编译原理（复习）  
  
主要复习内容：  
一、介绍  
1. 什么是编译程序：将一种语言翻译为另一种语言的计算机程序。  
2、编译程序的基本结构，每个部分的简单介绍。  
3、若干名字：前端、后端、遍  
  
二、 词法分析（自动生成，LEX的应用）  
1、词法分析程序： 自动生成。  
给出一组关于单词的正规式描述，生成表驱动的词法分析器。正规表达式 → 有限状态自动机 → 生成表驱动的词法分析程序  
输入缓冲区  
2、正规表达式：  
（1） 正规表达式的操作：链接，乘方 Xn = X•X•X•••••X  
（2） 闭包\*：L\* = L0 U L1 U L2 U L3 U••••• , L0={ε}   
（3） 正则闭包+：L+ = L1 U L2 U L3 U•••••  
（4） 选择操作：|   
3、有限自动机：识别由正规表达式描述的单词  
DFA, NFA  
正规表达式 → NFA → DFA（确定化，最小化）  
4、 利用lex自动生成扫描程序  
lex的源程序：  
文件一般格式：  
{定义}  
%%  
{规则}  
%%  
{用户程序}  
  
定义段：可选，包括一些定义，C代码要放在%{和%}之间。  
规则段：具有如下的形式，动作是一个C语言的语句，或由{及}括起来的一串C语言程序段。   
正则表达式 动作  
用户程序段：在规则段中要用到的一些函数或子程序等。  
  
三、 语法分析   
  
上下文无关文法与分析  
1、基本概念与定义： 左部 → 右部  
上下文无关文法G是一个四元组 G=（VN，VT，P，S）  
VN ：非终结符集合  
VT ：终结符集合  
P ：A →α 的集合，A∈VN，α∈（VN∪VT）\*  
S ：开始符号  
2、 推导、由文法定义的语言与分析树（抽象语法树）  
最右推导，最左推导  
3、文法的二义性：若一个句子有二棵以上的分析树，则该文法称为二义文法（或歧义文法）  
3、 二义性的消除  
4、消除左递归和提取左因子：对文法做的修改  
消除左递归：A → Aα┃β  
A →βA ’  
A ’ →αA ’ ┃ε  
  
自顶向下的分析  
1、 递归下降的分析算法  
2、 FIRST集与FOLLOW集的计算  
3、 LL（1）分析  
LL（1）文法定义，LL（1）的含义  
LL（1）的分析算法（步骤）  
4、 构造LL（1）分析表  
5、 满足什么条件，该文法是LL（1）文法。  
6、LL（1）文法的构造，如何把一非LL（1）文法转换为LL（1）文法  
提取左公因子法、消除左递归。（消除二义性）  
7、错误恢复技术  
  
LR分析 （自底向上的分析）  
1、什么是LR分析？LR（1）的含义，什么是句柄。  
自下而上的分析过程：移进、归约、接受、出错。  
LR分析算法  
4、 LR（0）文法、SLR（1）文法、LR（1）文法、LALR（1）文法的定义。（LR状态机和LR分析表。）  
5、 满足什么条件，该文法是LR（1）文法等。  
6、 如何利用二义文法  
7、 LALR（1）分析器的自动生成工具：YACC的使用  
8、 YACC如何利用歧义文法，冲突的消除等等。  
9、 自底向上的错误校正  
  
四、 语义分析（对应第四、五章）  
抽象语法树  
符号表  
类型检查  
  
五、 运行环境 （对应第六章）  
活动记录  
活动树  
调用序列  
栈式存储（控制链，访问链）  
  
六、 中间代码生成（对应第七章）  
IR Tree表示：ESEQ，SEQ，MEM，MOVE，CJUMP…