



北京邮电大学计算机学院(国家示范性软件学院)

SCHOOL OF COMPUTER SCIENCE(NATIONAL PILOT SOFTWARE ENGINEERING SCHOOL)BUPT

编译原理与技术习题课

王吴凡

2024年12月24日

课程安排

1. 习题易错点分析
2. 重要知识点总结

■ 习题3.4 (考察正规表达式、有限状态机、线性文法)

构造一文法，使其语言是无符号偶整数的集合。

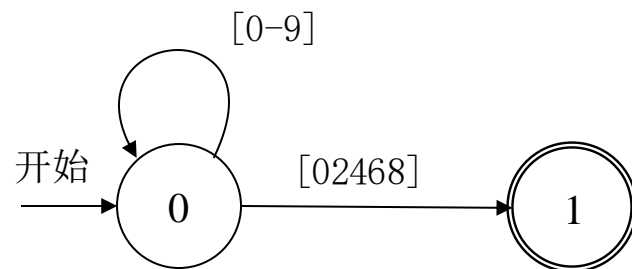
- (1) 假设允许无符号偶整数以0打头。
- (2) 假设不允许无符号偶整数以0打头。

■ 参考答案

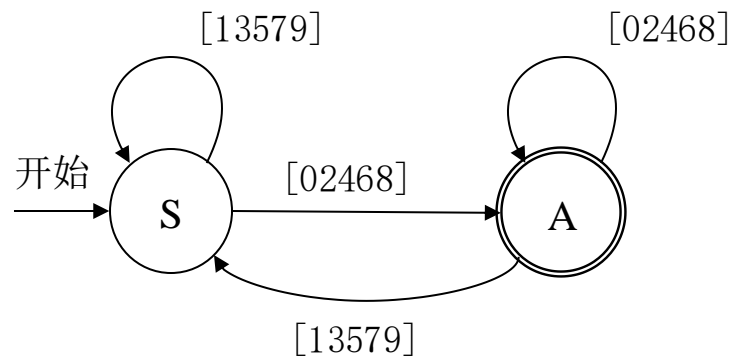
所有无符号偶整数均以0、2、4、6或者8结尾。

- (1) 若允许无符号偶整数以0打头，则相应的正则式为： $[0-9]^*(0|2|4|6|8)$

右线性文法如下：

$$\begin{aligned} S &\rightarrow 0 \mid 2 \mid 4 \mid 6 \mid 8 \\ &\quad \mid 1S \mid 3S \mid 5S \mid 7S \mid 9S \mid 0A \mid 2A \mid 4A \mid 6A \mid 8A \\ A &\rightarrow 0 \mid 2 \mid 4 \mid 6 \mid 8 \\ &\quad \mid 1S \mid 3S \mid 5S \mid 7S \mid 9S \mid 0A \mid 2A \mid 4A \mid 6A \mid 8A \end{aligned}$$


NFA



DFA

■ 习题3.4 (考察正规表达式、有限状态机、线性文法)

构造一文法，使其语言是无符号偶整数的集合。

- (1) 假设允许无符号偶整数以0打头。
- (2) 假设不允许无符号偶整数以0打头。

■ 参考答案

(2) 若不允许无符号偶整数以0打头，则相应的正则式为：

$$(\varepsilon | [1-9][0-9]^*)(0|2|4|6|8)$$

相应的右线性文法如下：

$$S \rightarrow 0 | 2 | 4 | 6 | 8 | 1B | 3B | 5B | 7B | 9B | 2C | 4C | 6C | 8C$$

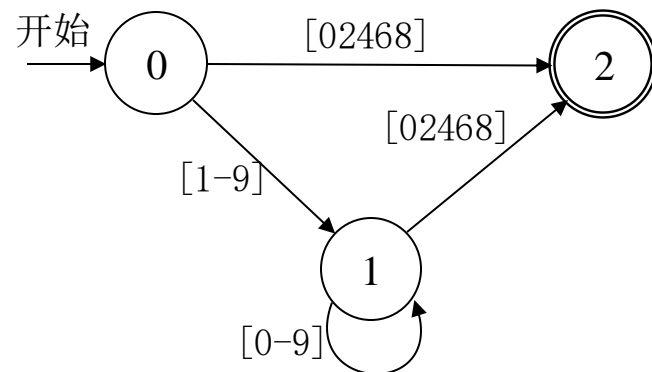
$$B \rightarrow 0 | 2 | 4 | 6 | 8 | 1B | 3B | 5B | 7B | 9B | 0C | 2C | 4C | 6C | 8C$$

$$C \rightarrow 0 | 2 | 4 | 6 | 8 | 1B | 3B | 5B | 7B | 9B | 0C | 2C | 4C | 6C | 8C$$

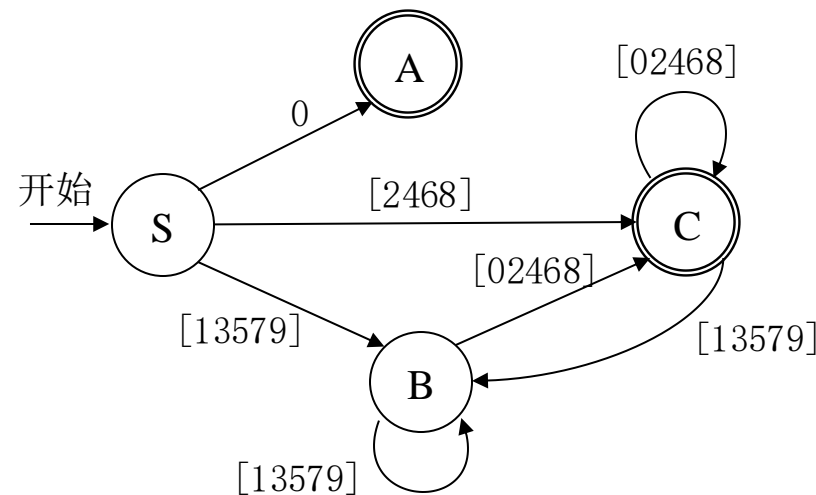
■ 易错点

✗ 状态覆盖不全，有同学定义文法：

$$S \rightarrow 1S | 2S | \dots | 9S | A, A \rightarrow 0 | 2 | 4 | 6 | 8 \quad \text{无法覆盖102这类情况}$$



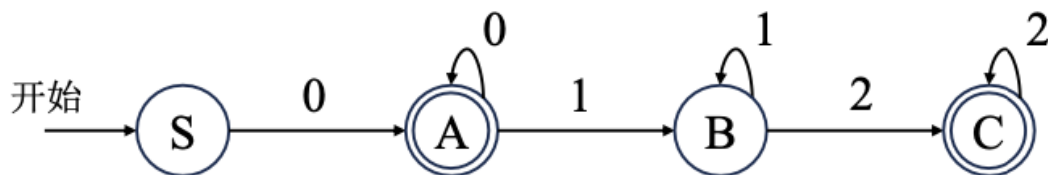
NFA



DFA

■ 期中试题1 (考察状态转换图与线性文法/正规表达式间的变换)

一、(20 分) 某自动机有如下状态转换图:



(1) (10 分) 写出与之等价的右线性文法;

(2) (10 分) 写出与之等价的正规表达式。

■ 参考答案

(1) 右线性文法:

$S \rightarrow 0A \mid 0$

$A \rightarrow 0A \mid 0 \mid 1B$

$B \rightarrow 1B \mid 2C \mid 2$

$C \rightarrow 2C \mid 2$

或者:

$S \rightarrow 0A$

$A \rightarrow 0A \mid 1B \mid \epsilon$

$B \rightarrow 1B \mid 2C$

$C \rightarrow 2C \mid \epsilon$

(2) 正规表达式:

$0^+ \mid 0^+1^+2^+$

或者:

$0^+(1^+2^+ \mid \epsilon)$

■ 易错点

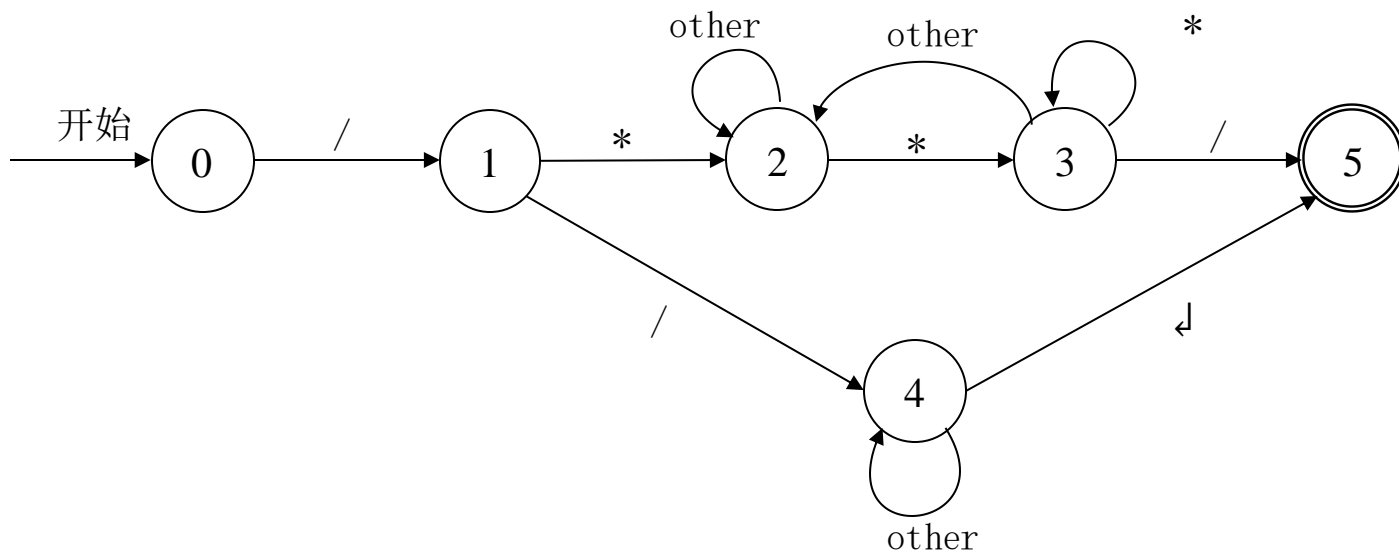
× 右线性文法缺项, 如缺 ϵ

× 正规表达式考虑不全面, $0^+1^+2^+$

■ 习题3.8 (考察DFA)

C语言规定其程序中的注释可以有单行和多行两种不同的格式，单行注释出现在行首，其格式形如 `// . . .`，多行注释格式形如 `/* . . . */`，请给出一个可以识别这两种风格的注释的DFA。

■ 参考答案



■ 易错点

- ✗ 忘记了标记起始和终止状态
- ✗ 忘记了//注释以回车结尾
- ✗ 未合并两种注释的DFA

■ 期中试题2 (考察LL(1)文法)

二、(30 分) 有如下文法G[S]:

$S \rightarrow aSA \mid a$

$A \rightarrow Ab \mid d$

- (1) (5 分) 判断该文法是否为 LL(1)文法, 若不是, 说明理由, 继续做 (2); 若是, 继续做 (3)。
- (2) (10 分) 将该文法变换为等价的 LL(1)文法G'。
- (3) (8 分) 计算文法中每个非终结符号的 FIRST 集合和 FOLLOW 集合。
- (4) (7 分) 为文法构造 LL(1)分析表。

■ 参考答案

- (1) 该文法不是 LL(1)文法。理由如下:
对于 $S \rightarrow aSA \mid a$, $first(aSA) \cap first(a) = \{a\}$, 存在左公因子。
或者:
对于 $A \rightarrow Ab \mid d$, 含有左递归。

■ 易错点

- ✗ FIRST集和FOLLOW集求解有误
- ✗ 遗漏\$列, 多了ε列

(2) 提取左公因子, 消除左递归:

$S \rightarrow aS'$

$S' \rightarrow SA \mid \epsilon$

$A \rightarrow dA'$

$A' \rightarrow bA' \mid \epsilon$

(3) 文法中每个非终结符号的 FIRST 集合和 FOLLOW 集合:

	FIRST	FOLLOW
S	a	\$, d
S'	a, ε	\$, d
A	d	\$, d
A'	b, ε	\$, d

(4) 文法的 LL(1)分析表:

	a	b	d	\$
S	$S \rightarrow aS'$			
S'	$S' \rightarrow SA$		$S' \rightarrow \epsilon$	$S' \rightarrow \epsilon$
A			$A \rightarrow dA'$	
A'		$A' \rightarrow bA'$	$A' \rightarrow \epsilon$	$A' \rightarrow \epsilon$

■ 习题4.9 (考察LR文法判别)

考虑如下文法 G :

$$S \rightarrow AS|b$$
$$A \rightarrow SA|a$$

- (1) 构造该文法的LR(0)项目集规范族及识别其所有活前缀的DFA。
- (2) 该文法是SLR(1)文法吗? 为什么?
- (3) 构造该文法的LR(1)项目集规范族, 该文法是LR(1)文法吗?

■ 参考答案

(1) 首先拓广文法,

(0) $S' \rightarrow S$ (1) $S \rightarrow AS$ (2) $S \rightarrow b$ (3) $A \rightarrow SA$ (4) $A \rightarrow a$

(2) 不是SLR(1)文法, $\text{FOLLOW}(A)=\{a,b\}$, 状态 I_5 中存在的移进-规约冲突无法通过向前看一个输入符号解决。

■ 关键点

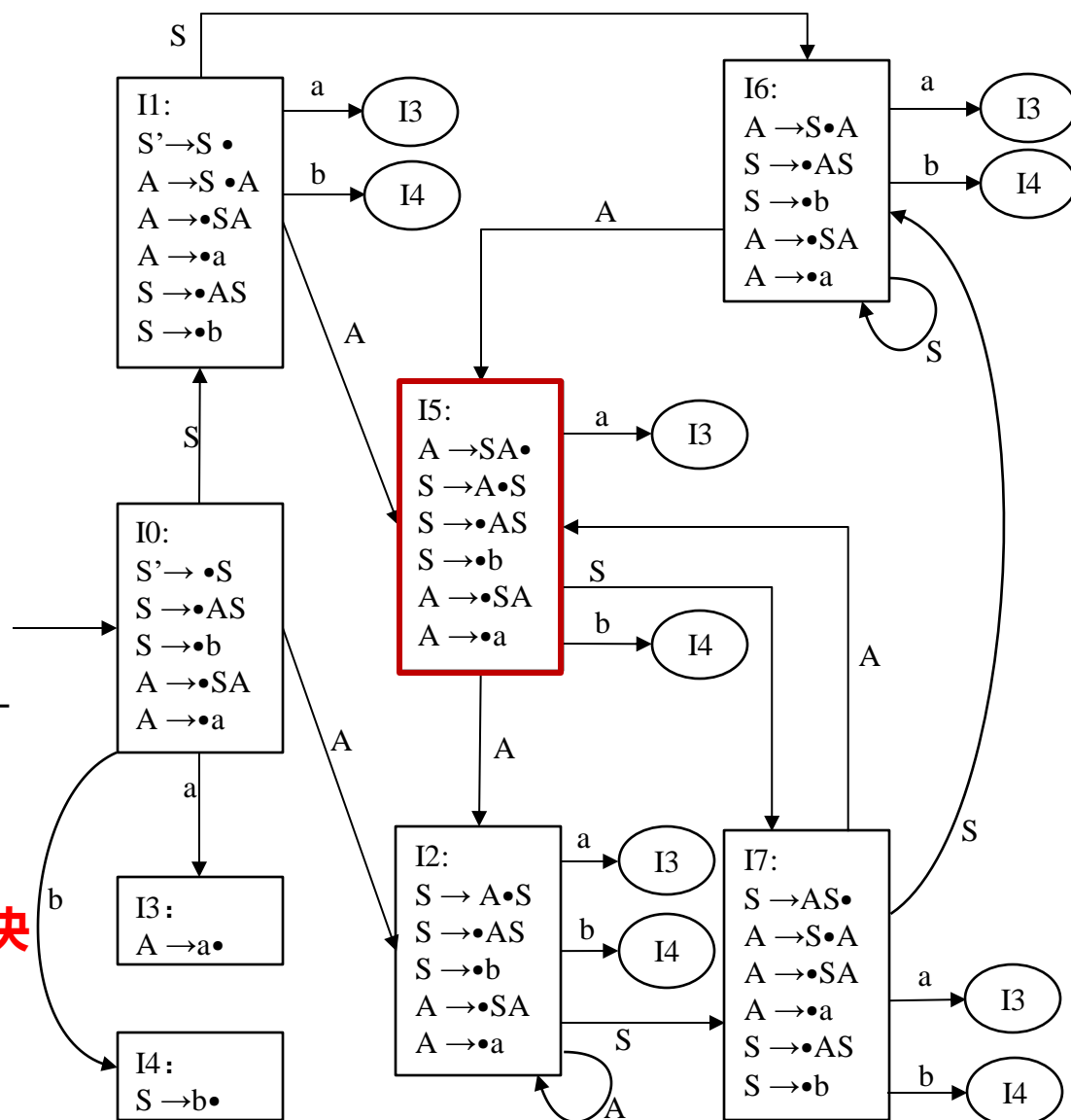
✓ SLR(1)文法判别: **移进-归约冲突, 不能通过向前看符号解决**

■ 易错点

✗ 漏了 \bullet 号后面非终结符的产生式

✗ 少“不能通过向前看一个输入符号来解决”这句话

LR(0)项目集规范族及识别所有活前缀的DFA



■ 习题4.9 (考察LR文法判别)

考虑如下文法G:

$$S \rightarrow AS|b$$
$$A \rightarrow SA|a$$

- (1) 构造该文法的LR(0)项目集规范族及识别其所有活前缀的DFA。
- (2) 该文法是SLR(1)文法吗? 为什么?
- (3) 构造该文法的LR(1)项目集规范族, 该文法是LR(1)文法吗?

