Proofs for file C:\Program Files\Escher Technologies\Perfect Developer\Examples\Refinement\BinarySearch.pd

Generated by Perfect Developer at 14:51:39 UTC on Friday February 17th 2006

Tool file versions: PDTool 3.03, builtin 3.03, rubric 3.03

Proved 33 of 33 verification conditions.

Proof of verification condition: Type constraint satisfied

Condition generated at: C:\Program Files\Escher Technologies\Perfect Developer\Examples\Refinement\BinarySearch.pd (22,27)

Condition defined at: built in declaration

To prove: $0 \leq 0$

Given: **self**.members.isndec

Proof:

[Take goal term]

[1.0] $0 \leq 0$

→ *[simplify]*

[1.1] **true**

Proof of verification condition: Loop initialisation establishes end condition or a valid variant

Condition generated at: C:\Program Files\Escher Technologies\Perfect Developer\Examples\Refinement\BinarySearch.pd (23,13)

Condition defined at: C:\Program Files\Escher Technologies\Perfect Developer\Examples\Refinement\BinarySearch.pd (33,29)

To prove: $0 \leq (\mathrm{k} - (\mathrm{i}$ **as int**$))$

Given: **self**.members.isndec, $\mathrm{i} = 0$ **as** nat, $0 \leq \mathrm{i}$, $\mathrm{k} = \#$**self**.members **as int**, $\neg(\mathrm{i} = \mathrm{k})$

Proof:

[Take given term]

[3.0] $\mathrm{i} = (0$ **as** nat$)$

→ *[simplify]*

[3.1] $\mathrm{i} = 0$

[Take given term]

[4.0] $\mathrm{k} = (\#$**self**.members **as int**$)$

→ *[simplify]*

[4.2] $0 = (-\mathrm{k} + \#$**self**.members$)$

[Take given term]

[5.0] $\neg(\mathrm{i} = \mathrm{k})$

→ *[from term 3.1, i is equal to 0]*

[5.1] $\neg(0 = \mathrm{k})$

→ *[from term 4.2, k is equal to #***self***.members]*

[5.2] $\neg(0 = \#$**self**.members$)$

→ *[simplify]*

[5.3] $0 < \#\mathbf{self}.\mathrm{members}$

[Take goal term]

[1.0] $0 \leq (\mathrm{k} - (\mathrm{i}\ \mathbf{as\ int}))$

→ *[from term 4.2, k is equal to #**self**.members]*

[1.1] $0 \leq (\#\mathbf{self}.\mathrm{members} - (\mathrm{i}\ \mathbf{as\ int}))$

→ *[from term 3.1, i is equal to 0]*

[1.2] $0 \leq (\#\mathbf{self}.\mathrm{members} - (0\ \mathbf{as\ int}))$

→ *[simplify]*

[1.6] $-1 < \#\mathbf{self}.\mathrm{members}$

→ *[from term 5.3, literala < #**self**.members is true whenever -1 < (0 + −literala)]*

Proof of rule precondition:

[1.6.0] $-1 < (0 + --1)$

→ *[simplify]*

[1.6.3] **true**

[1.7] **true**

Proof of verification condition: Loop body establishes end condition or decreases variant

Condition generated at: C:\Program Files\Escher Technologies\Perfect Developer\Examples\Refinement\BinarySearch.pd (36,17)

Condition defined at: C:\Program Files\Escher Technologies\Perfect Developer\Examples\Refinement\BinarySearch.pd (33,17)

To prove: $(\mathrm{k}_{loopend} - (\mathrm{i}_{loopend}\ \mathbf{as\ int})) < (\mathrm{k}_{loopstart\_23,13} - (\mathrm{i}_{loopstart\_23,13}\ \mathbf{as\ int}))$

