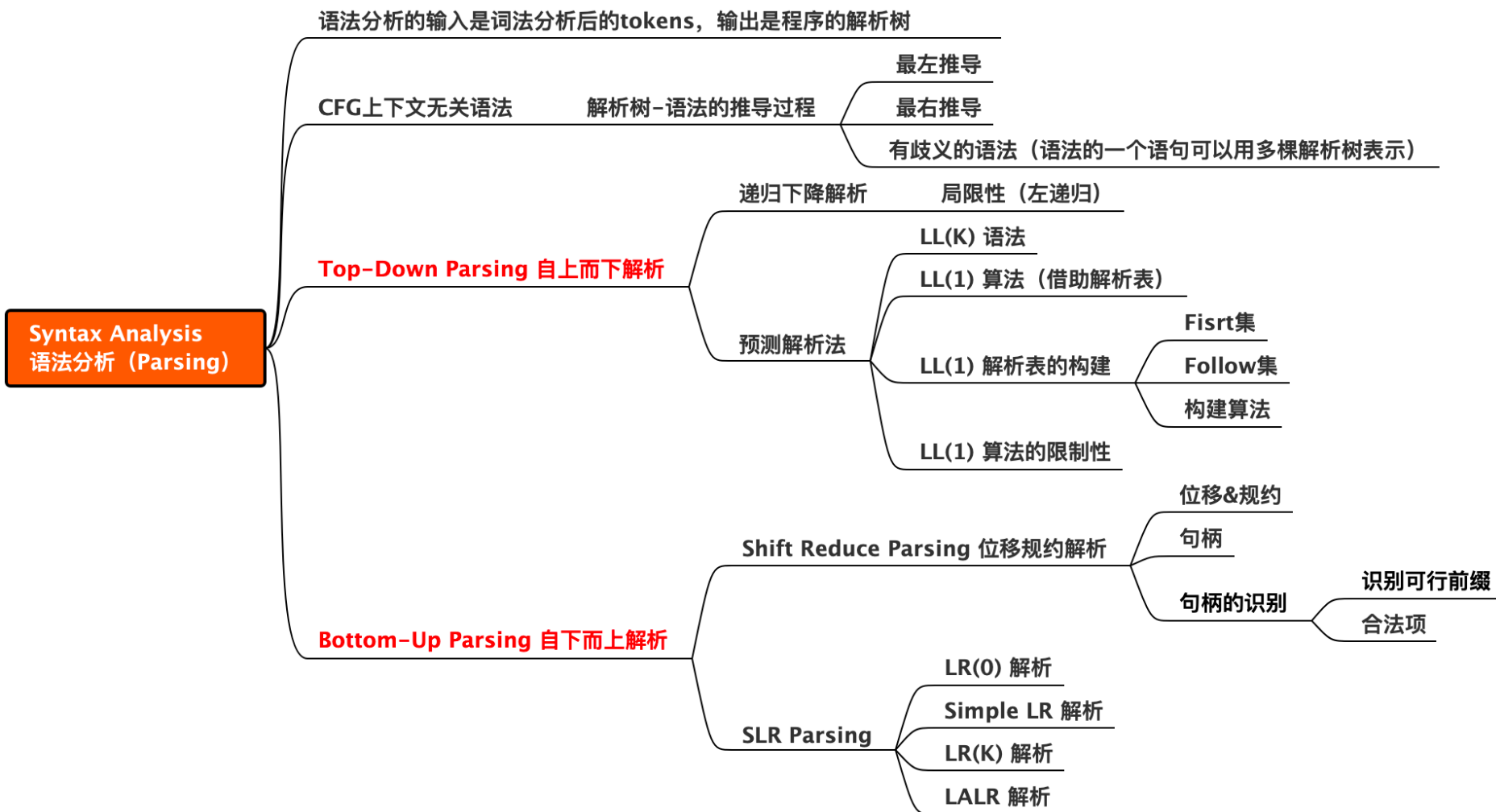


DFA:





课后作业一 (Sec1-4)





课后作业一 (Sec1-4)



1. 考虑上下文无关文法： $S \rightarrow SS+ | SS^* | a$ 以及串 $aa+a^*$

(1) 给出这个串的一个最左推导：

$$\text{lm: } S \Rightarrow \underline{S}S^* \Rightarrow \underline{S}S+S^* \Rightarrow a\underline{S}+S^* \Rightarrow aa+S^* \Rightarrow aa+a^*$$