■ 参考答案

(3) 首先, 构造该文法的LR(1)项目集规范族及识别它所有活前缀的DFA。在状态I₅中, 项目 $[A \rightarrow SA \bullet, a/b]$ 和 $[A \rightarrow \bullet a, a/b]$ 存在移进-归约冲突, 所以不是LR(1)文法。

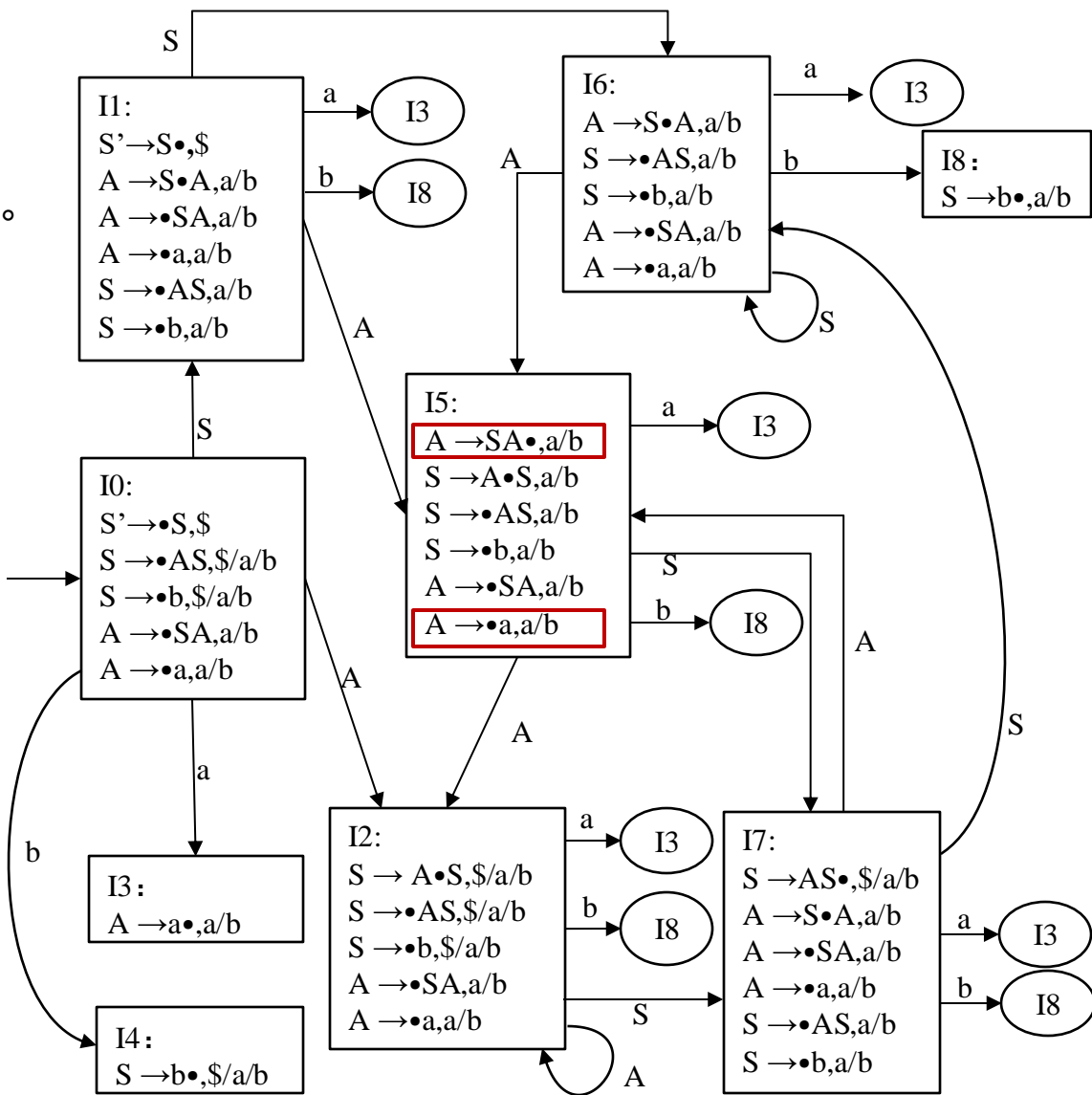
■ 关键点

✓ LR(1)向前看1个符号, LR(0)不看, SLR(1)只需要时看

■ 易错点

✗ 向前看符号

LR(1)项目集规范族及识别所有活前缀的DFA



■ 习题4.16 (考察LR文法判别)

下面的文法属于哪一类LR文法？试构造其分析表。

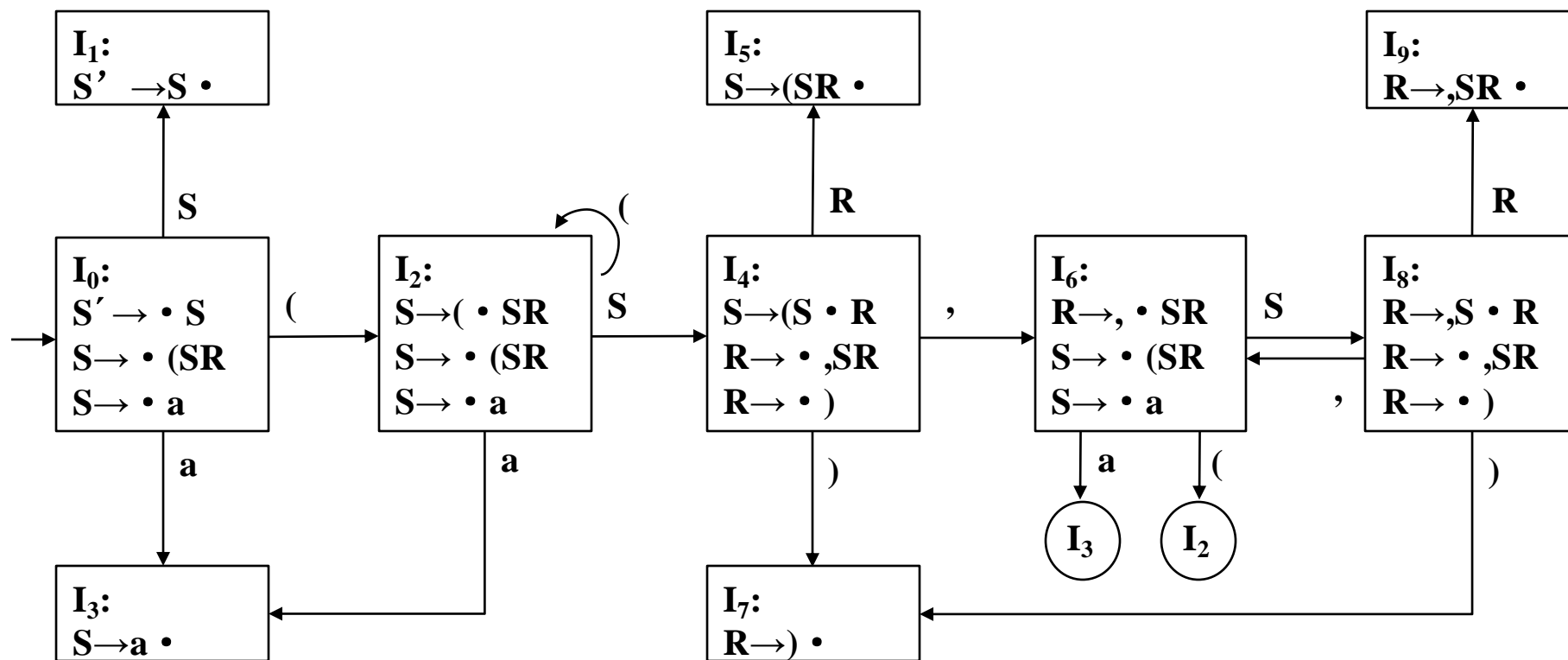
$S \rightarrow (SR|a$

$R \rightarrow , SR|)$

■ 参考答案

该文法拓广为: (0) $S' \rightarrow S$ (1) $S \rightarrow (SR$ (2) $S \rightarrow a$ (3) $R \rightarrow ,SR$ (4) $R \rightarrow)$

其次，构造文法的LR(0)项目集规范族及识别其所有活前缀的DFA



■ 习题4.16 (考察LR文法判别)

下面的文法属于哪一类LR文法？试构造其分析表。

$S \rightarrow (SR|a$

$R \rightarrow , SR|)$

■ 参考答案

LR(0)有效项目集中，要么只含有一个归约项目（如 I_1 、 I_3 、 I_5 、 I_7 和 I_9 ），要么只含有移进或待约项目（如 I_0 、 I_2 、 I_4 、 I_6 和 I_8 ），所以该文法是LR(0)文法，同时也是SLR(1)文法、LR(1)文法和LALR(1)文法。