Given: **self**.members.isndec, $\mathrm{i} = 0\ \mathbf{as}\ \mathrm{nat}$, $0 \leq \mathrm{i}$, $\mathrm{k} = \#\mathbf{self}.\mathrm{members}\ \mathbf{as\ int}$, $0 \leq \mathrm{i}_{loopstart\_23,13}$, $0 \leq \mathrm{i}_{loopstart\_23,13}$, $\mathrm{i}_{loopstart\_23,13} \leq \mathrm{k}_{loopstart\_23,13}$, $\mathrm{k}_{loopstart\_23,13} \leq \#\mathbf{self}.\mathrm{members}$, $\forall\ \mathrm{z} \in 0\ ..\ <\mathrm{i}_{loopstart\_23,13} \bullet \mathbf{self}.\mathrm{members}[\mathrm{z}\ \mathbf{as}\ \mathrm{nat}] < \mathrm{x}$, $\forall\ \mathrm{z} \in \mathrm{k}_{loopstart\_23,13}\ ..\ <(\#\mathbf{self}.\mathrm{members}) \bullet \neg(\mathbf{self}.\mathrm{members}[\mathrm{z}\ \mathbf{as}\ \mathrm{nat}] < \mathrm{x})$, $\forall\ \$\mathrm{x} \in \$\mathrm{attributeNames}(\mathbf{int}) \bullet \mathbf{different}(\mathrm{i}_{loopstart\_23,13}.\$\mathrm{x};\ \mathrm{i}_{loopstart\_23,13}) \Rightarrow \mathrm{i}.\$\mathrm{x}{=}\mathrm{i}_{loopstart\_23,13}.\$\mathrm{x}$, $\neg(\mathrm{i}_{loopstart\_23,13} = \mathrm{k}_{loopstart\_23,13})$, $0 \leq (\mathrm{k}_{loopstart\_23,13} - (\mathrm{i}_{loopstart\_23,13}\ \mathbf{as\ int}))$, $(\mathrm{k}_{loopstart\_23,13} - (\mathrm{i}_{loopstart\_23,13}\ \mathbf{as\ int})) \leq (\mathrm{k} - (\mathrm{i}\ \mathbf{as\ int}))$, $\mathrm{p} = (\mathrm{i}_{loopstart\_23,13} + \mathrm{k}_{loopstart\_23,13}) / 2$, $(\neg(\mathbf{self}.\mathrm{members}[\mathrm{p}\ \mathbf{as}\ \mathrm{nat}] < \mathrm{x}) \wedge (\mathrm{i}_{loopstart\_23,13} = \mathrm{i}_{loopend}) \wedge (\mathrm{k}_{loopend} = \mathrm{p})) \vee ((\mathrm{i}_{loopend} = (>\mathrm{p}\ \mathbf{as}\ \mathrm{nat})) \wedge (\mathrm{k}_{loopstart\_23,13} = \mathrm{k}_{loopend}) \wedge (\mathbf{self}.\mathrm{members}[\mathrm{p}\ \mathbf{as}\ \mathrm{nat}] < \mathrm{x}) \wedge (0 \leq \mathrm{i}_{loopend}))$, $(\mathrm{p} < \mathrm{k}_{loopstart\_23,13}) \wedge (\mathrm{i}_{loopstart\_23,13} \leq \mathrm{p})$, $\neg(\mathrm{i}_{loopend} = \mathrm{k}_{loopend})$

Proof:

[Take goal term]

[1.0] $(\mathrm{k}_{loopend} - (\mathrm{i}_{loopend}\ \mathbf{as\ int})) < (\mathrm{k}_{loopstart\_23,13} - (\mathrm{i}_{loopstart\_23,13}\ \mathbf{as\ int}))$

→ *[simplify]*

[1.8] $0 < (-\mathrm{i}_{loopstart\_23,13} + -\mathrm{k}_{loopend} + \mathrm{i}_{loopend} + \mathrm{k}_{loopstart\_23,13})$

→ *[negate goal and search for contradiction]*

[1.9] $\neg(0 < (-\mathrm{i}_{loopstart\_23,13} + -\mathrm{k}_{loopend} + \mathrm{i}_{loopend} + \mathrm{k}_{loopstart\_23,13}))$

→ *[simplify]*

[1.18] $-1 < (-\mathrm{i}_{loopend} + -\mathrm{k}_{loopstart\_23,13} + \mathrm{i}_{loopstart\_23,13} + \mathrm{k}_{loopend})$

[Take given term]

[12.0] $((\mathrm{i}_{loopstart\_23,13} + \mathrm{k}_{loopstart\_23,13}) / 2) = \mathrm{p}$

→ *[simplify]*

[12.1] $0 = (-\mathrm{p} + ((\mathrm{i}_{loopstart\_23,13} + \mathrm{k}_{loopstart\_23,13})\ /\ 2))$

[Take given term]

[14.0] $(\mathrm{p} < \mathrm{k}_{loopstart\_23,13}) \wedge (\mathrm{i}_{loopstart\_23,13} \leq \mathrm{p})$

$\rightarrow$ *[simplify]*

[14.8] $(0 < (-\mathrm{p} + \mathrm{k}_{loopstart\_23,13})) \wedge (\text{-}1 < (-\mathrm{i}_{loopstart\_23,13} + \mathrm{p}))$

[Work on sub-term 2 of conjunction in term 14.8]

[15.0] $0 < (-\mathrm{p} + \mathrm{k}_{loopstart\_23,13})$

$\rightarrow$ *[from term 12.1, p is equal to $(\mathrm{i}_{loopstart\_23,13} + \mathrm{k}_{loopstart\_23,13})\ /\ 2$]*

[15.1] $0 < (-((\mathrm{i}_{loopstart\_23,13} + \mathrm{k}_{loopstart\_23,13})\ /\ 2) + \mathrm{k}_{loopstart\_23,13})$

$\rightarrow$ *[simplify]*

[15.11] $0 < (-\mathrm{i}_{loopstart\_23,13} + \mathrm{k}_{loopstart\_23,13})$

[Take given term]

