

1. 语义分析的基本功能：确定类型、类型检查、语义处理和某些静态语义检查。
2. 对于 Chomsky 四种文法的四种语言之间的关系是： $L3 \in L2 \in L1 \in L0$ 。
3. 文法 G 所产生的 句子 的全体是该文法描述的语言。
4. 正规式 $M1$ 和 $M2$ 等价是指 $M1$ 和 $M2$ 所识别的语言集相等 或 $M1$ 和 $M2$ 表示的正规集相同。
5. 如果语法 G 是无二义的，那么对于每个句子，都只有一颗语法分析树（即使用最左/最右分析得到的树相同）。
6. 编译程序使用说明标识符的过程或函数的静态层次区别标识符的作用域。
7. 构造编译程序应掌握：源程序、编译方法和目标语言。
8. 扫描器是词法分析器，它接受输入的源程序，对其进行词法分析并识别出一个个单词符号，供语法分析器使用。
9. 词法分析器的输出结果是单词的种类编码和自身值。
10. 自下而上分析法采用移近、归约、错误处理、接受等四种操作。
11. 一个 LR 分析器包括两部分：一个总控程序和一张分析表。
12. 局部优化是在基本块范围内进行的一种优化。
13. 四元式之间的联系是通过临时变量实现的。（这个知识点告诉我们不要太相信 AI）
14. 针对循环优化：强度削弱、删除归纳变量、代码外提。
15. 同心集合并可能产生的新冲突是归约/归约冲突。
16. 代码优化所依据的是等价变换规则。
17. 过程的 DISPLAY 表记录了过程的嵌套层次。
18. 最右推导的逆过程成为规范归约（最左归约），规范归约的可归约串是句柄；算符优先文法的可归约串是最左素短语；简单优先分析每次规约的是最左直接短语。
 - 素短语首先是短语，其次至少包含一个终结符，并且自身不包含其他素短语。

1. 二义性文法的证明：该文法的某个句子存在 ≥ 2 棵语法树。
 2. 正规式转 DFA
 - 对于 x^* 来说，令 $node$ 自环一个 x 即可。
 3. 消除左递归： $s \rightarrow \beta s', s' \rightarrow \alpha s' | \epsilon$ ，其中 β 是终结部分， α 是递归部分。
 4. 判断 LL(1) 文法：求 select 集，对于 $A \rightarrow \alpha$ ，如果 $\alpha \overset{*}{\neq} \epsilon$ ， $select = FIRST(\alpha)$ ；否则， $select = FIRST(\alpha) - \{\epsilon\} \cup FOLLOW(A)$ 。
 - 注意：求 FOLLOW 集首先令 $\# \rightarrow S$ 。
 5. LL(1) 预测分析表：行是 V_N ，列是 V_T ，如果 $a \in select(A \rightarrow \alpha)$ ，那么在 (A, a) 这一格上填入 $\rightarrow \alpha$ （这里省略了左部）。
 6. LR(1) 展望符求法： $A \rightarrow \alpha \cdot B\beta, a$ ，对于 $B \rightarrow \dots, b$ 来说，如果 $\beta = \epsilon$ ， $b = a$ ；否则 $b = FIRST(A)$ ；另外，如果 $\epsilon \in FIRST(A)$ ，貌似要用 $\#$ 替代。
- LR(1) 分析表填法， S 正常填，对于 r ，只填展望符部分，格式为：

步骤	状态栈	符号栈	当前字符	剩余字符串	动作
1	0	#	a	bab#	移近

注意多个展望符用 | 连接。

7. 给定文法构造最小的 DFA。

- 首先画出其对应的 $\epsilon - NFA$ ，然后采用子集法将 $\epsilon - NFA$ 转换为 DFA。（ $\epsilon - closure$ ）
- 然后通过分割法进行化简：将 P_0 按照是否是终结符划分一次，之后对于每个组枚举出边接着划分到不能划分为止。

8. 算法优先文法（对于任意两个终结符之间至多只有一种优先关系）

- 先使用拓广文法， $E' \rightarrow E$ 。
- 求 FIRSTVT 集和 LASTVT 集。
 - 对于 $A \rightarrow a \cdots | Ba \cdots$ ，将 a 加入 $FIRSTVT(A)$ ；对于 $A \rightarrow B \cdots$ ，将 $FIRSTVT(B)$ 加入 $FIRSTVT(A)$ ；
 - 对于 $A \rightarrow \cdots a | \cdots aB$ ，将 a 加入 $LASTVT(A)$ ；对于 $A \rightarrow \cdots B$ ，将 $LASTVT(B)$ 加入 $LASTVT(A)$ 。
- 构造算符优先关系表，依次按照 $a \doteq b$ ， $a < b$ 和 $a > b$ 的顺序构造。
- 进行算符优先关系，如果 $top < in$ ，入栈； $top > in$ ，寻找合适的表达式归约； $top \doteq in$ ，入栈或结束。

9. 综合属性来自于子节点或自己；继承属性来自于父节点或兄弟节点。

10. 语法制导与循环、回边、流图：<https://zhuanlan.zhihu.com/p/1921637660969132194>