(2) 给出这个串的一个最右推导：

$$\text{rm: } S \Rightarrow S\underline{S}^* \Rightarrow \underline{S}a^* \Rightarrow S\underline{S}+a^* \Rightarrow \underline{S}a+a^* \Rightarrow aa+a^*$$



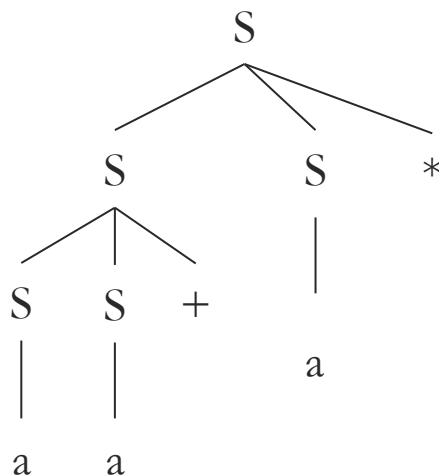
第四章



1. 考虑上下文无关文法： $S \rightarrow SS+ | SS^* | a$ 以及串 $aa+a^*$

(3) 给出这个串的一颗语法分析树；

以最左推导为例： $\text{lm: } S \Rightarrow \underline{S}S^* \Rightarrow \underline{S}S+S^* \Rightarrow a\underline{S}+S^* \Rightarrow aa+S^* \Rightarrow aa+a^*$





第四章



2. 考虑上下文无关文法： $S \rightarrow 0S1 \mid 01$

给出该文法的预测分析表（需要先提取左公因子并消除左递归）

提取左公因子（书算法4.10）：

$S \rightarrow 0A$

$A \rightarrow S1 \mid 1$

预测分析表：

非终结 符号	输入符号		
	0	1	\$
S	$S \rightarrow 0A$		
A	$A \rightarrow S1$	$A \rightarrow 1$	



第四章



3. 考虑上下文无关文法： $S \rightarrow SS+ | SS^* | a$ 以及串 $aa+a^*$

(1) 给出该文法的First集和Follow集；

开始符号一定勿忘\$

$$\text{First}(S) = \{a\} \quad \text{Follow}(S) = \{+, *, a, \$\}$$

(2) 指出下列最右句型的句柄：

$SSS+a^{*}+$

句柄: $SS+$

$SS+a^{*}a+$

句柄: $SS+$

$aaa^{*}a++$

句柄: a

句型：包含非终结符和终结符的串，可以是空串。

最右句型：包含了一个句柄，可以完成一次最右推导。

句柄：和某个产生式体匹配的子串（非正式）

(3) 给出串 $aaa^{*}a++$ 自底向上的解析过程；



第四章



3. 考虑上下文无关文法： $S \rightarrow SS+ | SS^* | a$ 以及串 $aa+a^*$

(3) 给出串 aaa^*a++ 自底向上的解析过程：

栈	输入	句柄	动作
\$	$aaa^*a++\$$		移入a
\$a	$aa^*a++\$$	a	规约： $S \rightarrow a$
\$S	$aa^*a++\$$		移入a
\$Sa	$a^*a++\$$	a	规约： $S \rightarrow a$
\$SS	$a^*a++\$$		移入a
\$SSa	$*a++\$$	a	规约： $S \rightarrow a$
\$SSS	$*a++\$$		移入*
\$SSS^*	$a++\$$	SS^*	规约： $S \rightarrow SS^*$
\$SS	$a++\$$		移入a

栈	输入	句柄	动作
\$SSa	$++\$$	a	规约： $S \rightarrow a$
\$SSS	$++\$$		移入+
\$SSS+	$+\$$	$SS+$	规约： $S \rightarrow SS+$
\$SS	$+\$$		移入+
\$SS+	$\$$	$SS+$	规约： $S \rightarrow SS+$
\$S	$\$$		接受

关于句柄的总结：

1. 句柄永远出现在 **栈顶**；
2. 句柄永远不会出现在非终结符的左侧；
3. 自下而上解析完全基于 **句柄的识别**；



随堂测试



将下列正则表达式转换成DFA，并将DFA最小化：

$(a^* | b^*)^*$