[16.0] $\neg(\mathrm{i}_{loopend} = \mathrm{k}_{loopend})$

$\rightarrow$ *[simplify]*

[16.1] $\neg(0 = (-\mathrm{k}_{loopend} + \mathrm{i}_{loopend}))$

[Assume known post-assertion, class invariant or type constraint for term 16.1]

[17.0] $0 \leq \mathrm{i}_{loopend}$

$\rightarrow$ *[simplify]*

[17.2] $\text{-}1 < \mathrm{i}_{loopend}$

[Take given term]

[13.0] $(\neg(\textbf{self}.\mathrm{members}[\mathrm{p}\ \textbf{as}\ \mathrm{nat}] < \mathrm{x}) \wedge (\mathrm{i}_{loopstart\_23,13} = \mathrm{i}_{loopend}) \wedge (\mathrm{k}_{loopend} = \mathrm{p})) \vee ((\mathrm{i}_{loopend} = (>\mathrm{p}\ \textbf{as}\ \mathrm{nat})) \wedge (\mathrm{k}_{loopstart\_23,13} = \mathrm{k}_{loopend}) \wedge (\textbf{self}.\mathrm{members}[\mathrm{p}\ \textbf{as}\ \mathrm{nat}] < \mathrm{x}) \wedge (0 \leq \mathrm{i}_{loopend}))$

$\rightarrow$ *[simplify]*

[13.16] $(\neg(\textbf{self}.\mathrm{members}[\mathrm{p}] < \mathrm{x}) \wedge (0 = (-\mathrm{i}_{loopend} + \mathrm{i}_{loopstart\_23,13})) \wedge (0 = (-\mathrm{p} + \mathrm{k}_{loopend}))) \vee ((1 = (-\mathrm{p} + \mathrm{i}_{loopend})) \wedge (0 = (-\mathrm{k}_{loopend} + \mathrm{k}_{loopstart\_23,13})) \wedge (\textbf{self}.\mathrm{members}[\mathrm{p}] < \mathrm{x}) \wedge (\text{-}1 < \mathrm{i}_{loopend}))$

$\rightarrow$ *[from term 12.1, p is equal to $(\mathrm{i}_{loopstart\_23,13} + \mathrm{k}_{loopstart\_23,13})\ /\ 2$]*

[13.20] $(\neg(\textbf{self}.\mathrm{members}[(\mathrm{i}_{loopstart\_23,13} + \mathrm{k}_{loopstart\_23,13})\ /\ 2] < \mathrm{x}) \wedge (0 = (-((\mathrm{i}_{loopstart\_23,13} + \mathrm{k}_{loopstart\_23,13})\ /\ 2) + \mathrm{k}_{loopend})) \wedge (0 = (-\mathrm{i}_{loopend} + \mathrm{i}_{loopstart\_23,13}))) \vee ((0 = (-\mathrm{k}_{loopend} + \mathrm{k}_{loopstart\_23,13})) \wedge (1 = (-((\mathrm{i}_{loopstart\_23,13} + \mathrm{k}_{loopstart\_23,13})\ /\ 2) + \mathrm{i}_{loopend})) \wedge (\text{-}1 < \mathrm{i}_{loopend}) \wedge (\textbf{self}.\mathrm{members}[(\mathrm{i}_{loopstart\_23,13} + \mathrm{k}_{loopstart\_23,13})\ /\ 2] < \mathrm{x}))$

$\rightarrow$ *[from term 17.2, literala $< \mathrm{i}_{loopend}$ is true whenever -1 < (-1 + $-$literala)]*

Proof of rule precondition:

[13.20.0] $\text{-}1 < (\text{-}1 + -\text{-}1)$

$\rightarrow$ *[simplify]*

[13.20.3] **true**

[13.21] $(\neg(\textbf{self}.\mathrm{members}[(\mathrm{i}_{loopstart\_23,13} + \mathrm{k}_{loopstart\_23,13})\ /\ 2] < \mathrm{x}) \wedge (0 = (-((\mathrm{i}_{loopstart\_23,13} + \mathrm{k}_{loopstart\_23,13})\ /\ 2) + \mathrm{k}_{loopend})) \wedge (0 = (-\mathrm{i}_{loopend} + \mathrm{i}_{loopstart\_23,13}))) \vee ((0 = (-\mathrm{k}_{loopend} + \mathrm{k}_{loopstart\_23,13})) \wedge (1 = (-((\mathrm{i}_{loopstart\_23,13} + \mathrm{k}_{loopstart\_23,13})\ /\ 2) + \mathrm{i}_{loopend})) \wedge \textbf{true} \wedge (\textbf{self}.\mathrm{members}[(\mathrm{i}_{loopstart\_23,13} + \mathrm{k}_{loopstart\_23,13})\ /\ 2] < \mathrm{x}))$

