编译原理与技术

计算机科学与技术学院 王一多,许冠斌



HW 1

请上网查找并结合自身实践经验,列举不少于十种编译器,并提供如下信息 (不局限于):

- 适用语言
- 出现的时间
- 代表人物
- 发展过程的简单描述
- 与其他类似编译器的对比



■■ HW 1 出现的主要问题

- IDE & 编译器 & 语言
- 原封不动复制

举例:

C: 一种语言

GCC: C语言的一种编译器

Dev-C++: 一套适用于C/C++的IDE,使用MinGW编译





HW 1 部分同学作业

LLVM

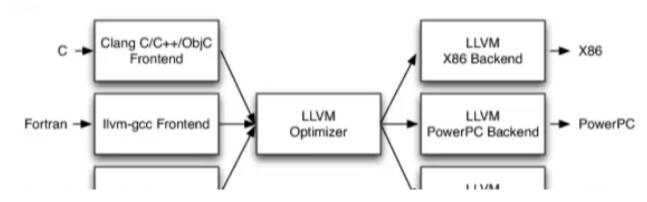
LLVM是Low Level Virtual Machine的简称,是一个编译器架构,提供了与编译器有关的支持,能够进行程序语言的编译期优化、链接优化、在线编译优化、代码生成。简而言之,可以作为多种语言编译器的后台来使用。

LLVM是一个开源项目,启动于2000年,最初由美国UIUC大学的**Chris Lattner**博士主持开展。2006年**Chris Lattner**加盟Apple Inc.并致力于LLVM在Apple开发体系中的应用。Apple也是LLVM计划的主要资助者。

llvm的主要作用是它可以作为多种语言的后端,它可以提供可编程语言无关的优化和针对很多种CPU的代码生成功能。此外llvm目前已经不仅仅是个编程框架,还包含了很多的子项目,比如最具盛名的clang.

LLVM有优点:

- 开源
- 模块化设计, 便于排除漏洞和维护
- 使用LLVM IR 中间代码,分离编译器前端和后端的设计,使其能够更为高效开发
- 提供了一系列工具,可以比较容易的实现新出现的编程语言的编译器开发,并复用现有的优化/调试手段,提高了效率







■ HW 1 部分同学作业

8. Cpython

适用语言

python

出现的时间

2000-10-16第一版

代表人物

Python core developers and the Python community, supported by the Python Software Foundation

发展过程的简单描述

作为python语言的默认编译器,使用广泛

与其他类似编译器的对比

Cpython是解释器,官方版本支持全面。但有几个缺陷,一是全局锁使Python在多线程效能上表现不佳,二是CPython无法支持JIT(即时编译),导致其执行速度不及Java和Javascipt等 语言。



《编译原理第3版 陈意云、张昱编著》第二章 2.1 e, 2.2, 2.3 e, 2.4 a,f,i 《编译原理第3版 陈意云、张昱编著》第二章 2.7 c 2.11 2.15



题目:从下列每种语言的参考手册确定它们形成输入字母表的字符集(不包括那些只可以出现在字符串或注释中的字符):SQL

本题解答合理即可

常见错误:将字符集理解为编码,如写utf-8等



2.2题目

题目: 在下面的C 函数中,按序列出所有的记号,并给每个记号以合理的属性值

```
long gcd(long p, long q) {
  if ( p%q == 0 )
    /* then part */
  return q;
else
/* else part */
return gcd( q, p%q );
}
```





2.2答案举例,合理即可

记号名	属性	
type	64bit_type	
id	指向 "gcd" 的指针	
(
type	64bit_type	
id	指向 "p" 的指针	
,		
type	64bit_type	
id	指向 "q" 的指针	
)		
{		
if		
(
id	指向 "p" 的指针	

记号名	属性	
operator	mod_operator	
id	指向 "q" 的指针	
operator	equal_operator	
number	0 (integer)	
)		
comment _begin		
comment	"then part"	
comment _end		
return		
id	指向 "q" 的指针	
,		
else		

