Points-to Analysis

```
module PointsToAnalysis
 Returns the points-to tuple targets for a variable before the execution of CFG node 'n'.
 @param n the CFG node
 @param u the pointer variable
 @return the pointed-to variable
def pointsToBefore(n : ICFGNode, u : IVariableDeclaration) : IVariableDeclaration = {
 b := cfa(n)
                                                                                                     11
 return pointsToAfter(b, u)
                                                                                                     13
 Returns the points-to tuple targets for a variable after the execution of CFG node 'n'.
                                                                                                     16
 @param n the CFG node
 @param u the pointer variable
 @return the pointed-to variable
                                                                                                     20
                                                                                                     21
def pointsToAfter(n : ICFGNode, u : IVariableDeclaration) : IVariableDeclaration = {
 // no new binding at the current node 'n'
                                                                                                     23
 assert undef pointsToAt(n)
 return pointsToBefore(n, u)
                                                                                                     25
                                                                                                     26
 // there is a binding at the current node but it does not affect '\mathbf{u}'
                                                                                                     27
 (x, y) := pointsToAt(n)
                                                                                                     28
 assert x != u
 return pointsToBefore(n, u)
                                                                                                     30
} alt {
 // there is a binding at the current node and it affects '\mathrm{u}'
                                                                                                     32
 // and the binding does not point to the null literal
                                                                                                     33
 (x, y) := pointsToAt(n)
 assert x == u
                                                                                                     35
 assert y not instanceOf NullExpression
 return variableInAssignmentSide_right(y)
                                                                                                     37
                                                                                                     39
                                                                                                     40
 Returns the points-to binding at a CFG node.
 The pointed-to element of the tuple has type IAssignmentSide, because
                                                                                                     42
 pointsToAfter needs to handle the null assignment as well.
                                                                                                     44
 @param n the CFG node
                                                                                                     45
 @return the current binding
                                                                                                     47
private def pointsToAt(n : ICFGNode) : (IVariableDeclaration, IAssignmentSide) = {
  (1, r) := extractSides(n)
                                                                                                     49
 u := variableInAssignmentSide_left(l)
                                                                                                     50
 return (u, r)
                                                                                                     51
                                                                                                     52
                                                                                                     54
 Returns the potential sides of assignments at a CFG node.
                                                                                                     55
                                                                                                     56
 @param n the CFG node
                                                                                                     57
 @return the sides of assignment(s)
                                                                                                     59
private def extractSides(n : ICFGNode) : (IAssignmentSide, IAssignmentSide) = {
 assert n instanceOf LocalVariableDeclaration
                                                                                                     61
 return (n, n.init)
                                                                                                     62
} alt {
                                                                                                     63
 e := extractExpression(n)
                                                                                                     64
 c := extractAssignment(e)
                                                                                                     65
```

```
return (c.left, c.right)
                                                                                                       66
                                                                                                       67
                                                                                                       68
 Returns the expressions at a CFG node n.
                                                                                                       70
                                                                                                       71
  @param n the CFG node
                                                                                                       72
 Oreturn the expressions at the node
                                                                                                       73
private def extractExpression(n : ICFGNode) : Expression = {
                                                                                                       75
 assert n instanceOf ExpressionStatement
                                                                                                       76
 return n.expr
                                                                                                       77
} alt {
  assert n instanceOf ForStatement
 return n.incr
                                                                                                       80
} alt {
                                                                                                       81
 assert n instanceOf WhileStatement
                                                                                                       82
 return n.condition
                                                                                                       83
} alt {
 assert n instanceOf DoWhileStatement
                                                                                                       85
  return n.condition
} alt {
                                                                                                       87
 assert n instanceOf IfStatement
  return n.condition
                                                                                                       89
} alt {
                                                                                                       90
  assert n instanceOf ElseIfPart
                                                                                                       91
 return n.condition
                                                                                                       92
} alt {
 assert n instanceOf SwitchStatement
                                                                                                       94
 return n.expression
                                                                                                       95
} alt {
                                                                                                       96
 assert n instanceOf LocalVariableDeclaration
                                                                                                       97
  return n.init
                                                                                                       99
                                                                                                       100
/**
                                                                                                       101
 Returns the assignment expression(s) in an expression node.