$\rightarrow$ *[simplify]*

[13.22] $(\neg(\textbf{self}.\mathrm{members}[(\mathrm{i}_{loopstart\_23,13} + \mathrm{k}_{loopstart\_23,13})\ /\ 2] < \mathrm{x}) \wedge (0 = (-((\mathrm{i}_{loopstart\_23,13} + \mathrm{k}_{loopstart\_23,13})\ /\ 2) + \mathrm{k}_{loopend})) \wedge (0 = (-\mathrm{i}_{loopend}$

$+ \, i_{loopstart\_23,13}))) \lor ((0 = (-k_{loopend} + k_{loopstart\_23,13})) \land (1 = (-((i_{loopstart\_23,13} + k_{loopstart\_23,13}) \, / \, 2) + i_{loopend})) \land (\mathbf{self}.\mathrm{members}[(i_{loopstart\_23,13} + k_{loopstart\_23,13}) \, / \, 2] < x))$

Proof branches here giving 2 sub-goals:

Proof of sub-goal 1:

[Branch on disjunction or conditional in term 13.22 and work on branch 1]

[18.0] $\neg(\mathbf{self}.\mathrm{members}[(i_{loopstart\_23,13} + k_{loopstart\_23,13}) \, / \, 2] < x) \land (0 = (-((i_{loopstart\_23,13} + k_{loopstart\_23,13}) \, / \, 2) + k_{loopend})) \land (0 = (-i_{loopend} + i_{loopstart\_23,13}))$

[Work on sub-term 2 of conjunction in term 18.0]

[19.0] $0 = (-((i_{loopstart\_23,13} + k_{loopstart\_23,13}) \, / \, 2) + k_{loopend})$

[Work on sub-term 3 of conjunction in term 18.0]

[20.0] $0 = (-i_{loopend} + i_{loopstart\_23,13})$

[Copy term 1.18]

[21.0] $-1 < (-i_{loopend} + -k_{loopstart\_23,13} + i_{loopstart\_23,13} + k_{loopend})$

→ *[from term 20.0, $-i_{loopend} + i_{loopstart\_23,13}$ is equal to 0]*

[21.1] $-1 < (0 + -k_{loopstart\_23,13} + k_{loopend})$

→ *[simplify]*

[21.3] $-1 < (-k_{loopstart\_23,13} + k_{loopend})$

→ *[from term 19.0, $k_{loopend}$ is equal to $(i_{loopstart\_23,13} + k_{loopstart\_23,13}) \, / \, 2$]*

[21.4] $-1 < (-k_{loopstart\_23,13} + ((i_{loopstart\_23,13} + k_{loopstart\_23,13}) \, / \, 2))$

→ *[simplify]*

[21.15] $-1 < (-k_{loopstart\_23,13} + i_{loopstart\_23,13})$

→ *[from term 15.11, literala $< (-k_{loopstart\_23,13} + i_{loopstart\_23,13})$ is false whenever $-2 < (0 + \text{literala})$]*

Proof of rule precondition:

[21.15.0] $-2 < (-1 + 0)$

→ *[simplify]*

[21.15.2] **true**

[21.16] **false**

Proof of sub-goal 2:

[Work on branch 2 from term 13.22]

[23.0] $(0 = (-k_{loopend} + k_{loopstart\_23,13})) \land (1 = (-((i_{loopstart\_23,13} + k_{loopstart\_23,13}) \, / \, 2) + i_{loopend})) \land (\mathbf{self}.\mathrm{members}[(i_{loopstart\_23,13} + k_{loopstart\_23,13}) \, / \, 2] < x)$

[Work on sub-term 2 of conjunction in term 23.0]

[25.0] $1 = (-((i_{loopstart\_23,13} + k_{loopstart\_23,13}) \, / \, 2) + i_{loopend})$

[Copy term 1.18]

[24.0] $-1 < (-i_{loopend} + -k_{loopstart\_23,13} + i_{loopstart\_23,13} + k_{loopend})$

→ *[from term 23.0, $-k_{loopstart\_23,13} + k_{loopend}$ is equal to 0]*

[24.1] $-1 < (0 + -i_{loopend} + i_{loopstart\_23,13})$

→ *[simplify]*

[24.3] $-1 < (-i_{loopend} + i_{loopstart\_23,13})$

→ *[from term 25.0, $i_{loopend}$ is equal to $1 + ((i_{loopstart\_23,13} + k_{loopstart\_23,13}) \, / \, 2)$]*

[24.4] $-1 < (-(1 + ((i_{loopstart\_23,13} + k_{loopstart\_23,13}) \, / \, 2)) + i_{loopstart\_23,13})$

→ *[simplify]*

[24.23] $0 < (-k_{loopstart\_23,13} + i_{loopstart\_23,13})$

→ *[from term 15.11, literala < ($-k_{loopstart\_23,13}$ + $i_{loopstart\_23,13}$) is false whenever -2 < (0 + literala)]*

Proof of rule precondition:

[24.23.0] -2 < (0 + 0)