记号名	属性
comment_begin	
comment	"else part"
comment_end	
return	
id	指向 "gcd" 的指针
(
id	指向 "q" 的指针
,	
id	指向 "p" 的指针
operator	mod_operator
id	指向 "q" 的指针
)	
,	
}	



____2.3 e

题目: 叙述以下正规式描述的语言

 $(00 \mid 11)*((01 \mid 10)(00 \mid 11)*(01 \mid 10)(00 \mid 11)*)*$

答案: 匹配由偶数个0和偶数个1组成的所有0和1的串。

思考:

1: 这个正规式产生的串是否都是偶数个0和1的串?

2: 这个正规式是否能够匹配所有偶数个0和1的串?



2.4 a

题目:写出正规式定义:

包含5个元音的所有字母串,其中每个元音只出现一次且按顺序排列。

答案: 设 Consonant = [b-d]|[f-h]|[j-n]|[p-t]|[v-z]|[B-D]|[F-H]|[J-N]|[P-T]|[V-Z]

Consonant*[aA]Consonant*[eE]Consonant*[iI]Consonant*[oO]Consonant*[uU]Consonant*

常见错误:

- 1. 未考虑大小写
- 2. 理解错误题意(包含元音字母的字母串也可以包含非元音字母)



2.4 f

题目:写出正规式定义:

由偶数个0和偶数个1构成的所有01串。

答案:参考2.3e

 $(00 \mid 11)*((01 \mid 10)(00 \mid 11)*(01 \mid 10)(00 \mid 11)*)*$



写出正规式定义:字母表{a,b}上,a,b不会相邻出现的所有串

答案举例:

(b|ab)*a?

答案不止一种形式,但以下都不对:

 $b^*(ab|b)^*$ 或 $b^*((\epsilon|a)b+)^*$ 或 $(b^*(ab)^*b^*(ba)^*b^*)^*$

尝试找出反例



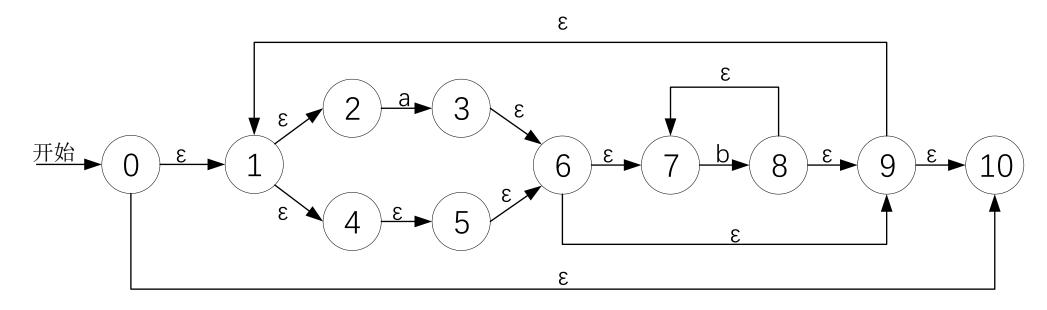
2.7 c

题目:用算法2.4为下列正规式构造不确定有限自动机,给出它们处理输入串 ababbab 的状态转换序列。

正规式: ((ε|a)b*)*



严格按照**P32算法2.4**构造($(\epsilon|a)b^*$)*的NFA,应该是如下图所示:

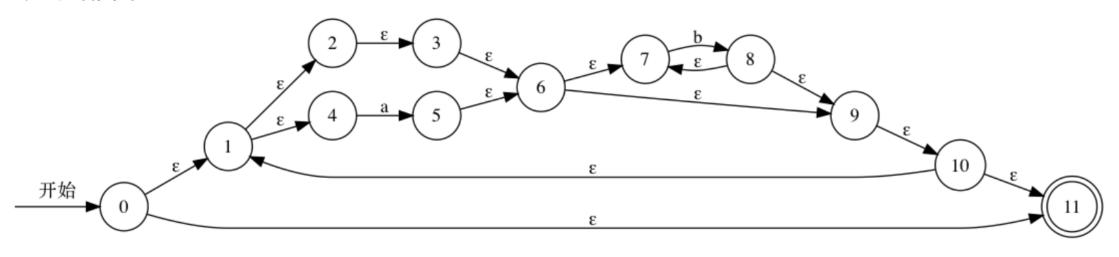


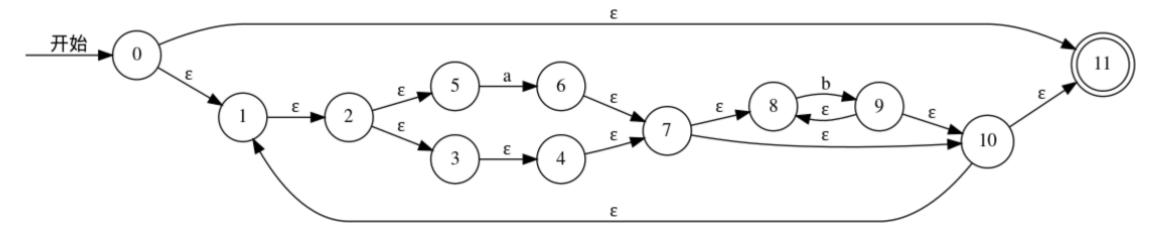
ababbab 状态转换序列:

$$0 \to 1 \to 2 \to 3 \to 6 \to 7 \to 8 \to 9 \to 1 \to 2 \to 3 \to 6 \to 7 \to 8 \to 7 \to 8 \to 9 \to 1 \to 2 \to 3 \to 6 \to 7 \to 8 \to 9 \to 10$$



以下都算对

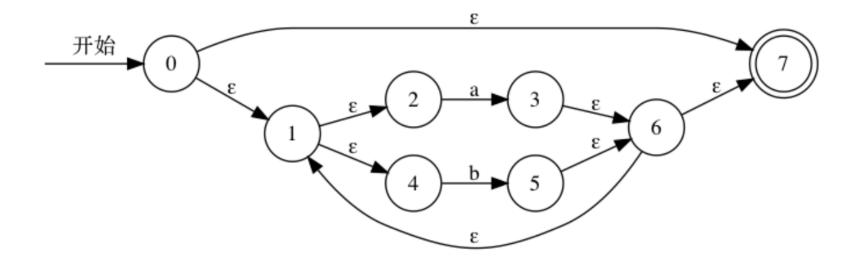






■ 2.11-1: 绘制NFA

绘制 (a|b)* 的 NFA 如下:







2.11-1: NFA转DFA

对此 NFA 转换为 DFA。

A =
$$\{0, 1, 2, 4, 7\}$$

B = $\{3, 6, 7, 1, 4\}$, $A \xrightarrow{a} B$
C = $\{5, 6, 1, 7, 2, 4\}$, $A \xrightarrow{b} C$

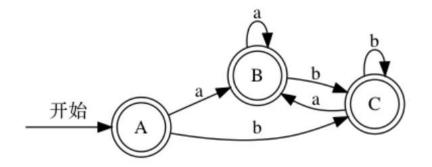
状态表:

	a	b
А	В	С
В	В	С
С	В	С

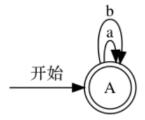


■ 2.11-1: 化简DFA

画出DFA



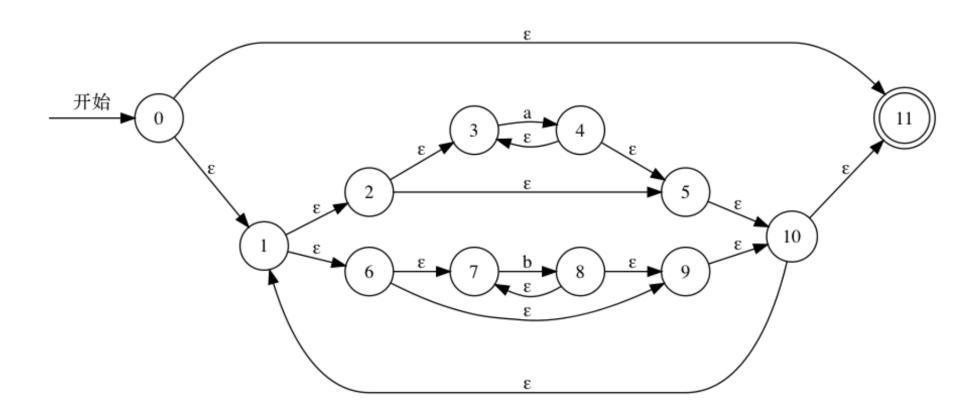
接受状态{A, B, C}, 非接收状态Ø, 而接受状态不需要再划分, 化简。





2.11-2

绘制(a*|b*)* 的 NFA 如下:







对此 NFA 转换为 DFA:

A = {0, 1, 2, 6, 3, 7, 5, 9, 10, 11}
B = {4, 5, 10, 1, 2, 6, 3, 7, 5, 9, 11},
$$A \xrightarrow{a} B$$

C = {8, 9, 10, 1, 2, 6, 3, 7, 5, 9, 11}, $A \xrightarrow{b} C$

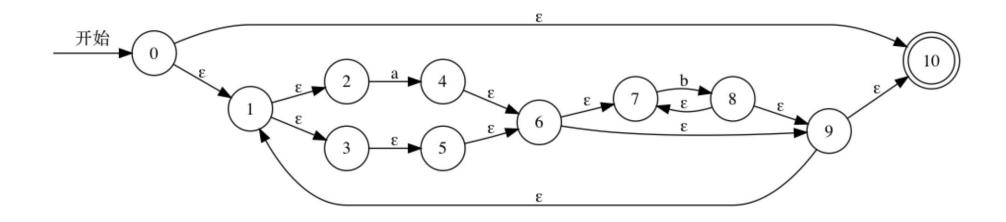
状态表:

	a	b
Α	В	С
В	В	С
С	В	С

状态表与前一个正规式一致,则构造出最简DFA也相同



绘制((ε|a)b*)*的 NFA 如下:





对此 NFA 转换为 DFA:

$$A = \{0, 1, 2, 3, 5, 6, 7, 9, 10\}$$

B =
$$\{4, 6, 7, 9, 1, 2, 3, 5, 10\}, A \stackrel{a}{\rightarrow} B$$

$$C = \{8, 9, 1, 2, 3, 5, 6, 7, 10\}, A \xrightarrow{b} C$$

状态表:

