



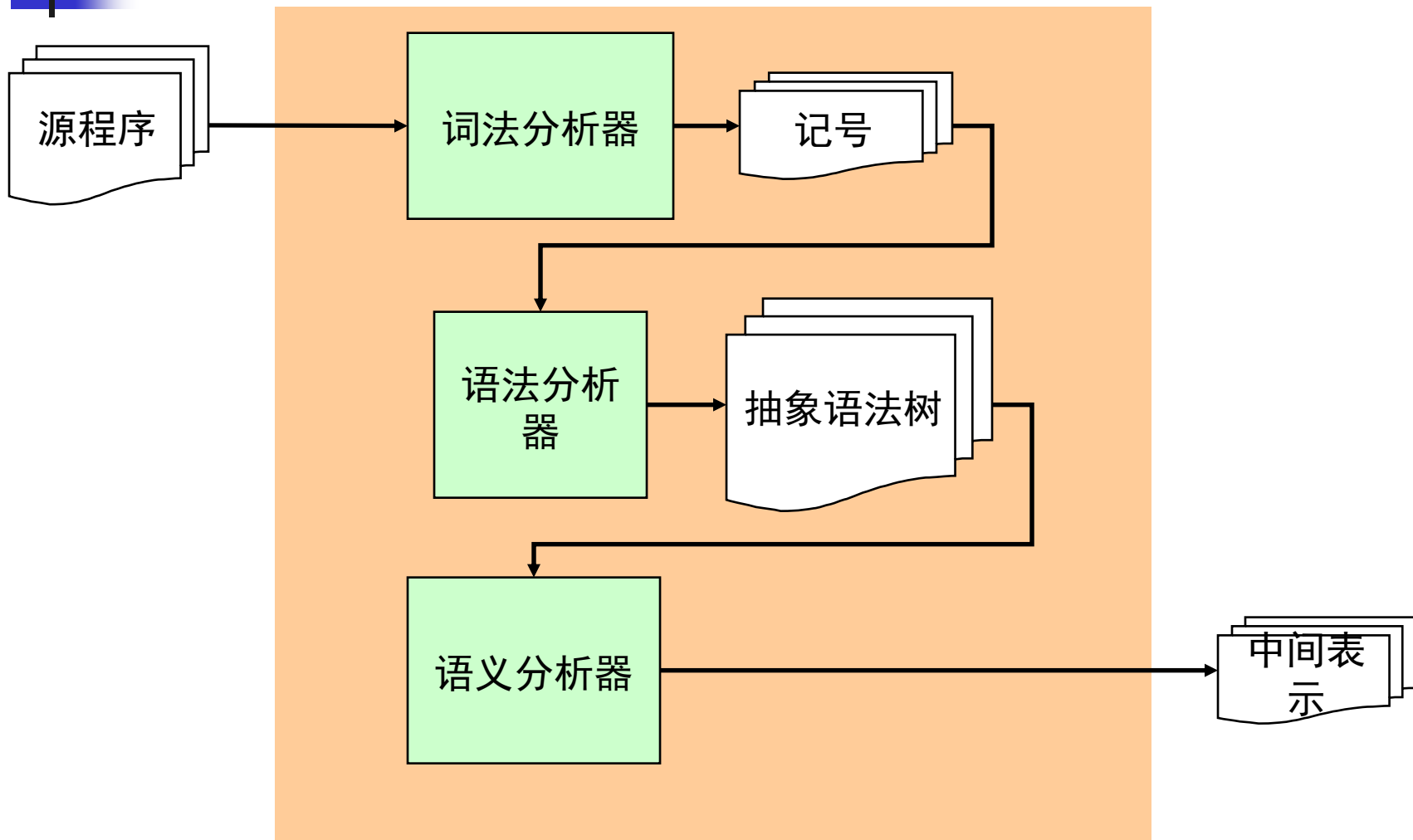
语法制导翻译

编译原理

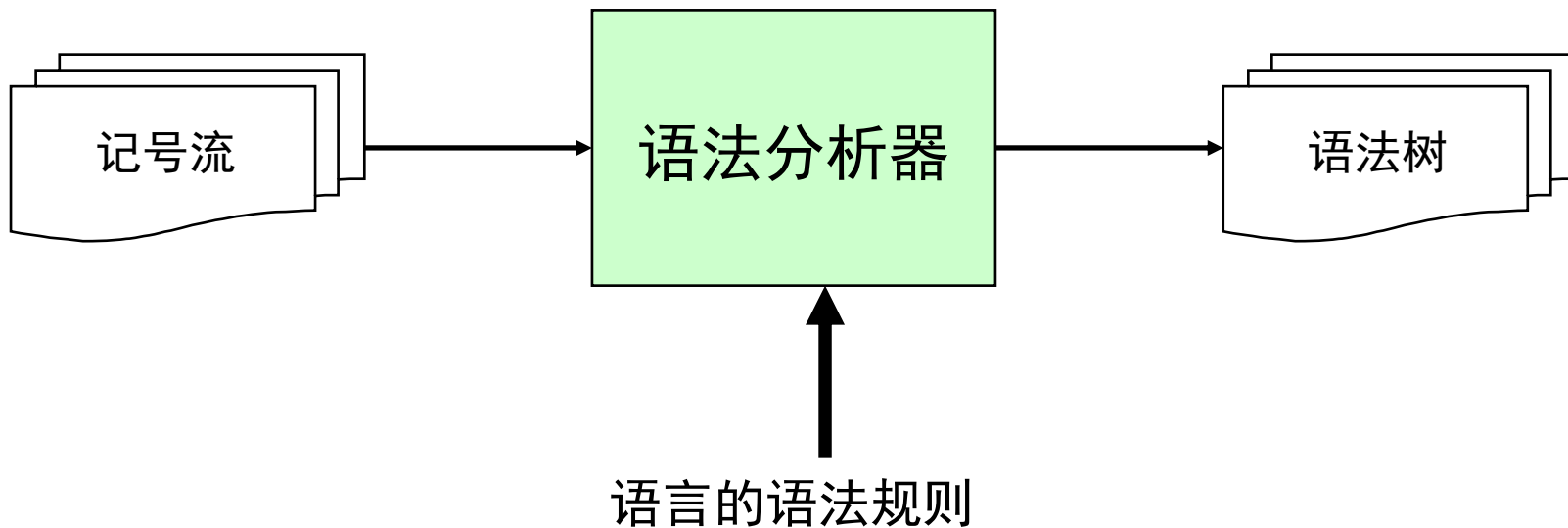
华保健

bjhua@ustc.edu.cn

前端



语法分析器的任务





语法制导的翻译

- 编译器在做语法分析的过程中，除了回答程序语法是否合法外，还必须完成后续工作
 - 可能的工作包括（但不限于）
 - 类型检查
 - 目标代码生成
 - 中间代码生成
 - 。 。 。
- 这些后续的工作一般可通过语法制导的翻译完成



基本思想

- 给每条产生式规则附加一个语义动作
 - 一个代码片段
- 语义动作在产生式“**归约**”时执行
 - 即当**右部**分析完毕时刻
 - 由**右部**的值计算**左部**的值
 - 自顶向下分析和自底向上分析采用的技术类似
 - 接下来重点讨论在自底向上的技术中的语法制导翻译

1:	$X \rightarrow \beta_1$	a_1
2:	$\mid \beta_2$	a_2
3:	$\mid \beta_3$	a_3
...		...
n:	$\mid \beta_n$	a_n



LR分析中的语法制导翻译

```
if (action[s, t]=="ri")  
    ai  
    pop( $\beta_i$ )  
    state s' = stack[top]  
    push (X)  
    push (goto[s', X])
```

1:	$X \rightarrow \beta_1$	a1
2:	β_2	a2
3:	β_3	a3
...		...
n:	β_n	a _n



示例

- 计算表达式的值

0: $E \rightarrow E + E$	$\{E = E1 + E2\}$
1: n	$\{E = n\}$

- 对应后序遍历的序
- 接下来看Yacc中的实现示例