                                                                                                       102
                                                                                                       103
  @param e the expression
                                                                                                       104
 @return the assignment expression(s)
                                                                                                       106
private def extractAssignment(e : Expression) : AssignmentExpr = {
                                                                                                       107
  assert e instanceOf AssignmentExpr
 return e
                                                                                                       109
} alt {
                                                                                                       110
 assert e instanceOf ParensExpression
                                                                                                       111
 return extractAssignment(e.expression)
                                                                                                       112
} alt {
                                                                                                       113
 assert e instanceOf ExpressionList
                                                                                                       114
  return extractAssignment(e.expressions)
                                                                                                       115
} alt {
                                                                                                       116
 assert e instanceOf BinaryExpression
                                                                                                       117
  return extractAssignment(e.left)
                                                                                                       118
} alt {
                                                                                                       119
  assert e instanceOf BinaryExpression
                                                                                                       120
 return extractAssignment(e.right)
                                                                                                       121
 assert e instanceOf TernaryExpression
                                                                                                       123
  return extractAssignment(e.condition)
                                                                                                       124
} alt {
                                                                                                       125
 assert e instanceOf TernaryExpression
                                                                                                       126
                                                                                                       127
 return extractAssignment(e.thenExpr)
} alt {
                                                                                                       128
 assert e instanceOf TernaryExpression
                                                                                                       129
  return extractAssignment(e.elseExpr)
                                                                                                       130
                                                                                                       131
                                                                                                       132
                                                                                                       133
```

```
Returns the pointer variable from an assignmet left hand side
                                                                                                      134
  based on Andersen's rules.
                                                                                                      135
                                                                                                      136
  @param s the left hand side of an assignment
 @return the pointer variable
                                                                                                      138
                                                                                                      139
private def variableInAssignmentSide_left(s : IAssignmentSide) : IVariableDeclaration = {
 assert s instanceOf DerefExpr
                                                                                                      141
 u := variableInAssignmentSide_left(s.expression)
                                                                                                      142
 n := eval(s.ancestor<concept = ICFGNode>)
                                                                                                      143
 v := pointsToBefore(n, u)
                                                                                                      144
 return v
                                                                                                      145
} alt {
                                                                                                      146
  assert s instanceOf ParensExpression
 return variableInAssignmentSide_left(s.expression)
                                                                                                      148
                                                                                                      149
 return variableInAssignmentSide_primitive(s)
                                                                                                      150
                                                                                                      151
                                                                                                      152
                                                                                                      153
 Returns the pointed-to variable from an assignmet right hand side
 based on Andersen's rules.
                                                                                                      155
                                                                                                      156
 Oparam s the right hand side of an assignment
                                                                                                      157
 @return the pointed-to variable
                                                                                                      158
private def variableInAssignmentSide_right(s : IAssignmentSide) : IVariableDeclaration = {
                                                                                                      160
 assert s instanceOf ReferenceExpr
  return variableInAssignmentSide_primitive(s.expression)
                                                                                                      162
} alt {
                                                                                                      163
 assert s instanceOf DerefExpr
                                                                                                      164
 u := variableInAssignmentSide_right(s.expression)
                                                                                                      165
 n := eval(s.ancestor<concept = ICFGNode>)
 v := pointsToBefore(n, u)
                                                                                                      167
  return v
                                                                                                      168
} alt {
                                                                                                      169
 assert s instanceOf ParensExpression
                                                                                                      170
 return variableInAssignmentSide_right (s.expression)
                                                                                                      171
} alt {
                                                                                                      172
 u := variableInAssignmentSide_primitive(s)
                                                                                                      173
 n := eval(s.ancestor<concept = ICFGNode>)
                                                                                                      174
 v := pointsToBefore(n, u)
                                                                                                      175
  return v
                                                                                                      177
                                                                                                      179
 Returns the variable in an assignment side.
                                                                                                      180
                                                                                                      181
  @param s the assignment side (left or right)
                                                                                                      182
  @return the variable
                                                                                                      184
private def variableInAssignmentSide_primitive(s : IAssignmentSide) : IVariableDeclaration = {
  assert s instanceOf GlobalVarRef
 return s.var
                                                                                                      187
} alt {
                                                                                                      188
 assert s instanceOf LocalVarRef
                                                                                                      189
  return s.var
} alt {
                                                                                                      191
 assert s instanceOf ArgumentRef
                                                                                                      192
  return s.arg
} alt {
                                                                                                      194
 assert s instanceOf LocalVariableDeclaration
 return s
                                                                                                      196
                                                                                                      197
```