	a	b
Α	В	С
В	В	С
С	В	С

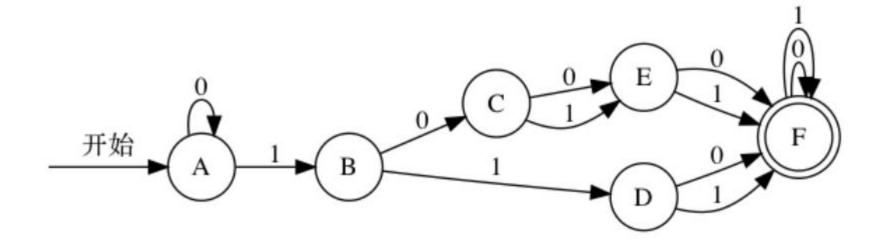
状态表与前两个正规式一致,所以三种正规式等价



题目: 构造一个最简DFA,接受所有大于101的二进制数

答案因人而异,合理即可,但是后面必须能看出是最简DFA,如果按照书上 P30算法2.3, 就是对的,这里举例

首先手工构造DFA如下





根据接受状态划分为 {A, B, C, D, E} 与 {F}

 $move(\{D, E\}, 0)=\{F\}$

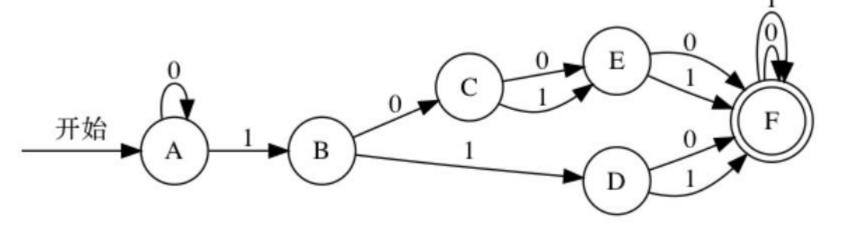
 $move(\{D, E\}, 1)=\{F\}$

 $move({A, B, C}, 0)={A, C, E}$

 $move({A, B, C}, 1)={B, D, E}$

划分为{A, B, C}, {D, E}, {F}

 $move({A}, 0)={A}$ $move({A}, 1)={B}$ $move({B, C}, 0)={C, E}$ $move({B, C}, 1)={D, E}$

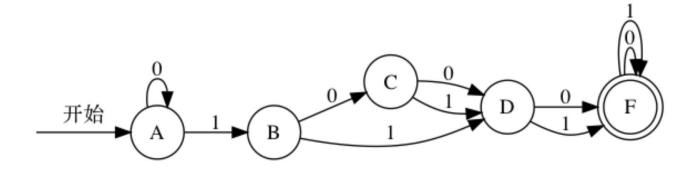




2.15

继续划分为{A}, {B, C}, {D, E}, {F} move({B}, 0)={C} move({B}, 1)={D} move({C}, 0)={E} move({C}, 1)={E}

最终划分为{A}, {B}, {C}, {D, E}, {F}, 合并D, E得





《编译原理第3版 陈意云、张昱编著》第三章 3.1 a 和 b, 3.6 b和c 《编译原理第3版 陈意云、张昱编著》第三章 3.8,3.10,3.11,3.12





文法如下:

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

 $L \rightarrow L, S \mid S$

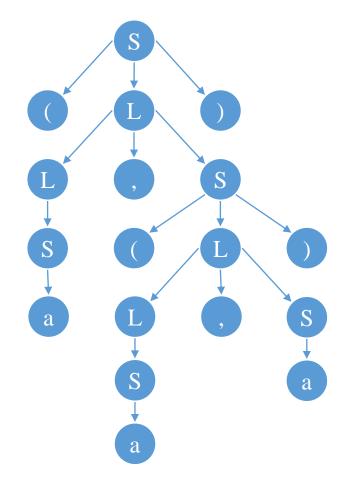
建立句子(a, (a, a))和(a, ((a, a), (a, a)))的分析树