■ 易错点

- ✗ LR文法判断依据错误，LR文法间关系弄错
- ✗ action表多了ε列，goto表多了S'列
- ✗ 状态标记错误，归约标记可以填到全部action列

LR(0)分析表

状态	action					goto	
	a	,	()	\$	S	R
0	S3		S2			1	
1					ACC		
2	S3		S2			4	
3	R2						
4		S6		S7			5
5	R1						
6	S3		S2			8	
7	R4						
8		S6		S7			9
9	R3						

■ 期中试题3 (考察LR分析方法)

三、(50 分) 有如下文法G[S]:

$S \rightarrow M = N \mid N$

$N \rightarrow M$

$M \rightarrow * N \mid id$

(1) (4 分) 给出该文法的拓广文法。

(2) (10 分) 构造文法G的 LR(1)项目集规范族及识别其所有活前缀的 DFA。

(3) (10 分) 判断该文法是否为 LR(1)文法, 简述理由。若是, 继续做 (4)。

(4) (10 分) 构造该文法的 LR(1)分析表。

(5) (8 分) 判断该文法是否为 SLR(1)文法, 简述理由。

(6) (8 分) 判断该文法是否为 LALR(1)文法, 简述理由。

■ 参考答案

(2) 构造文法G的LR(1)项目集规范族及识别其所有活前缀的DFA

(1) 拓广文法:

(0) $S' \rightarrow S$

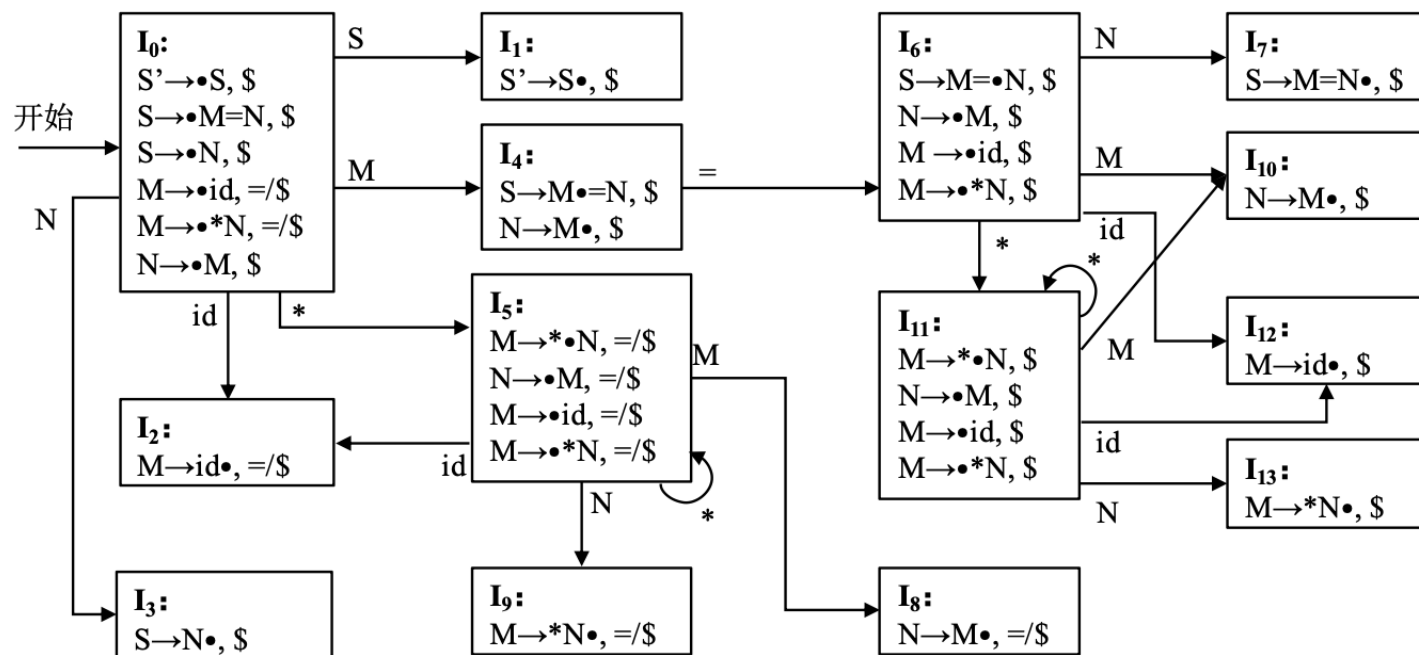
(1) $S \rightarrow M = N$

(2) $S \rightarrow N$

(3) $N \rightarrow M$

(4) $M \rightarrow * N$

(5) $M \rightarrow id$



■ 期中试题3 (考察LR分析方法)

- 三、(50 分) 有如下文法 $G[S]$:
- $$S \rightarrow M = N \mid N$$
- $$N \rightarrow M$$
- $$M \rightarrow * N \mid id$$
- (1) (4 分) 给出该文法的拓广文法。
- (2) (10 分) 构造文法 G 的 LR(1) 项目集规范族及识别其所有活前缀的 DFA。
- (3) (10 分) 判断该文法是否为 LR(1) 文法, 简述理由。若是, 继续做 (4)。
- (4) (10 分) 构造该文法的 LR(1) 分析表。
- (5) (8 分) 判断该文法是否为 SLR(1) 文法, 简述理由。
- (6) (8 分) 判断该文法是否为 LALR(1) 文法, 简述理由。

■ 参考答案

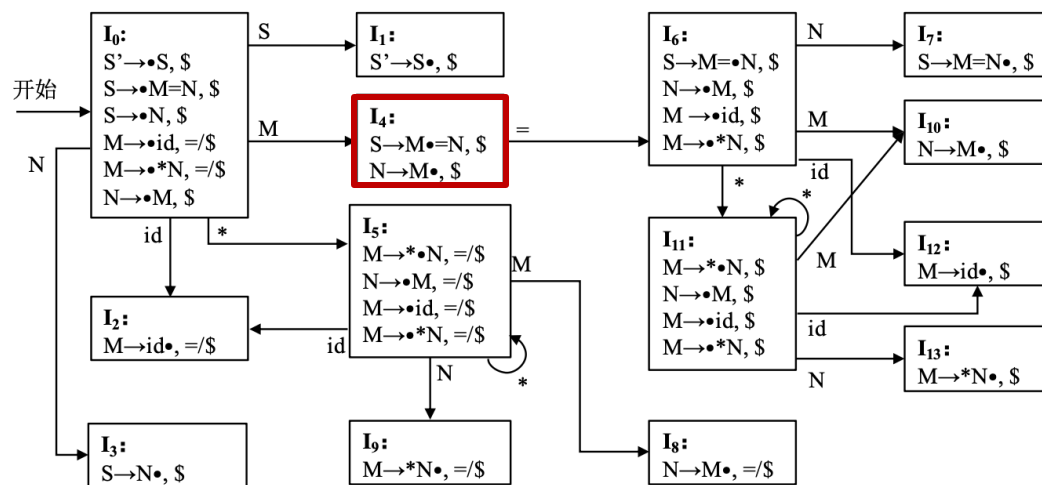
- (3) 判断该文法是否为 LR(1) 文法, 简述理由。若是, 继续做 (4)。

该文法是 LR(1) 文法, 理由如下:

项目集 I_0 、 I_5 、 I_6 、 I_{11} 只含有移进项目和待约项目, 不存在冲突;

项目集 I_1 、 I_2 、 I_3 、 I_7 、 I_8 、 I_9 、 I_{10} 、 I_{12} 、 I_{13} 都只含有一个归约项目, 不存在冲突;

只有项目集 I_4 中同时含有移进项目和归约项目, 但是移进符号和归约符号没有交集, 所以不存在冲突。



■ 期中试题3（考察LR分析方法）

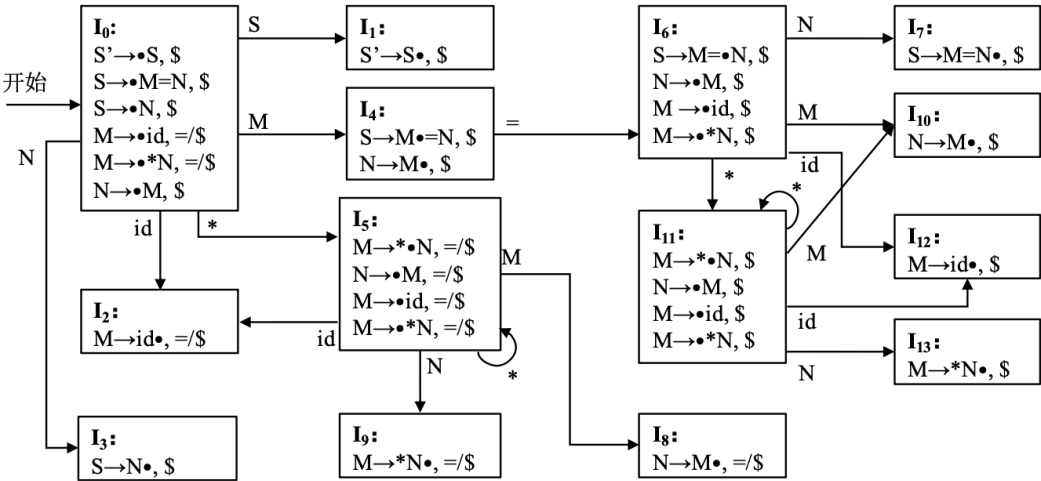
三、（50 分）有如下文法G[S]：

$S \rightarrow M = N \mid N$
 $N \rightarrow M$
 $M \rightarrow * N \mid id$

- (1)（4 分）给出该文法的拓广文法。
- (2)（10 分）构造文法G的 LR(1)项目集规范族及识别其所有活前缀的 DFA。
- (3)（10 分）判断该文法是否为 LR(1)文法，简述理由。若是，继续做（4）。
- (4)（10 分）构造该文法的 LR(1)分析表。
- (5)（8 分）判断该文法是否为 SLR(1)文法，简述理由。
- (6)（8 分）判断该文法是否为 LALR(1)文法，简述理由。