→ *[simplify]*

[24.23.2] **true**

[24.24] **false**

Proof of verification condition: Loop body establishes end condition or preserves validity of variant

Condition generated at: C:\Program Files\Escher Technologies\Perfect Developer\Examples\Refinement\BinarySearch.pd (36,17)

Condition defined at: C:\Program Files\Escher Technologies\Perfect Developer\Examples\Refinement\BinarySearch.pd (33,29)

To prove: $0 \le (k_{loopend} - (i_{loopend}$ **as int**$))$

Given: **self**.members.isndec, i = 0 **as** nat, $0 \le$ i, k = #**self**.members **as int**, $0 \le i_{loopstart\_23,13}$, $0 \le i_{loopstart\_23,13}$, $i_{loopstart\_23,13} \le k_{loopstart\_23,13}$, $k_{loopstart\_23,13} \le$ #**self**.members, ∀ z ∈ 0 .. <$i_{loopstart\_23,13}$ • **self**.members[z **as** nat] < x, ∀ z ∈ $k_{loopstart\_23,13}$ .. <(#**self**.members) • ¬(**self**.members[z **as** nat] < x), ∀ $x ∈ $attributeNames(**int**) • **different**($i_{loopstart_23,13}$.$x; $i_{loopstart\_23,13}$) ⇒ i.$x=$i_{loopstart_23,13}$.$x, ¬($i_{loopstart\_23,13} = k_{loopstart\_23,13}$), $0 \le (k_{loopstart\_23,13} - (i_{loopstart\_23,13}$ **as int**)), ($k_{loopstart\_23,13} - (i_{loopstart\_23,13}$ **as int**)) ≤ (k − (i **as int**)), p = ($i_{loopstart\_23,13} + k_{loopstart\_23,13}$) / 2, (¬(**self**.members[p **as** nat] < x) ∧ ($i_{loopstart\_23,13} = i_{loopend}$) ∧ ($k_{loopend}$ = p)) ∨ (($i_{loopend}$ = (>p **as** nat)) ∧ ($k_{loopstart\_23,13} = k_{loopend}$) ∧ (**self**.members[p **as** nat] < x) ∧ ($0 \le i_{loopend}$)), (p < $k_{loopstart\_23,13}$) ∧ ($i_{loopstart\_23,13} \le$ p), ¬($i_{loopend} = k_{loopend}$)

Proof:

[Take goal term]

[1.0] $0 \le (k_{loopend} - (i_{loopend}$ **as int**$))$

→ *[simplify]*

[1.4] $-1 < (-i_{loopend} + k_{loopend})$

→ *[negate goal and search for contradiction]*

[1.5] $\neg(-1 < (-i_{loopend} + k_{loopend}))$

→ *[simplify]*

[1.9] $0 < (i_{loopend} + -k_{loopend})$

[Take given term]

[12.0] $((i_{loopstart\_23,13} + k_{loopstart\_23,13}) / 2) = p$

→ *[simplify]*

[12.1] $0 = (-p + ((i_{loopstart\_23,13} + k_{loopstart\_23,13}) / 2))$

[Take given term]

[14.0] $(p < k_{loopstart\_23,13}) \wedge (i_{loopstart\_23,13} \le p)$

→ *[simplify]*

[14.8] $(0 < (-p + k_{loopstart\_23,13})) \wedge (-1 < (-i_{loopstart\_23,13} + p))$

[Work on sub-term 2 of conjunction in term 14.8]

[15.0] $0 < (-p + k_{loopstart\_23,13})$

→ *[from term 12.1, p is equal to* $(i_{loopstart\_23,13} + k_{loopstart\_23,13})$ */ 2]*

[15.1] $0 < (-((i_{loopstart\_23,13} + k_{loopstart\_23,13}) / 2) + k_{loopstart\_23,13})$

→ *[simplify]*

[15.11] $0 < (-i_{loopstart\_23,13} + k_{loopstart\_23,13})$

[Take given term]

[16.0] $\neg(i_{loopend} = k_{loopend})$

→ *[simplify]*

[16.1] $\neg(0 = (-k_{loopend} + i_{loopend}))$

[Assume known post-assertion, class invariant or type constraint for term 16.1]

[17.0] $0 \leq i_{loopend}$

→ *[simplify]*

[17.2] $-1 < i_{loopend}$

[Take given term]

[13.0] $(\neg(\textbf{self}.\text{members}[p\ \textbf{as}\ \text{nat}] < x) \wedge (i_{loopstart\_23,13} = i_{loopend}) \wedge (k_{loopend} = p)) \vee ((i_{loopend} = (>p\ \textbf{as}\ \text{nat})) \wedge (k_{loopstart\_23,13} = k_{loopend}) \wedge (\textbf{self}.\text{members}[p\ \textbf{as}\ \text{nat}] < x) \wedge (0 \leq i_{loopend}))$