(a, ((a, a), (a, a)))的分析树

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

 $L \rightarrow L, S \mid S$





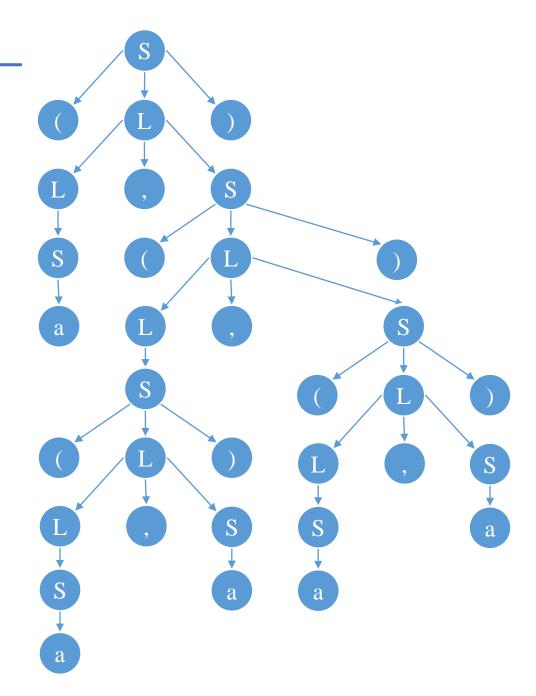


HW 3 - 3.1a

(a, (a, a))的分析树

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

 $L \rightarrow L, S \mid S$







为句子(a, (a, a))构造最左推导,如下

- \Rightarrow (L)
- \Rightarrow (L, S)
- \Rightarrow (S, S)
- \Rightarrow (a, S)
- \Rightarrow (a, (L))
- \Rightarrow (a, (L, S))
- \Rightarrow (a, (S, S))
- \Rightarrow (a, (a, S))
- \Rightarrow (a, (a, a))

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

 $L \rightarrow L, S \mid S$





为句子(a, ((a, a), (a, a)))构造最左推导,如下

S

$$\Rightarrow$$
 (L)

$$\Rightarrow$$
 (L, S)

$$\Rightarrow$$
 (S, S)

$$\Rightarrow$$
 (a, S)

$$\Rightarrow$$
 (a, (L))

$$\Rightarrow$$
 (a, (L, S))

$$\Rightarrow$$
 (a, (S, S))

$$\Rightarrow$$
 (a, ((L), S))

$$\Rightarrow$$
 (a, ((L, S), S))

$$\Rightarrow$$
 (a, ((S, S), S))

$$\Rightarrow$$
 (a, ((a, S), S))

$$\Rightarrow$$
 (a, ((a, a), S))

$$\Rightarrow$$
 (a, ((a, a), (L)))

$$\Rightarrow$$
 (a, ((a, a), (L, S)))

$$\Rightarrow$$
 (a, ((a, a), (S, S)))

$$\Rightarrow$$
 (a, ((a, a), (a, S)))

$$\Rightarrow$$
 (a, ((a, a), (a, a)))

题目:为字母表 $\Sigma = \{a, b\}$ 上的语言设计文法,其中哪些是正规的

(b): a和b个数相等的所有串

(c): a和b个数不相等的所有串





(b): a和b个数相等的所有串

思考:

字母表上的ab串可分为三种

- 1. ab数量相等
- 2. a比b多(可分成若干个a比b多一个的串)
- 3. b比a多(可分成若干个b比a多一个的串)





解法1:

 $S \rightarrow aSbS \mid bSaS \mid \epsilon$

解法2:

 $S \rightarrow aB \mid bA \mid \epsilon$

 $A \rightarrow aS \mid bAA$

 $B \rightarrow bS \mid aBB$

解法3:

 $S \rightarrow abS \mid aSb \mid Sab \mid baS \mid bSa \mid Sba \mid \epsilon$

不是正规的(部分同学忘记判断)





解法思路:在3.6b的思路基础上,加上若干个a或b即可(注意:不能是空串)

首先

 $S \rightarrow AaA \mid BbB$

如:在解法1的基础上:

A → aAbA | bAaA | aA | ε (A代表a大于等于b的串)

 $B \rightarrow aBbB \mid bBaB \mid bB \mid \epsilon$

解法不唯一,在你所做的3.6b的基础上,推导即可

不是正规的(部分同学忘记判断)





题目:消除下面文法的左递归。

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

 $L \rightarrow L, S \mid S$

答案:

按书上消除直接左递归的算法,可得

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

$$L \rightarrow SL$$

$$L' \rightarrow ,SL' \mid \varepsilon$$



画出预测分析表如下

	()	,	a	\$
S	$S \rightarrow (L)$			$S \rightarrow a$	
L	$L \rightarrow SL'$			$L \rightarrow SL'$	
L'		$L' \rightarrow \epsilon$	L'→ ,SL'		$L' \rightarrow \epsilon$