■ 参考答案

(4) 构造该文法的 LR(1)分析表。



(4) 构造该文法的 LR(1)分析表。

状态	action				goto		
	id	*	=	\$	S	M	N
0	S2	S5			1	4	3
1				ACC			
2			R5	R5			
3				R2			
4			S6	R3			
5	S2	S5				8	9
6	S12	S11				10	7
7				R1			
8			R3	R3			
9			R4	R4			
10				R3			
11	S12	S11				10	13
12				R5			
13				R4			

■ 期中试题3 (考察LR分析方法)

- 三、(50 分) 有如下文法 $G[S]$:
- $$S \rightarrow M = N \mid N$$
- $$N \rightarrow M$$
- $$M \rightarrow * N \mid id$$
- (1) (4 分) 给出该文法的拓广文法。
- (2) (10 分) 构造文法 G 的 LR(1) 项目集规范族及识别其所有活前缀的 DFA。
- (3) (10 分) 判断该文法是否为 LR(1) 文法, 简述理由。若是, 继续做 (4)。
- (4) (10 分) 构造该文法的 LR(1) 分析表。
- (5) (8 分) 判断该文法是否为 SLR(1) 文法, 简述理由。
- (6) (8 分) 判断该文法是否为 LALR(1) 文法, 简述理由。

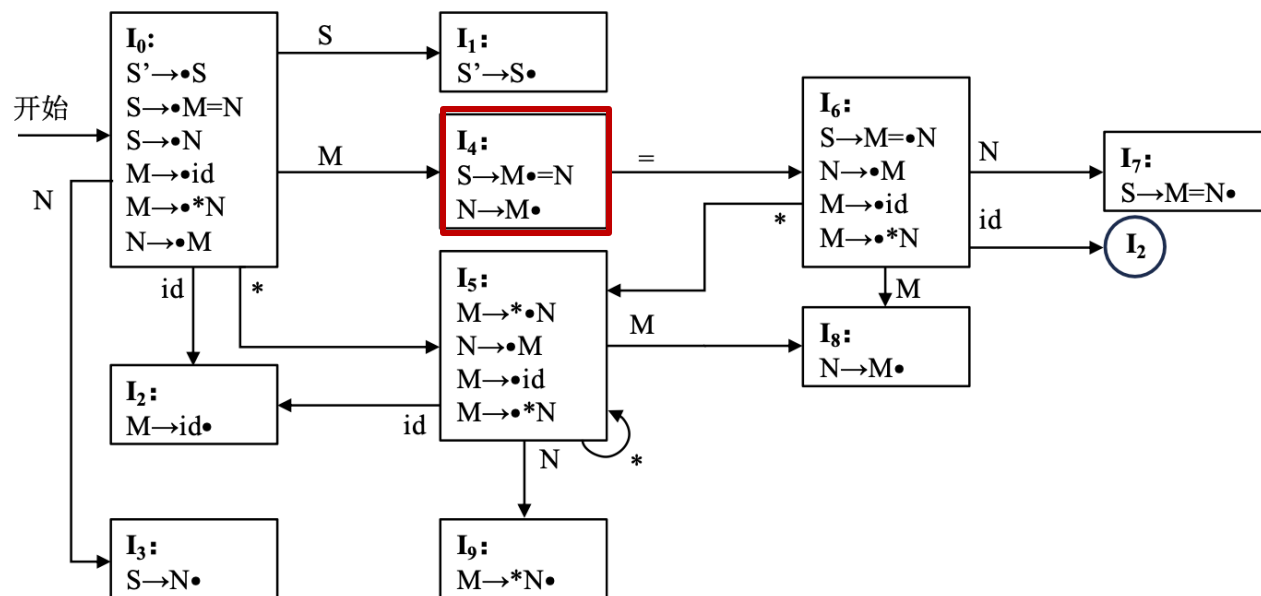
■ 参考答案

- (5) 判断该文法是否为 SLR(1) 文法, 简述理由。

该文法不是 SLR(1) 文法。

理由如下:

构造 LR(0) 项目集规范族及识别其所有活前缀的 DFA:



项目集 I_4 中同时含有移进-归约冲突, 其它项目集中不存在冲突。

$FOLLOW(N) = \{=, \$\}$, 在面临 '=' 号时无法解归约和移进的冲突动作, 所以该文法不是 SLR(1) 文法。

■ 期中试题3 (考察LR分析方法)

- 三、(50 分) 有如下文法G[S]:
- $$S \rightarrow M = N \mid N$$
- $$N \rightarrow M$$
- $$M \rightarrow * N \mid id$$
- (1) (4 分) 给出该文法的拓广文法。
- (2) (10 分) 构造文法G的 LR(1)项目集规范族及识别其所有活前缀的 DFA。
- (3) (10 分) 判断该文法是否为 LR(1)文法, 简述理由。若是, 继续做 (4)。
- (4) (10 分) 构造该文法的 LR(1)分析表。
- (5) (8 分) 判断该文法是否为 SLR(1)文法, 简述理由。
- (6) (8 分) 判断该文法是否为 LALR(1)文法, 简述理由。

■ 参考答案

- (6) 判断该文法是否为 LALR(1)文法, 简述理由。

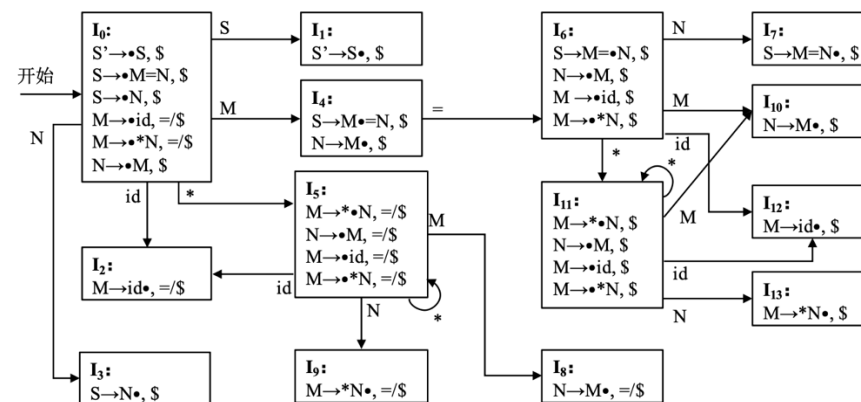
该文法是 LALR(1)文法。

理由如下:

首先该文法是 LR(1)文法, 合并同心集 I_8 和 I_{10} 、 I_9 和 I_{13} 、 I_2 和 I_{12} 以及 I_5 和 I_{11} 后不会产生冲突, 所以相应的文法是 LALR(1)文法。

■ 关键点 (几类LR文法概念及判定方法)

- ✓ 能力关系: $LR(0) < SLR(1) < LALR(1) < LR(1)$
- ✓ LR(0): 无冲突
- ✓ SLR(1): LR(0)有冲突, 但冲突时能通过向前看一个符号解决
- ✓ LR(1): 无冲突, 相比SLR(1)持续向前看
- ✓ LALR(1): LR(1)基础上, 无同心集, 或合并同心集后无冲突



- ✓ 不需要消除左递归
- ✓ 构造LR(1)项目集时, 注意LR(1)项目的向前看符号的完整性, 检查所有待约项目, 直到集合不再变化为止。

■ 习题5.10 (考察S和L属性定义、自底向上语法制导翻译)

考虑如下的语法制导定义：

产生式	语义规则
$S \rightarrow B$	$B.ps = 10$ $S.ht = B.ht$
$B \rightarrow B_1 B_2$	$B_1.ps = B.ps$ $B_2.ps = B.ps$ $B.ht = \max(B_1.ht, B_2.ht)$
$B \rightarrow B_1 \text{sub} B_2$	$B_1.ps = B.ps$ $B_2.ps = \text{shrink}(B.ps)$ $B.ht = \text{disp}(B_1.ht, B_2.ht)$
$B \rightarrow \text{text}$	$B.ht = \text{test.h} \times B.ps$

