



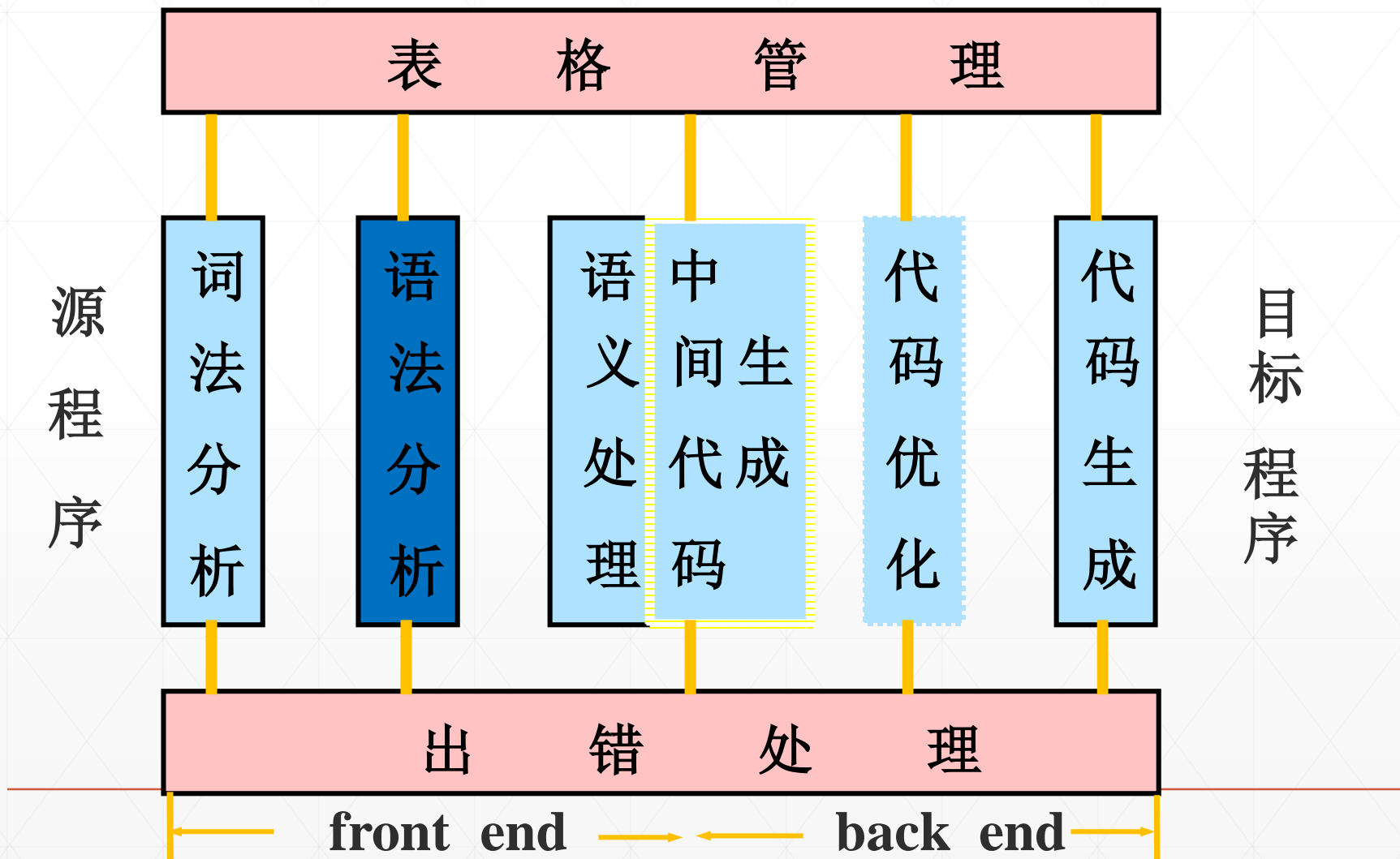
# 编译原理与设计

北京理工大学 计算机学院

---

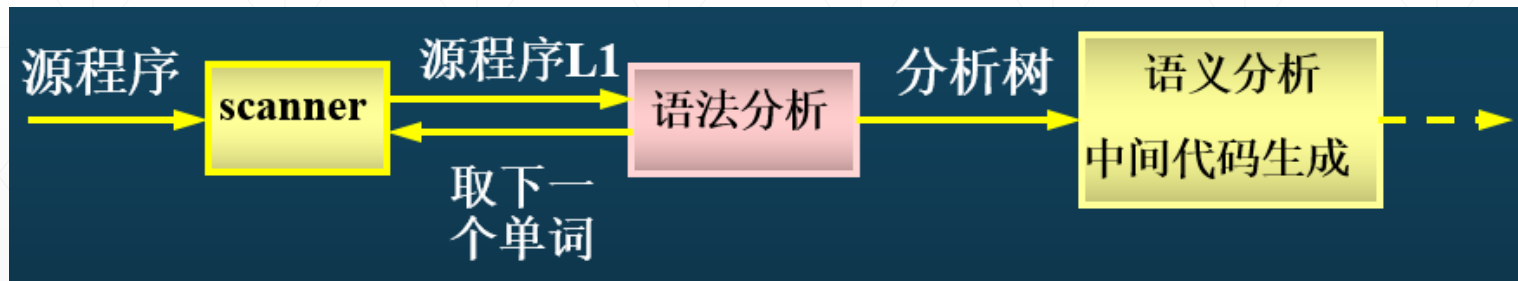
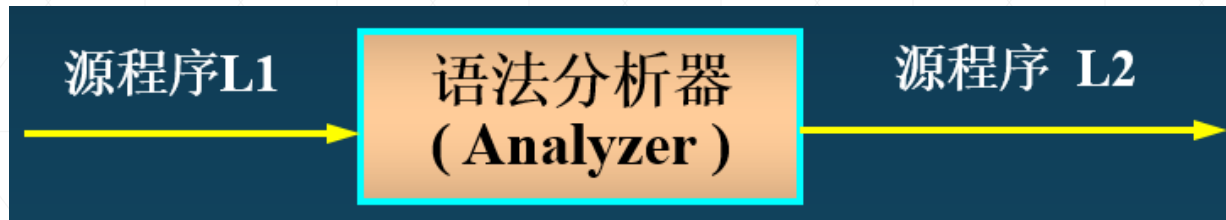


# 语法分析：概览





# 语法分析器功能





# 语法分析器功能

按照源语言的文法，从源程序串识别出相应的语法范畴，同时进行语法检查.

设给定文法 $G$ 和字符串(句子)  $\alpha$  ( $\alpha \in V_T^*$ )，检查、判定  $\alpha \in L(G)$ ? 即检查、判定  $\alpha$  是否是文法  $G$  所能产生的合法的句子，同时检查和处理语法错误。



# 语法分析器构造要素

- 处理对象：源程序串（L1形式）
- 分析依据：源语言的文法G
- 分析结果：识别出的语法范畴的表示

完成语法分析任务的程序称为语法分析器，或语法分析程序。