接着按照书上P57图3.8给出预测分析器伪代码即可





■ HW 3 - 3.8b预测分析器伪代码示例

	()	,	a	\$
S	$S \rightarrow (L)$			$S \rightarrow a$	
L	$L \rightarrow SL'$			$L \rightarrow SL'$	
L'		$L' \rightarrow \epsilon$	$L' \rightarrow ,SL'$		$L' \rightarrow \epsilon$

```
void match ( terminal t ) {
    if ( lookahead == t ) lookahead = nextToken();
    else error();
void S() {
   if ( lookahead == '(' ) {
        match('(');
        L();
        match(')');
    } else if ( lookahead == 'a' ) {
        match('a');
   } else {
        error();
void L() {
   S();
   L1();
void L1() {
    if ( lookahead == ',' ) {
        match(',');
        S();
        L1();
    } else {
        match('');
```





题目:构造下面文法的LL(1)分析表

 $D \rightarrow TL$

 $T \rightarrow int \mid real$

 $L \rightarrow id R$

 $R \rightarrow$, id $R \mid \epsilon$



HW 3 - 3.10

根据文法构造First集和Follow集

```
First(TL)
                                          ={int real}
D \rightarrow TL
T \rightarrow int
                    First(int)
                                          =\{int\}
T \rightarrow real
                   First(real)
                                          ={real}
L \rightarrow id R
                   First(id R)
                                          =\{id\}
R \rightarrow, id R
                   First(, id R)
                                          =\{,\}
                   First(ε)
                                         =\{\epsilon\}
R \rightarrow \epsilon
                                          =Follow{L}=Follow{D}={$}
                    Follow(R)
```





构造LL(1)分析表

	int	real	id	,	\$
D	$D \rightarrow TL$	$D \rightarrow TL$			
T	$T \rightarrow int$	$T \rightarrow real$			
L			$L \rightarrow id R$		
R				$R \rightarrow$, id R	$R \rightarrow \epsilon$



HW 3 - 3.11

题目:构造下面文法的LL(1)分析表

 $S \rightarrow aBS \mid bAS \mid \epsilon$

 $A \rightarrow bAA \mid a$

 $B \rightarrow aBB|b$





HW 3 - 3.11: 构造First集和Follow集

文法:

 $S \rightarrow aBS \mid bAS \mid \epsilon$

 $A \rightarrow bAA \mid a$

 $B \rightarrow aBB|b$

	First	Fellow
S	a, b, ε	\$
A	a, b	a, b, \$
В	a, b	a, b, \$

	a	b	\$
S	$S \rightarrow aBS$	$S \rightarrow bAS$	S→ε
A	$A \rightarrow a$	$A \rightarrow bAA$	
В	$B\rightarrow aBB$	$B \rightarrow b$	

First集和Follow集

LL(1)分析表





题目:下列文法是否为LL(1)文法

 $S \rightarrow AB \mid PQx$

 $A \rightarrow xy$

 $B \rightarrow bc$

 $P \rightarrow dP \mid \epsilon$

 $Q \rightarrow aQ | \epsilon$





思考:如何证明?

P56: 对任意两个产生式 $A \rightarrow \alpha | \beta$ 都有:

1) First(α) \cap First(β)= Φ

2) 若β⇒* ε, 则First(A)∩ Follow(A)=Φ

如果要证明不是LL(1)文法,

那么只要能找到反例即可





```
S \rightarrow AB \mid PQx;
```

 $A \rightarrow xy$; $B \rightarrow bc$

 $P \rightarrow dP \mid \epsilon; \quad Q \rightarrow aQ \mid \epsilon$

由于只有S,P,Q三种非终结符有两种推导选择,逐一分析

P的右部: dP、ε,开始符号肯定不相交

Q的右部: aQ、ε, 开始符号肯定不相交

S的右部: AB、PQx

由于PQx的First集含有x(当P、Q均推出ε)

AB的First含有x

 $First(AB) \cap First(PQx)! = \Phi$

得证该文法不是LLQ





Q&A