- (1) 判断该语法制导语义是否为L属性定义
- (2) 给出语法制导定义相应的翻译方案
- (3) 改造(2)所得翻译方案，使之可用LR方法进行翻译
- (4) 根据(3)所得翻译方案，设计与各产生式相应的代码段
- (5) 根据(4)所设计代码段，举例说明，每当把一个右部规约为B时，继承属性B.ps的值在栈中的位置总是恰好在规约串的下面

■ 参考答案

(1) 注意L属性定义要点即可。B.ps要么取值常数，要么依赖于产生式左部符号B的属性。

(2) 翻译方案：

```
S → { B.ps = 10 } B { S.ht = B.ht }
B → { B1.ps = B.ps } B1 { B2.ps = B.ps } B2 { B.ht = max(B1.ht, B2.ht) }
B → { B1.ps = B.ps } B1 sub { B2.ps = shrink(B.ps) } B2 { B.ht = disp(B1.ht, B2.ht) }
B → text { B.ht = test.h × B.ps }
```

■ 习题5.10 (考察S和L属性定义、自底向上语法制导翻译)

考虑如下的语法制导定义：

产生式	语义规则
$S \rightarrow B$	$B.ps = 10$ $S.ht = B.ht$
$B \rightarrow B_1 B_2$	$B_1.ps = B.ps$ $B_2.ps = B.ps$ $B.ht = \max(B_1.ht, B_2.ht)$
$B \rightarrow B_1 \text{sub} B_2$	$B_1.ps = B.ps$ $B_2.ps = \text{shrink}(B.ps)$ $B.ht = \text{disp}(B_1.ht, B_2.ht)$
$B \rightarrow \text{text}$	$B.ht = \text{test.h} \times B.ps$

- (1) 判断该语法制导语义是否为L属性定义
- (2) 给出语法制导定义相应的翻译方案
- (3) 改造 (2) 所得翻译方案，使之可用LR方法进行翻译
- (4) 根据 (3) 所得翻译方案，设计与各产生式相应的代码段
- (5) 根据 (4) 所设计代码段，举例说明，每当把一个右部规约为B时，继承属性B. ps的值在栈中的位置总是恰好在规约串的下面

■ 参考答案

(3) 改造要点：去掉产生式中间的语义动作
引入标记非终结符L\M\N来实现，标记非终结符代表了要执行某个语义动作。

产生式	语义规则
$S \rightarrow LB$	$B.ps = L.s$ $S.ht = B.ht$
$L \rightarrow \epsilon$	$L.s = 10$
$B \rightarrow B_1 M B_2$	$B_1.ps = B.ps$ $M.i = B.ps$ $B_2.ps = M.s$ $B.ht = \text{disp}(B_1.ht, B_2.ht)$
$B \rightarrow B_1 \text{sub} N B_2$	$B_1.ps = B.ps$ $N.i = B.ps$ $B_2.ps = N.s$ $B.ht = \text{disp}(B_1.ht, B_2.ht)$
$B \rightarrow \text{text}$	$B.ht = \text{test.h} \times B.ps$
$M \rightarrow \epsilon$	$M.s = M.i$
$N \rightarrow \epsilon$	$N.s = \text{shrink}(N.i)$

■ 习题5.10 (考察S和L属性定义、自底向上语法制导翻译)

考虑如下的语法制导定义：

产生式	语义规则
$S \rightarrow B$	$B.ps = 10$ $S.ht = B.ht$
$B \rightarrow B_1 B_2$	$B_1.ps = B.ps$ $B_2.ps = B.ps$ $B.ht = \max(B_1.ht, B_2.ht)$
$B \rightarrow B_1 \text{sub} B_2$	$B_1.ps = B.ps$ $B_2.ps = \text{shrink}(B.ps)$ $B.ht = \text{disp}(B_1.ht, B_2.ht)$
$B \rightarrow \text{text}$	$B.ht = \text{test.h} \times B.ps$

- (1) 判断该语法制导语义是否为L属性定义
- (2) 给出语法制导定义相应的翻译方案
- (3) 改造 (2) 所得翻译方案，使之可用LR方法进行翻译
- (4) 根据 (3) 所得翻译方案，设计与各产生式相应的代码段
- (5) 根据 (4) 所设计代码段，举例说明，每当把一个右部规约为B时，继承属性B. ps的值在栈中的位置总是恰好在规约串的下面

■ 参考答案

(4) 根据LR翻译方案，设计各产生式的代码段

由于上述翻译方案中的所有继承属性都由复制规则赋值，所以语法制导定义的实现都是通过跟踪它们在val栈中的位置来获得属性值的。

■ 易错点

✗ 指针位置标注错误

产生式	实现语义规则的代码
$S \rightarrow LB$	$\text{val}[\text{ntop}] = \text{val}[\text{top}]$
$L \rightarrow \varepsilon$	$\text{val}[\text{ntop}] = 10$
$B \rightarrow B_1 M B_2$	$\text{val}[\text{ntop}] = \max(\text{val}[\text{top}-2], \text{val}[\text{top}])$
$B \rightarrow B_1 \text{sub} N B_2$	$\text{val}[\text{ntop}] = \text{disp}(\text{val}[\text{top}-3], \text{val}[\text{top}])$
$B \rightarrow \text{text}$	$\text{val}[\text{ntop}] = \text{val}[\text{top}] \times \text{val}[\text{top}-1]$
$M \rightarrow \varepsilon$	$\text{val}[\text{ntop}] = \text{val}[\text{top}-1]$
$N \rightarrow \varepsilon$	$\text{val}[\text{ntop}] = \text{shrink}(\text{val}[\text{top}-2])$

■ 习题5.10 (考察S和L属性定义、自底向上语法制导翻译)

(5) 举例说明，每当把一个右部规约为B时，继承属性B.ps的值在栈中的位置总是恰好在规约串的下面

假定有输入符号串 E sub 2.s，并且假定: ①词法分析识别出 E、sub、2、.、和 s 的属性 h 的值均为 1; ② shrink(x)=0.3*x, disp(x,y)=x+y; ③为说明方便，用文法符号代替与之对应的状态。则其分析过程如表 5 所示。

步骤	栈	输入	分析动作
(1)	state: Val:	Esub2.s\$	归约 $L \rightarrow \epsilon$ $val[ntop]=10$
(2)	state:L Val:10	Esub2.s\$	移进
(3)	state:L E Val:10 1	sub2.s\$	归约 $B \rightarrow text$ $val[ntop]=val[top] \times val[top-1]$
(4)	state:L B Val:10 10	sub2.s\$	移进
(5)	state:L B sub Val:10 10 -	2.s\$	归约 $N \rightarrow \epsilon$ $val[ntop]=shrink(val[top-2])$
(6)	state:L B sub N Val:10 10 -3	2.s\$	移进
(7)	state:L B sub N 2 Val:10 10 -3 1	.s\$	归约 $B \rightarrow text$ $val[ntop] = val[top] \times val[top-1]$
(8)	state:L B sub N B Val:10 10 -3 3	.s\$	归约 $B \rightarrow BsubNB$ $val[ntop]=disp(val[top-3],val[top])$
(9)	state:L B Val:10 13	.s\$	归约 $M \rightarrow \epsilon$ $val[ntop]=val[top-1]$

(10)	state: L B M Val:10 13 10	.s\$	移进
(11)	state: L B M Val:10 13 10 1	s\$	归约 $B \rightarrow text$ $val[ntop]=val[top] \times val[top-1]$
(12)	state: L B M B Val:10 13 10 10	s\$	归约 $B \rightarrow BMB$ $val[ntop]=max(val[top-2],val[top])$
(13)	state:L B Val:10 13	s\$	归约 $M \rightarrow \epsilon$ $val[ntop]=val[top-1]$
(14)	state:L B M Val:10 13 10	s\$	移进
(15)	state: L B M S Val:10 13 10 1	\$	归约 $B \rightarrow text$ $val[ntop]=val[top] \times val[top-1]$
(16)	state: L B M B Val:10 13 10 10	\$	归约 $B \rightarrow BMB$ $val[ntop]=max(val[top-2],val[top])$
(17)	state: L B Val:10 13	\$	归约 $S \rightarrow LB$ $val[ntop]=val[top]$
(18)	state: S Val:13	\$	接受

■ 易错点

× 分析动作中没有给出代码段，举例不充分

■ 习题6.7 (考察名字等价和结构等价)

有如下的C语言声明:

```
typedef struct{
    int a,b;
} CELL, *PCELL;
CELL foo[100];
PCELL bar(int x, CELL y){...}
```

写出变量 *foo* 和 *bar* 的类型表达式

■ 参考答案

CELL 的类型表达式: `record((a x integer)x(b x integer))`

PCELL 的类型表达式: `pointer(record((a x integer)x(b x integer)))`

foo 的类型表达式: `array(0..99,CELL)`

bar 的类型表达式: `intxCELL→PCELL`

■ 关键点

- ✓ 类型表达式形式化写法
- ✓ C语言类型表达式等价类型: C语言struct和union是名字等价

■ 易错点

× 把CELL, PCELL展开了, 破坏了名字等价

✓ 先说明CELL的类型表达式, 再引用

■ 习题7.5 (考察变量作用域、控制栈)

