

第三次实验报告

201250044 王星云

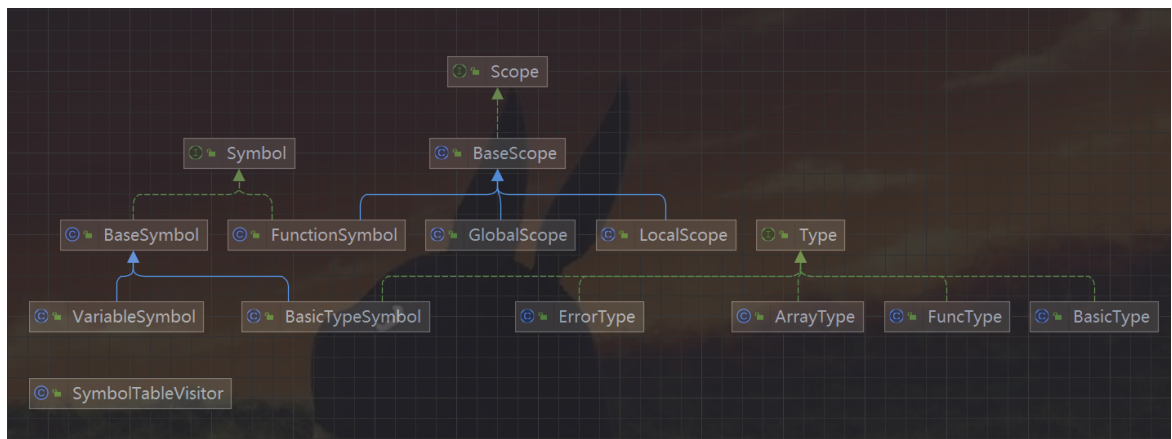
一、程序实现功能、实现方式、精巧设计

(一) 程序实现功能

程序对使用SysY语言书写的源代码进行完成类型检查与变量重命名，当输入文件存在完成类型错误时按要求输入所有的错误，当输入文件不存在完成类型错误时按要求输入重命名变量后的语法树（含高亮）。

(二) 实现方式

1. 设计类型



如图，我主要设计了Symbol、Scope和Type三个接口，其中Type分为ArrayType、FuncType和BasicType

2. 设计符号表

我在每个Scope里面，用一个Map存放符号，可以根据符号名来获取符号，并且提供了递归向父作用域查找符号的接口。

在进入一个函数或者block时，会进入一个新的作用域，而退出时，会回到父作用域。

3. 遍历语法树，发现语义错误

4. 变量重命名

(1) 第一次遍历语法树时构建了符号表并且记录下每个符号被使用的行号和列号。

(2) 在第一次遍历时将输出信息存储起来在一个List中，如果是符号就存放Symbol，否则就存放String。

(3) 如果没有类型错误，就输出语法树：进行判断，不是Symbol就直接输出，否则需要判断这个符号是否需要被重命名

(三) 精巧设计

1. 变量重命名的设计很精妙，只需要遍历一次语法树
2. 利于Java泛型，在visitor中每个方法返回一个Type，这样就能很容易判断类型

二、有趣的现象和印象深刻的bug

(一) 有趣的现象

我最终使用visitor完成本次实验，在使用visitor时，你如果在访问父节点时没有很好操控，子节点就很可能无法被访问到、或者是被多次访问。

(二) 印象深刻的bug

我一开始把SysY语言的类型判断当作C语言来做，把 `func()+1` 判断为对（如果函数 `func()` 返回int值），后来发现这是不可以的。