

PA2 实验报告

姓名：杨雅儒

班级：无 73

学号：2017011071

主要工作:

首先对主要新增代码进行了阅读，然后将 PA1-A 的部分代码复制进来，之后便开始进行修改。

主要进行的修改如下：

特性 1

Tree.java：在 Visitor 类中加入 visitScopy 空方法，并在 Scopy 节点上加入 accept 方法。

BuildSym.java：重载 visitScopy 方法。

TypeCheck.java：重载 visitScopy 方法，在其中对于两个参数进行了逻辑判断，并使用了 BadScopyArgError 和 BadScopySrcError。

Parser.y：将之前对于 OCStmt 的处理进行了更改，IDENTIFIER 不仅作为 String 存储，而新生成一个 Ident 节点存储。更改了 OCStmt 传入的 loc 为 SCOPY 的 loc。

特性 2

Class.java: 增加 isSealed 布尔型变量, 表示是否为 sealed 类, 并新增 public 方法 getSealed()。

BuildSym.java: 在 visitTopLevel 处新增了对继承 sealed 的类的处理, 使用 BadSealedInherError 报错。

特性 3

Tree.java: 在 Visitor 类中加入 visitGuardedIf 和 visitIfSubStmt 空方法, 并在 GuardedIf 节点和 IfSubStmt 节点上加入 accept 方法

BuildSym.java: 重载 visitGuardedIf 方法和 visitIfSubStmt 方法, 直接使用 accept 进行递归调用。

TypeCheck.java: 重载 visitGuardedIf 方法和 visitIfSubStmt 方法, 并使用 checkTestExpr 进行 bool 类型判定。

特性 4

BuildSym.java: 重载 visitIdent 方法, 判断其 isVar 为 true 时进行变量声明, 参照 visitVarDef 方法实现; 重载 visitAssign 方法, 通过; 重载 visitBinary 方法。

TypeCheck.java: 修改 checkBinaryOp 方法, 使其处理了有任一边为 UNKNOWN 的情况; 修改 visitAssign 方法, 使得左边为 UNKNOWN 而右边已知时进行处理

BaseType.java: 新增 UNKNOWN, 用于表示未知类型的类型。

Symbol.java: 新增 setType()方法用于改变 Type。

Tree.java: 在 Expr 抽象类中新增 isLeft 判断是否在赋值语句左侧, 若不在则不可以进行 var 变量定义。

特性 5

(1)

TypeCheck.java: 在两个 switch 语句中都加入对 DMOD 的处理, 使用到了 BadArrElementError 以及 BadNewArrayLength。增加了对 TypeCheck.java 中的 visitIdent 的处理, 使得在 ident 为 VAR 且不在赋值语句左边时不进行向上类型检查和 lookup, 否则会被之前定义的变量类型所替代。

(2)

Tree.java: 在 Visitor 类中加入 visitDefaultArrayRef 空方法, 并在 DefaultArrayRef 节点上加入 accept 方法。

BuildSym.java: 重载 visitDefaultArrayRef 方法。

TypeCheck.java: 重载 visitDefaultArrayRef 方法, 使用 BadArrOperArgError 和 BadArrIndexError 以及 BadDefError。增加 checkCanBeArrElement(Type type)方法用于检查 type 能否作为数组元素。

(3)

Tree.java: 在 Visitor 类中加入 visitForeach 空方法, 并在 Foreach 节点上加入 accept 方法, 并加入 LocalScope 类型的 associatedScope 变量用以存储作用域; 增加

AttachedStmtBlock 节点，为方便而使其从 Block 节点继承，并增加相应常量，以处理作用域问题；

BuildSym.java：重载 visitForeach 方法；重载 visitAttachedStmtBlock 方法。

TypeCheck.java：重载 visitForeach 方法，使用 BadArrOperArgError，

BadForeachTypeError 等，加入 break 的处理；重载 visitAttachedStmtBlock 方法。

Parser.y：更改 ForeachStmt 和 BoundVariable 中的处理，使得 BoundVariable 整体作为 VarDef 进行处理，这也表示需要对应在 visitVarDef 和 visitTypeIdent 中加入关于

VAR 的处理，并且在 Tree.java 的 Foreach 类中更改其成员变量为 VarDef。加入

AttachedStmtBlock，以处理作用域问题。增加 AttachedStmt 以处理单句的情况。

SemValue.java：新增 Tree.AttachedStmtBlock 类型的变量 attBlock。

部分调试记录：

1. 注意在特性 4 中，由于 BuildSym 并不会访问到 Assign 语句，所以更无法访问到 Var 赋值语句，所以需要将中间缺失的 visit 都补上。有 visitAssign 和 visitBinary
2. 依然在特性 4 中，对于 q4-5 中出现一定问题，就是当 var 出现在右侧时，似乎不能当作定义变量了，但觉得应该是测例的问题...因为在 q4-1 中 a 没有报出未定义的错误...所以最后改为了 var 在右边时不报出在定义时出现的错误

3. 一定要注意区分兼容和相同，在 foreach 中会出现这样的问题。
4. 最开始在 Parser.y 的 AttachedStmtBlock 中对于不加大括号的情况直接写了 Stmt，但是 Stmt 可以变为 StmtBlock，这样将会导致和自己写的带大括号的情况发生冲突进而错误。最终在 Parser.y 中加了 AttachedStmt 表示除 StmtBlock 之外的 Stmt。