→ *[simplify]*

[13.16] $(\neg(\textbf{self}.\text{members}[p] < x) \wedge (0 = (-i_{loopend} + i_{loopstart\_23,13})) \wedge (0 = (-p + k_{loopend}))) \vee ((1 = (-p + i_{loopend})) \wedge (0 = (-k_{loopend} + k_{loopstart\_23,13})) \wedge (\textbf{self}.\text{members}[p] < x) \wedge (-1 < i_{loopend}))$

→ *[from term 12.1, p is equal to* $(i_{loopstart\_23,13} + k_{loopstart\_23,13})$ */ 2]*

[13.20] $(\neg(\textbf{self}.\text{members}[(i_{loopstart\_23,13} + k_{loopstart\_23,13}) / 2] < x) \wedge (0 = (-((i_{loopstart\_23,13} + k_{loopstart\_23,13}) / 2) + k_{loopend})) \wedge (0 = (-i_{loopend} + i_{loopstart\_23,13}))) \vee ((0 = (-k_{loopend} + k_{loopstart\_23,13})) \wedge (1 = (-((i_{loopstart\_23,13} + k_{loopstart\_23,13}) / 2) + i_{loopend})) \wedge (-1 < i_{loopend}) \wedge (\textbf{self}.\text{members}[(i_{loopstart\_23,13} + k_{loopstart\_23,13}) / 2] < x))$

→ *[from term 17.2, literala <* $i_{loopend}$ *is true whenever -1 < (-1 + −literala)]*

Proof of rule precondition:

[13.20.0] $-1 < (-1 + --1)$

→ *[simplify]*

[13.20.3] **true**

[13.21] $(\neg(\textbf{self}.\text{members}[(i_{loopstart\_23,13} + k_{loopstart\_23,13}) / 2] < x) \wedge (0 = (-((i_{loopstart\_23,13} + k_{loopstart\_23,13}) / 2) + k_{loopend})) \wedge (0 = (-i_{loopend} + i_{loopstart\_23,13}))) \vee ((0 = (-k_{loopend} + k_{loopstart\_23,13})) \wedge (1 = (-((i_{loopstart\_23,13} + k_{loopstart\_23,13}) / 2) + i_{loopend})) \wedge \textbf{true} \wedge (\textbf{self}.\text{members}[(i_{loopstart\_23,13} + k_{loopstart\_23,13}) / 2] < x))$

→ *[simplify]*

[13.22] $(\neg(\textbf{self}.\text{members}[(i_{loopstart\_23,13} + k_{loopstart\_23,13}) / 2] < x) \wedge (0 = (-((i_{loopstart\_23,13} + k_{loopstart\_23,13}) / 2) + k_{loopend})) \wedge (0 = (-i_{loopend} + i_{loopstart\_23,13}))) \vee ((0 = (-k_{loopend} + k_{loopstart\_23,13})) \wedge (1 = (-((i_{loopstart\_23,13} + k_{loopstart\_23,13}) / 2) + i_{loopend})) \wedge (\textbf{self}.\text{members}[(i_{loopstart\_23,13} + k_{loopstart\_23,13}) / 2] < x))$

Proof branches here giving 2 sub-goals:

Proof of sub-goal 1:

[Branch on disjunction or conditional in term 13.22 and work on branch 1]

[18.0] $\neg(\textbf{self}.\text{members}[(i_{loopstart\_23,13} + k_{loopstart\_23,13}) / 2] < x) \wedge (0 = (-((i_{loopstart\_23,13} + k_{loopstart\_23,13}) / 2) + k_{loopend})) \wedge (0 = (-i_{loopend} + i_{loopstart\_23,13}))$

[Work on sub-term 2 of conjunction in term 18.0]

[19.0] $0 = (-((\mathrm{i}_{loopstart\_23,13} + \mathrm{k}_{loopstart\_23,13}) \ / \ 2) + \mathrm{k}_{loopend})$

[Work on sub-term 3 of conjunction in term 18.0]

[20.0] $0 = (-\mathrm{i}_{loopend} + \mathrm{i}_{loopstart\_23,13})$

[Copy term 1.9]

[21.0] $0 < (-\mathrm{k}_{loopend} + \mathrm{i}_{loopend})$

$\rightarrow$ *[from term 19.0, $\mathrm{k}_{loopend}$ is equal to $(\mathrm{i}_{loopstart\_23,13} + \mathrm{k}_{loopstart\_23,13}) \ / \ 2$]*

[21.1] $0 < (-((\mathrm{i}_{loopstart\_23,13} + \mathrm{k}_{loopstart\_23,13}) \ / \ 2) + \mathrm{i}_{loopend})$

$\rightarrow$ *[simplify]*

[21.8] $0 < (-\mathrm{i}_{loopstart\_23,13} + -\mathrm{k}_{loopstart\_23,13} + (2 * \mathrm{i}_{loopend}))$

$\rightarrow$ *[from term 20.0, $\mathrm{i}_{loopend}$ is equal to $\mathrm{i}_{loopstart\_23,13}$]*

