

实验二

141220145 张瑞琦

一、功能实现

步骤：

- 对语法树结点进行修改，添加对应的语法规则
- 构建符号表，实现插入与查表函数
- 深度遍历语法树，根据结点对应语法规则调用函数进行错误检查

具体来说：

- 当遇到ExtDef或者Def，首先遍历Specifier结点，获得类型，将类型做为继承属性传给其他结点，这些结点再将类型属性向子结点传，直到Vardec结点。
 - 对于Vardec->Vardec1 LB INT RB规则，调整Vardec1结点的类型为数组，数组元素类型为其继承的类型，大小为INT结点的值，做为Vardec1结点的继承属性，再继续对Vardec1结点进行深度遍历。遍历结束后，将Vardec1的类型做为综合属性传给Vardec。
 - 对于Vardec->ID规则，ID的类型为Vardec的继承属性，首先查表，判断是否重定义，如果没有重定义就进行变量插入操作。
- 但是对于ExtDef->Specifier FunDec Compst规则中的FunDec和Compst结点，除了进行上面的操作，还要进行以下操作：
 - 对于FunDec结点，VarList的继承属性为函数的参数个数argc与参数指针argv。当深度遍历到VarDec后，返回到ParamDec结点时，将类型添加到ParamDec的argv中，argc++，最后将argc一层层返回到FunDec结点，于是函数的参数个数和参数类型获得了。后面就先查表，没有重定义就将函数插入表中。
 - 对于Compst结点，将继承属性传到所有的子结点，到Stmnt->RETURN Exp时用于判断返回类型时候和函数返回类型相符。
- 当遇到Exp，先进行深度遍历，直到规则Exp->ID, Exp->INT, Exp->FLOAT的结点
 - 对于Exp->ID规则，查表，先判断是否进行了定义，如果进行了定义，将ID的类型传给Exp
 - 对于Exp->INT规则，将Exp类型赋值为INT
 - 对于Exp->FLOAT规则，将Exp类型赋值为FLOAT
 - 返回其他规则，根据Exp的类型进行相应的类型检查
- 当遇到Specifier
 - 对于Specifier->TYPE规则，根据TYPE结点的词素给Specifier结点赋相应的类型
 - 对于Specifier->StructSpecifier
 - 对于StructSpecifier->STRUCT OptTag LC DefList RC，先遍历OptTag结点，返回结构名；查表，看是否重定义，如果没有重定义，则初始化结构体的类型kind = STRUCTURE，并将结构体的名字和类型放入栈中。遍历DefList结点，Def结点，直到Vardec结点，在Vardec->ID规则的对应函数添加这个操作：如果栈不为空，就将

变量加入栈顶元素类型的域中。一层层返回到StructSpecifier结点，从栈顶取出结构名与类型，将StructSpecifier赋为该类型，并插入符号表。

- 对于StructSpecifier->STRUCT Tag，遍历Tag结点，查表，判断是否进行过定义，如果进行过，则返回结构体的类型给StructSpecifier
- StructSpecifier结点将类型赋给Specifier

二、编译方式

编译命令：make

运行：make test

三、亮点

1. 对算术运算结果进行了类型处理。（报错后）

实现：对于算术运算，如果操作数为int和float，将运算结果类型上升为float。

2. 区分域名在不在同一结构体中。

实现：在符号表中添加addr变量，域名的addr为结构名，其他不在结构中的变量的addr为"\0"。

3. 对于匿名结构体

实现：因为标识符不可能以数字开头，所以我给匿名结构体以数字（计数器提供）赋名，后面实现同2。但是这个结构名不会添加到符号表中。

4. 结构体嵌套结构体时确定结构体的域

实现：使用栈进行实现。栈的结构如下：

```
6    struct {
7        Type structType;
8        char* structName;
9    } structinfo[100];
10    int top;
```

每次进入StructSpecifier结点时，top++，将结构名和初始化过的结构类型，放到栈顶。子结点遍历结束后，从栈顶取出结构名和类型，top--

5. 函数名可以和变量名重复的问题，以及区分错误类型2与11

实现：在符号表中添加idkind变量，表示这个变量的类型，有变量，类型和函数。定义两个函数lookupVariable(char* name)与lookupFunction(char* name)。lookupVariable返回查询的变量 / 类型；lookupFunction返回查询的函数，如果查询不到函数，则会返回同名的变量 / 类型。

- 变量名重定义：lookupVariable返回不为空则重定义
- 函数名重定义：lookupFunction返回不为空且idkind为FUNCTION则重定义
- 错误类型11：lookupFunction返回不为空且idkind不为FUNCTION