# 语法分析方法

## ■ 自上而下语法分析方法

给定文法 $G$ 和源程序串 $s$ 。从 $G$ 的开始符号 $S$ 出发，通过反复使用产生式对句型中的非终结符进行替换(推导)，逐步推导出 $s$ 。

是一种产生的方法，面向目标的方法。

分析的主旨是选择产生式的合适的候选式进行推导，逐步使推导结果与 $s$ 匹配。



# 语法分析方法

## ■ 自上而下语法分析方法

例：设有文法  $G$  和输入串  $S$

$G: S \rightarrow aA \mid a$

$A \rightarrow BaA \mid \varepsilon$

$B \rightarrow + \mid - \mid * \mid ,$

$S: a^*a+a$

$S \Rightarrow aA$

$\Rightarrow aBaA$

$\Rightarrow a^*aA$

$\Rightarrow a^*aBaA$

$\Rightarrow a^*a+aA$

$\Rightarrow a^*a+a$

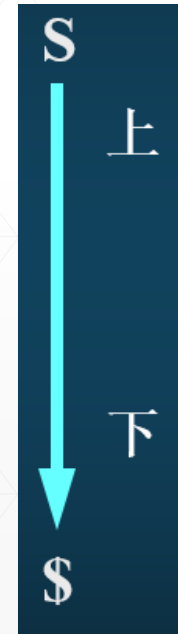
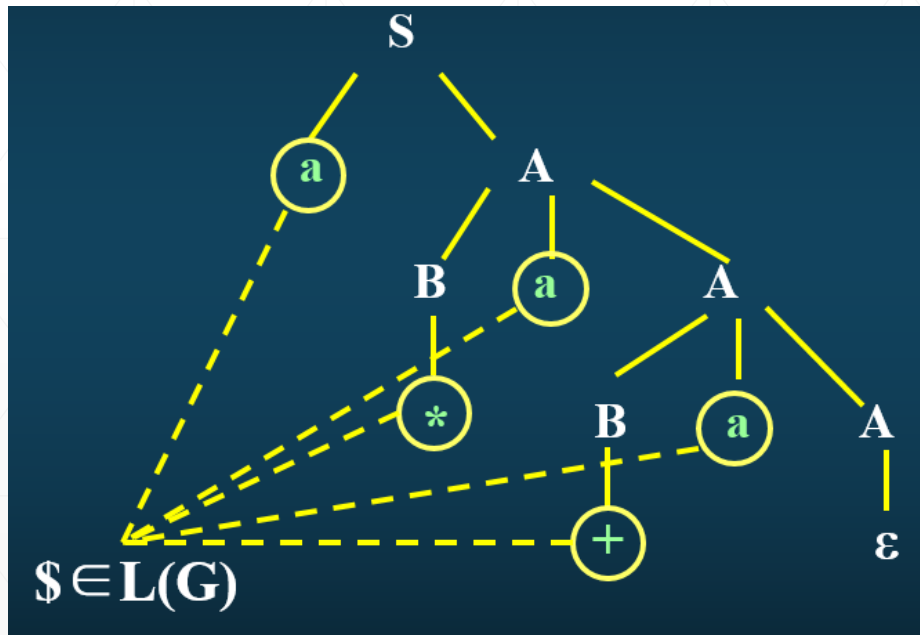
$S \in L(G)$