[21.9] $0 < (-\mathrm{i}_{loopstart\_23,13} + -\mathrm{k}_{loopstart\_23,13} + (2 * \mathrm{i}_{loopstart\_23,13}))$

$\rightarrow$ *[simplify]*

[21.12] $0 < (-\mathrm{k}_{loopstart\_23,13} + \mathrm{i}_{loopstart\_23,13})$

$\rightarrow$ *[from term 15.11, literala $< (-\mathrm{k}_{loopstart\_23,13} + \mathrm{i}_{loopstart\_23,13})$ is false whenever -2 < (0 + literala)]*

Proof of rule precondition:

[21.12.0] -2 < (0 + 0)

$\rightarrow$ *[simplify]*

[21.12.2] **true**

[21.13] **false**

Proof of sub-goal 2:

[Work on branch 2 from term 13.22]

[22.0] $(0 = (-\mathrm{k}_{loopend} + \mathrm{k}_{loopstart\_23,13})) \wedge (1 = (-((\mathrm{i}_{loopstart\_23,13} + \mathrm{k}_{loopstart\_23,13}) \ / \ 2) + \mathrm{i}_{loopend})) \wedge$ (**self**.members[$(\mathrm{i}_{loopstart\_23,13} + \mathrm{k}_{loopstart\_23,13}) \ / \ 2$] < x)

$\rightarrow$ *[separate conjunction and work on first sub-term]*

[22.1] $0 = (-\mathrm{k}_{loopend} + \mathrm{k}_{loopstart\_23,13})$

[Work on sub-term 2 of conjunction in term 22.0]

[23.0] $1 = (-((\mathrm{i}_{loopstart\_23,13} + \mathrm{k}_{loopstart\_23,13}) \ / \ 2) + \mathrm{i}_{loopend})$

[Copy term 1.9]

[25.0] $0 < (-\mathrm{k}_{loopend} + \mathrm{i}_{loopend})$

$\rightarrow$ *[from term 22.1, $\mathrm{k}_{loopend}$ is equal to $\mathrm{k}_{loopstart\_23,13}$]*

[25.1] $0 < (-\mathrm{k}_{loopstart\_23,13} + \mathrm{i}_{loopend})$

$\rightarrow$ *[from term 23.0, $\mathrm{i}_{loopend}$ is equal to $1 + ((\mathrm{i}_{loopstart\_23,13} + \mathrm{k}_{loopstart\_23,13}) \ / \ 2)$]*

[25.2] $0 < (-\mathrm{k}_{loopstart\_23,13} + (1 + ((\mathrm{i}_{loopstart\_23,13} + \mathrm{k}_{loopstart\_23,13}) \ / \ 2)))$

$\rightarrow$ *[simplify]*

[25.17] $-1 < (-\mathrm{k}_{loopstart\_23,13} + \mathrm{i}_{loopstart\_23,13})$

$\rightarrow$ *[from term 15.11, literala $< (-\mathrm{k}_{loopstart\_23,13} + \mathrm{i}_{loopstart\_23,13})$ is false whenever -2 < (0 + literala)]*

Proof of rule precondition:

[25.17.0] -2 < (-1 + 0)

$\rightarrow$ *[simplify]*

[25.17.2] **true**

[25.18] **false**

Proof of verification condition: Loop body preserves loop invariant

Condition generated at: C:\Program Files\Escher Technologies\Perfect Developer\Examples\Refinement\BinarySearch.pd (36,17)

Condition defined at: C:\Program Files\Escher Technologies\Perfect Developer\Examples\Refinement\BinarySearch.pd (27,23)

To prove: $0 \leq \mathrm{i}_{loopend}$

Given: **self**.members.isndec, i = 0 **as** nat, $0 \leq$ i, k = #**self**.members **as int**, $0 \leq \mathrm{i}_{loopstart\_23,13}$, $0 \leq \mathrm{i}_{loopstart\_23,13}$, $\mathrm{i}_{loopstart\_23,13} \leq \mathrm{k}_{loopstart\_23,13}$, $\mathrm{k}_{loopstart\_23,13} \leq$ #**self**.members, ∀ z ∈ 0 .. <$\mathrm{i}_{loopstart\_23,13}$ • **self**.members[z **as** nat] < x, ∀ z ∈ $\mathrm{k}_{loopstart\_23,13}$ .. <(#**self**.members) • ¬(**self**.members[z **as** nat] < x), ∀ \$x ∈ \$attributeNames(**int**) • **different**($\mathrm{i}_{loopstart\_23,13}$.\$x; $\mathrm{i}_{loopstart\_23,13}$) ⇒ i.\$x=$\mathrm{i}_{loopstart\_23,13}$.\$x, ¬($\mathrm{i}_{loopstart\_23,13} = \mathrm{k}_{loopstart\_23,13}$), 0 ≤ ($\mathrm{k}_{loopstart\_23,13}$ − ($\mathrm{i}_{loopstart\_23,13}$ **as int**)), ($\mathrm{k}_{loopstart\_23,13}$ − ($\mathrm{i}_{loopstart\_23,13}$ **as int**)) ≤ (k − (i **as int**)), p = ($\mathrm{i}_{loopstart\_23,13} + \mathrm{k}_{loopstart\_23,13}$) / 2, (¬(**self**.members[p **as** nat] < x) ∧ ($\mathrm{i}_{loopstart\_23,13} = \mathrm{i}_{loopend}$) ∧ ($\mathrm{k}_{loopend}$ = p)) ∨ (($\mathrm{i}_{loopend}$ = (>p **as** nat)) ∧ ($\mathrm{k}_{loopstart\_23,13} = \mathrm{k}_{loopend}$) ∧ (**self**.members[p **as** nat] < x) ∧ ($0 \leq \mathrm{i}_{loopend}$)), (p < $\mathrm{k}_{loopstart\_23,13}$) ∧ ($\mathrm{i}_{loopstart\_23,13} \leq$ p)

