

Lab 2 实验报告

201840281 史成璐

实验二的任务是在词法分析和语法分析程序的基础上编写一个程序，对C++源代码进行语义分析和类型检查，并打印分析结果。我完成了所有基本要求，此外，我还完成了要求2.1与要求2.3，未完成要求2.2。

实现功能

符号表

符号表的设计使用了散列表，同时使用了实验二指导中P.J. Weinberger所提出的散列函数。`void init_hash()`函数用于初始化符号表。由于散列表申请的空间比较大，因此直接使用open addressing来解决冲突。`bool fill_in(FieldList f)`函数用于获取空的位置并将f插入到符号表中，返回是否成功。`FieldList query(char* name)`函数用于查询是否有名为name的记录，返回查询结果。

未能完成要求2.2，符号表不支持多层作用域。

类型表示

使用了实验二指导中提供的Type_和FieldList_结构表示C++语言中的类型。基本类型中用1表示INT，2表示FLOAT。在Type_中联合体还增加了关于函数类型function的相关属性，包括一个枚举类型表明是定义\声明，函数参数列表和函数返回值类型。

函数bool check_equivalent(Type x, Type y)用于判断两个类型是否等价。实现思路是首先比较Type_*型的x和y的属性kind是否相同，若相同，再分基本类型、数组类型、结构体类型、函数类型四种情况进行处理。基本类型直接比较，数组类型递归判断elem属性即可。对于结构体类型，实现了要求2.3，结构等价，循环判断两个结构体对应的成员类型，全部对应相同则结构体等价。为了实现函数声明定义等内容，还实现了函数等价判断，首先判断返回值类型，然后类似结构体等价循环判断两个函数对应的所有参数类型，若全部对应相同则函数等价。

语义分析

语义分析主要需要实现的是符号表的读写与变量类型的判断。在实验一中已经构建了一颗语法树，与C--的文法有相似结构。为实现语义分析，采用将非终结符的产生式封装为函数的方法，通过函数参数和返回值来传递其继承属性和综合属性。遍历语法树，对相应子节点根据不同的产生进行函数递归调用，完成语义分析。主要根据语义分析情况实现相应动作。

特殊情况：

对于要求2.1，为了实现函数的声明，对文法进行改写，增加了函数声明：**ExtDef -> Specifier FunDec SEMI**.

对于错误类型15，还要求结构体在定义时不能对域进行初始化。增加了一个静态整型全局变量 **isstructue**，若在结构体中则加1，离开结构体则减去1，以处理嵌套情况。若此值不为0则说明在结构体内，初始化变量报错。

为了便于检查错误18，函数是否未定义，增加了一个**char***类型的字符数组来记录所有函数名，一个静态整型全局变量 **func_num**，每当分析出一个函数就填函数名表并记录数量。在语义分析最后阶段，对符号表查询每一个函数名，检查是否定义。

编译测试

可以直接使用**Makefile**，执行**make**命令即可生成**parser**程序。

正确实现了实验二PDF中的所有样例。