考虑下面的Pascal程序

```
(1) program main(input, output);
(2)   procedure b(function h(n: integer): integer);
(3)     var m: integer;
(4)     begin m:=3;writeln(h(2)) end; {end of b}
(5)   procedure c;
(6)     var m: integer;
(7)     function f(n: integer): integer;
(8)       begin f:=m+n end;    {end of f}
(9)     procedure r;
(10)      var m: integer;
(11)      begin m:=7;b(f) end; {end of r}
(12)    begin m:=0; r end;    {end of c}
(13)  begin c end.{end of main}
```

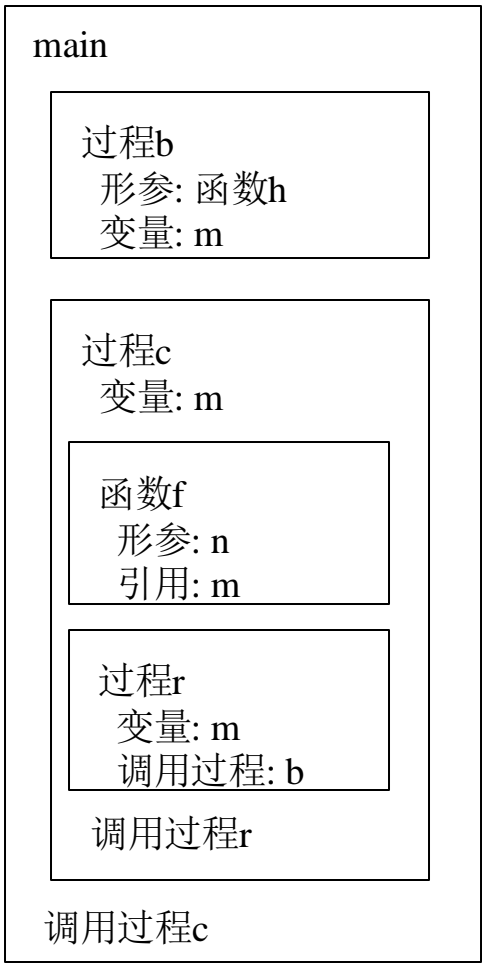
- (1) 该程序的输出结果是什么?
- (2) 试画出该程序的活动树。
- (3) 试画出当控制处于函数f中时的控制栈状态, 要求标出其中的控制链和访问链。

■ 关键点

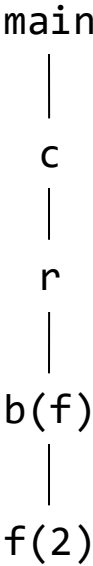
- ✓ 输出结果: 变量作用域
- ✓ 尽可能给出计算过程
- ✓ 函数f作为参数要自带访问链

■ 参考答案

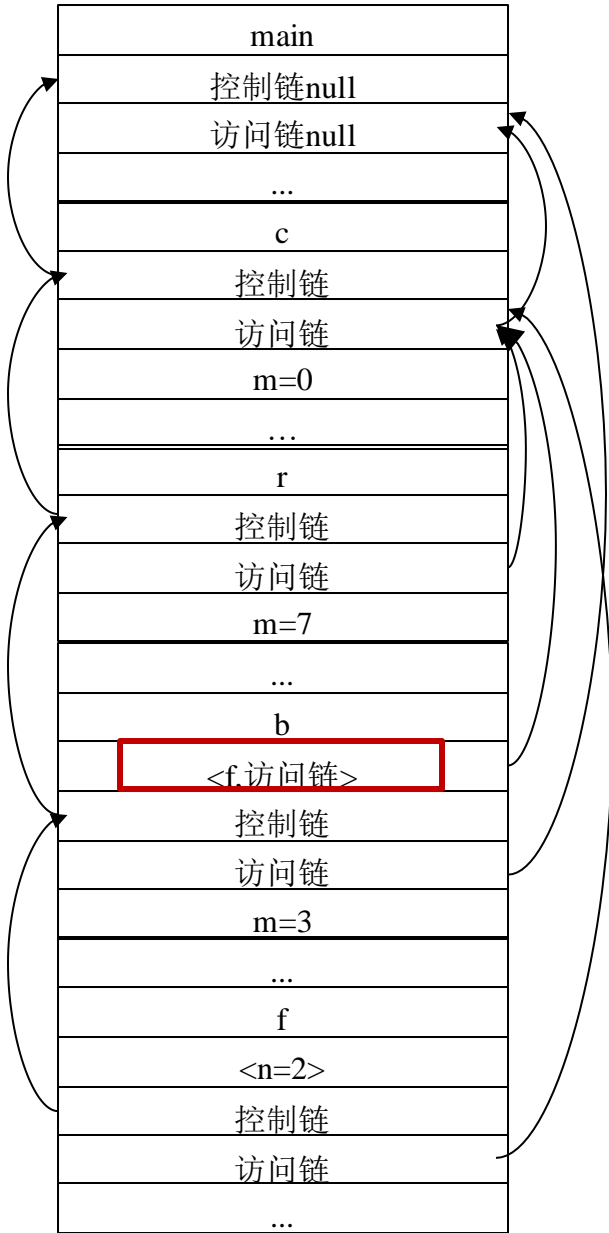
(1) 过程之间静态嵌套及引用关系如下, 结果为2。



(2)



(3)



■ 习题10.4 (考察三地址代码、基本块与流图、中间代码优化)

有如下程序段：

```
var a,b: array[1..m, 1..n] of integer;  
    i,j: integer;  
for i:=1 to m do  
    for j:=1 to n do  
        a[i,j]:=b[i,j]
```

(1) 请把该程序段中的可执行语句翻译为三地址代码。

(2) 将 (1) 的结果划分为基本块，并画出其流图。

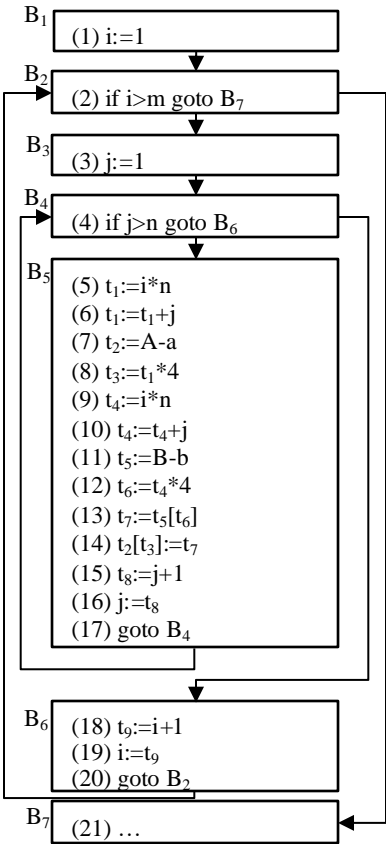
(3) 对内循环代码进行所有可能的优化，要求依次写出所采用的优化技术、关键步骤，给出每种优化后的结果。

■ 参考答案

(1) 三地址代码

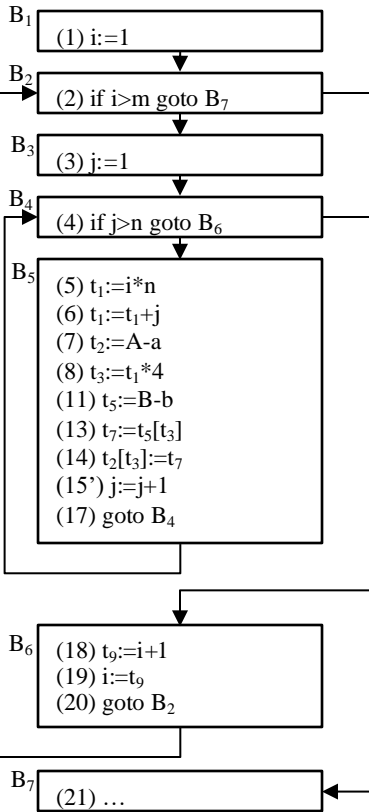
- (1) i:=1
- (2) if i>m goto (21)
- (3) j:=1
- (4) if j>n goto (18)
- (5) t₁:=i*n
- (6) t₁:=t₁+j
- (7) t₂:=A-a
- (8) t₃:=t₁*4
- (9) t₄:=i*n
- (10) t₄:=t₄+j
- (11) t₅:=B-b
- (12) t₆:=t₄*4
- (13) t₇:=t₅[t₆]
- (14) t₂[t₃]:=t₇
- (15) t₈:=j+1
- (16) j:=t₈
- (17) goto (4)
- (18) t₉:=i+1
- (19) i:=t₉
- (20) goto (2)
- (21) ...

