CMM语言

词法分析 语法分析 语义分析

编译则是：语义生成后产生中间代码，还要代码优化

解释执行：词法分析🡪语法分析🡪语义分析🡪执行

实验1：

目的：设计CMM解释器的总体架构，设计制作一个分析单词的词法分析器，加深对于词法分析的理解

要求：输入形式为文件输入，输出结果需要显示出词法分析器对于输入内容分析的结果

具体功能要求：

1. 忽略空格tab回车换行等
2. 识别不同类型标识符
3. 识别并完全忽略注释
4. 记录下每个标识符的行号/位置
5. 输出识别的标识符
6. 如果存在词法错误，予以报错

实验内容步骤：

1. 给出词法的形式化描述，附上定义的码表
2. 做好词法的相关文档的准备（定义，困难，解决，效果，充分测试等）

建议与附加点：

1. 语法错误/边界点的充分测试，并给出具体测试结果（说明测试内容，出现的问题，怎么解决，综述汇报）
2. 第一个实验要求当面检查验收

实验结果与总结

1. 占比25%（总体搭建5%+20%），分4部分，每部分都会25%

实验2 CMM语法分析

说明

1. 做好扩充的准备，因为词法可能比较简单，要求语法更复杂
2. 以单词token序列为输入，给出输出与错误处理
3. 在实验1基础上整合，最终输入是任意的CMM源程序文件，输出结果要求展示成类似语法树的结构形式（要求能说明语法成分，可以输出屏幕也可以输出到某个文件中去）
4. 任何语法分析方法均可以使用：递归下降/预测分析表/LR方法
5. 实验报告要求给出程序结构说明，也就是程序/算法流程图的形式，还要给出主要的数据结构，算法说明，调用步骤等
6. 设计并展示充分的测试数据与例子，验收过程会予以验证，老师也会给出足够的测试样例检查
7. 实验报告总结要求能体现：问题/难点，解决方法，独到见解，综述设计优缺点，存在问题
8. 可以分词法/语法两次检查，也可以一次检查
9. 报告中要求包括：语法成分的结构形式等

实验3：设计调试一个实现CMM的解释器

要求：

1. 在一、二的基础上，对输入的满足CMM语法的源程序文件进行解释执行
2. 输出：留下选择是否打印输出编译过程的每个阶段的输出结果，至少要求打印出代码最后生成的结果，以及最后解释执行得到的运行结果
3. 语义分析注意点：类型检查（字符型加减乘除），变量的作用域与生成，数组越界，除数不为0，数组定义/多维，布尔表达式类型检查
4. 鼓励增加更多的功能，但是不可以减少附录1的CMM要求
5. 指出所使用的语义分析方法，根据语义分析方法给出相关的中间推导过程，说明具体的属性文法，进行何种分析。采用中间代码形式对语义分析的中间结果进行展示。
6. 更多加分可选项参考课件

C Minus Minus要求参考课件

实验4 JavaCC的使用

编写JavaCC模板文件，使用JavaCC自动生成词法分析器等等，给出相应的文件内容，为什么编写模板？

实验评分要求：

1. 实验界面更友好，功能完整，测试充分，代码的可读性好（多写注释，代码低耦合，分成多个函数去分块执行，不要把函数写太长）（45%）
2. 实验报告完整，描述清晰正确，与源代码一致，源码组织合理，结构清晰，可读性好；（40%）
3. 出错分析与解决办法，好的设计思想，增添的语法功能以及相对应的解析，编译相关功能的提供等。（15%）

最后的评分项用来加分提升：

实验报告中🡪体现了

1. 问题/难点🡪怎么解决这个问题的
2. 比如过程中发现有没有可以改进的，换一种语法分析方法怎么样
3. 思考与拓展，比如设计一个多维数组的定义访问

提交与检查：

1. 允许从网络上参考学习下载，但是必须要理解代码的过程/问题解决
2. 前三次词法/语法/语义分析的源码与执行程序要求当面验收，也要求实验报告
3. 越早完成分数更高，但是尽量不要相似，发现抄袭减分
4. 加分项见提交内容的要求最后“另外：……”，多扩充功能加分
5. 每次验收在上机课提前申请，报号，然后上课依次验收