1. 短语、直接短语、句柄

短语：若有且，那么是句型相对A的短语

直接短语：A能一步推导出

句柄：最左直接短语。在语法树中，句柄是最左子树的端末结的自左至右排列，该子树有且只有两层。

1. 算符优先分析
2. LR分析法
3. 问题：构造一种自下而上分析法，从而求出输入串的最左规约
4. 核心：ACTION表&GOTO表
5. ACTION表决定移进还是规约。若移进，ACTION告诉了移进后的状态；若规约，ACTION告诉了用哪条产生式规约。
6. GOTO表决定规约后的新状态是什么。
7. 规约时，注意符号栈和状态栈的长度始终相等，若规约串的长度为a，则要同时从符号栈和状态栈中弹出a个符号/状态，然后再移进规约后的新符号和新状态。
8. 构造思路

构建拓广文法🡪构建CLOSURE闭包&GO函数🡪构建项目集规范族🡪构造LR(0)的ACTION表&GOTO表

1. 具体做法
2. 构建拓广文法：在原文法前加入一个新开始状态S’，目的是这样会有一个唯一的接受状态，S’🡪S·
3. 构建CLOSURE闭包
4. I的任意项目都属于CLOSURE(I)
5. 若属于CLOSURE(I)，则对B的任何产生式，把加入到CLOSURE(I)中。
6. 重复上述操作直到闭包不再变大
7. 构建GO函数
8. 直观理解：GO(I,X)=CLOSURE(J)就是在状态为I的时候，输入符号为X时转移到了状态J。然后再对J求闭包构成下一个状态。
9. 构建项目族规范集
10. 从开始符号开始，先构建开始符号的CLOSURE(S’)闭包
11. 对每个文法符号X，构建GO(S’,X)
12. 对每个GO函数产生的新项目，构建其CLOSORE闭包
13. 重复上述过程，直到项目族规范集不再增大
14. 构建LR(0)的ACTION表&GOTO表
15. SLR分析法

构造过程和LR分析法基本相同，唯一不同的是在判断规约动作（设置ACTION[K,a]）的时候，需要保证a属于FOLLOW(A)。