(2) 基本块及流图

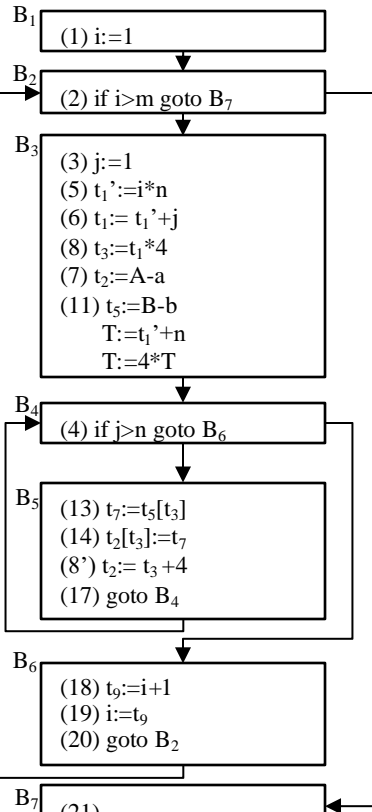


(3) 代码优化

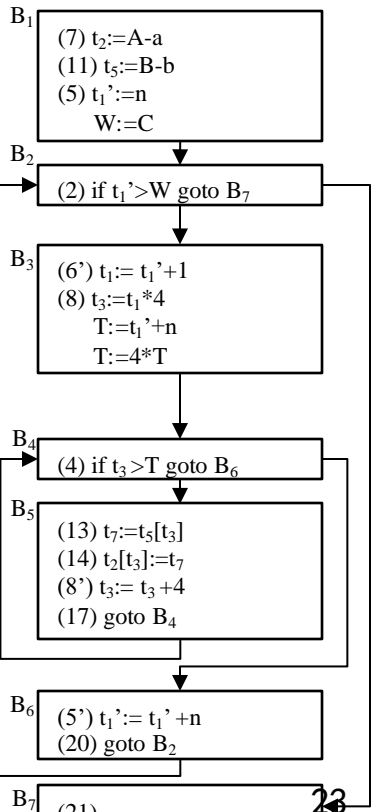
基本块优化：
删除公共子
表达式，
复制传播，
删除死代码



内循环优化：
代码外提
削减计算强度
删除归纳变量



外循环优化：
削减计算强度
删除归纳变量



■ 习题10.4 (考察三地址代码、基本块与流图、中间代码优化)

有如下程序段：

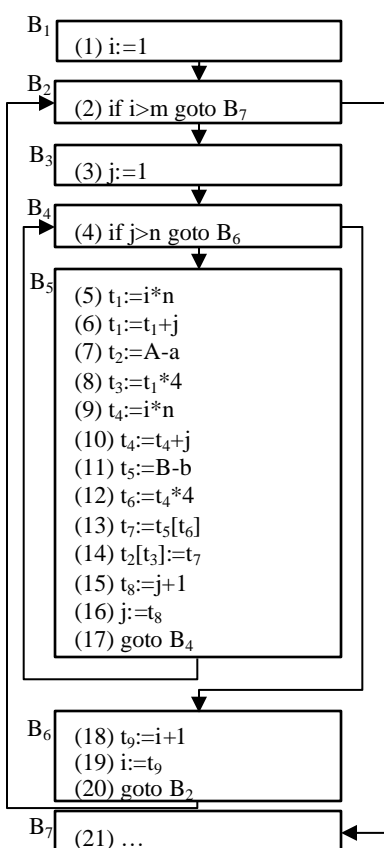
```
var a,b: array[1..m, 1..n] of integer;  
    i,j: integer;  
for i:=1 to m do  
    for j:=1 to n do  
        a[i,j]:=b[i,j]
```

■ 参考答案

(1) 三地址代码

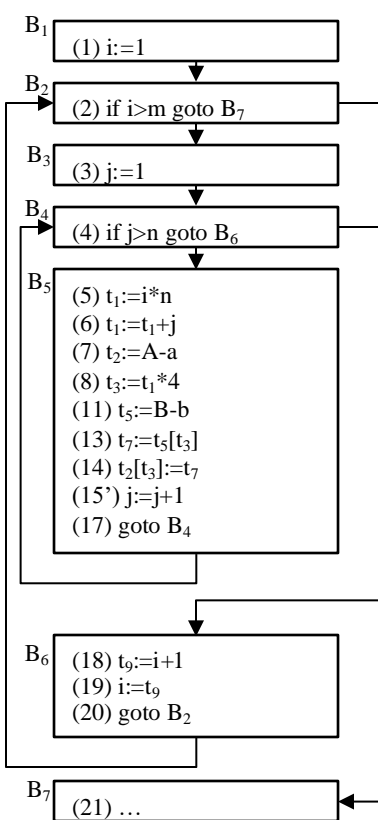
```
(1) i:=1  
(2) if i>m goto (21)  
(3) j:=1  
(4) if j>n goto (18)  
(5) t1:=i*n  
(6) t1:=t1+j  
(7) t2:=A-a  
(8) t3:=t1*4  
(9) t4:=i*n  
(10) t4:=t4+j  
(11) t5:=B-b  
(12) t6:=t4*4  
(13) t7:=t5[t6]  
(14) t2[t3]:=t7  
(15) t8:=j+1  
(16) j:=t8  
(17) goto (4)  
(18) t9:=i+1  
(19) i:=t9  
(20) goto (2)  
(21) ...
```

(2) 基本块及流图

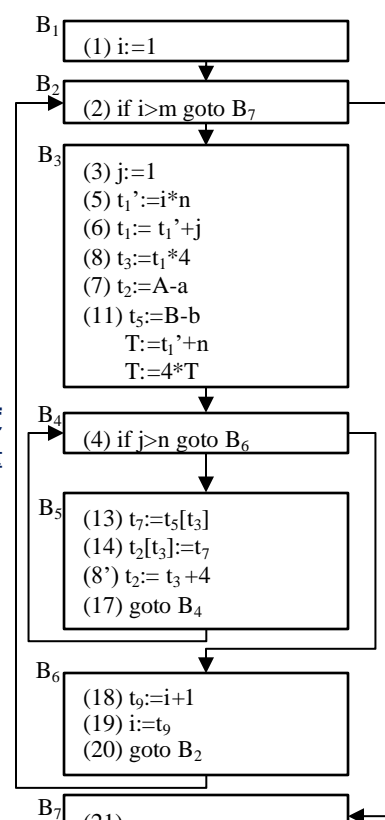


(3) 代码优化

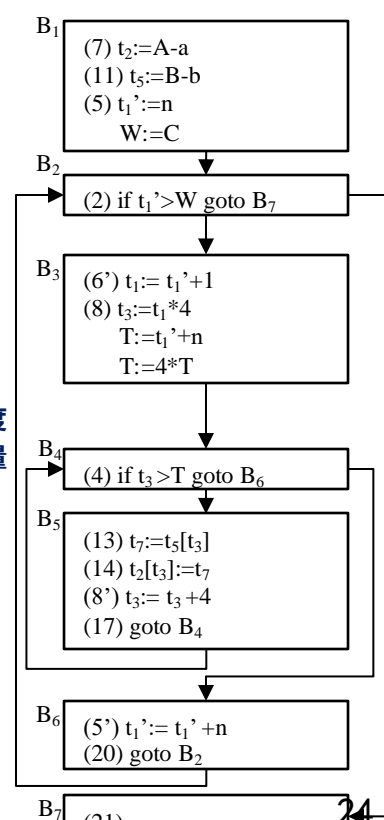
基本块优化：
删除公共子
表达式，
复制传播，
删除死代码



内循环优化：
代码外提
削减计算强度
删除归纳变量



外循环优化：
削减计算强度
删除归纳变量



■ 易错点

- ✗ 漏掉了连接或连接标错
- ✗ 未对内循环进行所有可能的优化。

■ 关键点

- ✓ 基本块优化技术、循环优化技术
- ✓ 可能需要多次重复应用，每次优化可能又出现了新的优化机会

课程安排

1. 习题易错点分析
2. 重要知识点总结

重要知识点总结

□ 词法分析

- 正规表达式、NFA、DFA、右线性文法
- DFA化简方法

□ 语法分析 (难点)

- 自顶向下分析方法
 - First集和Follow集构建
 - LL(1)文法判别: 1) 基于First集和Follow集; 2) 分析表无冲突项
 - LL(1)分析表构建、及基于分析表的分析过程
- 自底向上分析方法
 - $LR(0) < SLR(1) < LALR(1) < LR(1)$, 文法概念及能力关系
 - 各类文法的判别方法
 - SLR(1): 无冲突或冲突能够通过向前看一个符号来解决
 - LR项目集规范族及识别活前缀的DFA构建方法
 - LR(0)与SLR(1)的DFA构建方法
 - LR(1)与LALR(1)的DFA构建方法, 重点是向前看符号的确定

□ 语法制导翻译 (难点)

- 注释分析树
- 语法制导定义 (产生式语义规则) VS 翻译方案
- L属性定义和S属性定义
- 引入标记非终结符改造为LR方法翻译
- 翻译方案下, 产生式相应的代码段

□ 语义分析

- 语义分析概念、主要任务
- 符号表概念、符号表上操作
- 类型表达式、等价规则

□ 运行环境

- 不同参数传递机制
- 程序活动时的控制栈状态 (控制链和访问链)

□ 中间代码生成/优化

- 三地址代码、基本块与流图、中间代码优化

祝同学们期末收获理想成绩！

谢谢大家！