# 语法分析方法

- 自上而下语法分析方法

语法树的从左到右叶结点= $\$$ ，则  $\$ \in L(G)$ 。







# 语法分析方法

## ■ 自下而上语法分析方法

从给定的输入串 $S$ 开始，不断寻找子串与文法 $G$ 中某个产生式 $P$ 的候选式进行匹配，并用 $P$ 的左部代替(归约)之，逐步归约到 $S$ 。

是一种辨认的方法，基于目标的方法。

分析的主旨是寻找合适的子串与 $P$ 的侯选式进行匹配，直到归约到 $G$ 的 $S$ 为止。



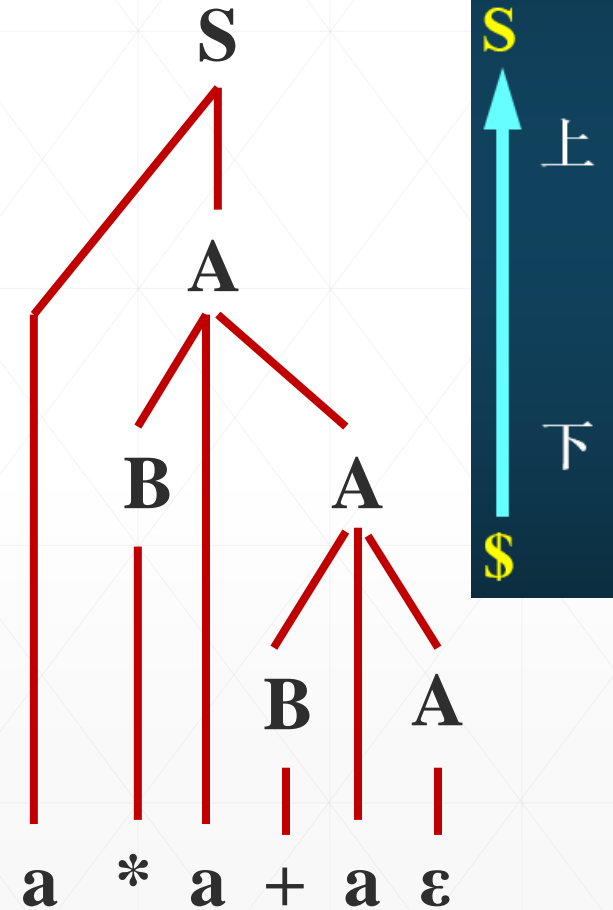
# 语法分析方法

## ■ 自下而上语法分析方法

例：设有文法G和输入串 \$

G:  $S \rightarrow aA \mid a$   
 $A \rightarrow BaA \mid \epsilon$   
 $B \rightarrow + \mid - \mid * \mid ,$   
\$:  $a*a+a$

$S \Rightarrow a\underline{A} \Rightarrow aBa\underline{A} \Rightarrow aBaBa\underline{A}$   
 $\Rightarrow aBa\underline{B}a \Rightarrow a\underline{B}a+a \Rightarrow a*a+a = \$$





# 语法分析方法

- 自上而下分析法:  $S \rightarrow \$$  (递归下降、LL(1))
  - 自下而上分析法:  $\$ \rightarrow S$  (OPG、LR)
  - 自上而下分析法的核心是不断寻找可匹配串对句型中最左的非终结符进行推导的过程;
  - 自下而上分析法的核心是不断寻找可归约串与P的候选式匹配, 并用P的左部的非终结符代替之。
-