编译原理 PA3 实验报告

计62 徐晟 2016011253

1. Scopy：

修改Tree.java：添加新的Variable支持拷贝。

修改Translator：添加genScopy()，支持从src到dst的复制。

修改TransPass2：重载visitScopy()函数，使用genScopy()进行类的拷贝。

1. Sealed：

没有任何修改。

1. Guard List：

分为Guard（单条条件语句）和Guarded（卫士条件语句）。

修改TransPass2：重载visitGuard()，判断当前expr是否为true，则跳转。

修改TransPass2：重载visitGuarded()，for循环依次执行Guard语句。

1. Var Infer：

由于之前已经对var是否能够执行进行判断了，所以这次只要直接赋值类型就好了。

修改TransPass2：重载visitVar()，对dst的类型进行赋值。

1. 数组操作：
   1. 数组常量初始化：

在Binary里进行操作，上次PA中已经完成了。

* 1. 数组下标动态访问：

修改TransPass2：重载visitArrayRef()，计算并检查数组下标范围。

* 1. Foreach语句：

日常不太会做，参考了网上的代码。

修改Parser.y：添加了新的类型和语句VarType。

修改Tree：添加的新的类VarType用于foreach中type的判断。

修改TransPass2：重载visitForeach，先计算array的位置，长度等信息，然后使用loop语句进行foreach操作。

1. 参考

Github：<https://github.com/adrenaline21/decaf-pa3>

参考原因：第一是我在沿用PA2的框架时不知道为什么Guard语句开始报错，是在Guard语句开始的地方报Sytnax Error错误，调试不出来，只能重新修改Parser和Lexer，同时添加了VarType。第二是不会写foreach，所以进行了参考。