Proof:

[Take goal term]

[1.0] $0 \leq \mathrm{i}_{loopend}$

→ *[simplify]*

[1.2] $-1 < \mathrm{i}_{loopend}$

→ *[negate goal and search for contradiction]*

[1.3] $\neg(-1 < \mathrm{i}_{loopend})$

→ *[simplify]*

[1.5] $0 < -\mathrm{i}_{loopend}$

[Assume known post-assertion, class invariant or type constraint for term 1.5]

[16.0] $0 \leq \mathrm{i}_{loopend}$

→ *[simplify]*

[16.2] $-1 < \mathrm{i}_{loopend}$

→ *[from term 1.5, literala < $\mathrm{i}_{loopend}$ is false whenever -2 < (0 + literala)]*

Proof of rule precondition:

[16.2.0] -2 < (-1 + 0)

→ *[simplify]*

[16.2.2] **true**

[16.3] **false**

Proof of verification condition: Loop body preserves loop invariant

Condition generated at: C:\Program Files\Escher Technologies\Perfect Developer\Examples\Refinement\BinarySearch.pd (36,17)

Condition defined at: C:\Program Files\Escher Technologies\Perfect Developer\Examples\Refinement\BinarySearch.pd (28,24)

To prove: $\mathrm{i}_{loopend} \leq \mathrm{k}_{loopend}$

Given: **self**.members.isndec, i = 0 **as** nat, $0 \leq$ i, k = #**self**.members **as int**, $0 \leq \mathrm{i}_{loopstart\_23,13}$, $0 \leq \mathrm{i}_{loopstart\_23,13}$, $\mathrm{i}_{loopstart\_23,13} \leq \mathrm{k}_{loopstart\_23,13}$, $\mathrm{k}_{loopstart\_23,13} \leq$ #**self**.members, ∀ z ∈ 0 .. <$\mathrm{i}_{loopstart\_23,13}$

• **self**.members[z **as** nat] < x, ∀ z ∈ $k_{loopstart\_23,13}$ .. <(#**self**.members) • ¬(**self**.members[z **as** nat] < x), ∀ $x ∈ $attributeNames(**int**) • **different**($i_{loopstart_23,13}$.$x; $i_{loopstart\_23,13}$) ⇒ i.$x=$i_{loopstart_23,13}$.$x, ¬($i_{loopstart\_23,13} = k_{loopstart\_23,13}$), $0 \leq$ ($k_{loopstart\_23,13}$ − ($i_{loopstart\_23,13}$ **as int**)), ($k_{loopstart\_23,13}$ − ($i_{loopstart\_23,13}$ **as int**)) ≤ (k − (i **as int**)), p = ($i_{loopstart\_23,13} + k_{loopstart\_23,13}$) / 2, (¬(**self**.members[p **as** nat] < x) ∧ ($i_{loopstart\_23,13} = i_{loopend}$) ∧ ($k_{loopend}$ = p)) ∨ (($i_{loopend}$ = (>p **as** nat)) ∧ ($k_{loopstart\_23,13} = k_{loopend}$) ∧ (**self**.members[p **as** nat] < x) ∧ ($0 \leq i_{loopend}$)), (p < $k_{loopstart\_23,13}$) ∧ ($i_{loopstart\_23,13} \leq$ p)

Proof:

[Take goal term]

[1.0] $i_{loopend} \leq k_{loopend}$

→ *[simplify]*

[1.7] -1 < ($-i_{loopend} + k_{loopend}$)

→ *[negate goal and search for contradiction]*

[1.8] ¬(-1 < ($-i_{loopend} + k_{loopend}$))

→ *[simplify]*

[1.12] 0 < ($i_{loopend} + -k_{loopend}$)

[Take given term]

[12.0] (($i_{loopstart\_23,13} + k_{loopstart\_23,13}$) / 2) = p

→ *[simplify]*

[12.1] 0 = (−p + (($i_{loopstart\_23,13} + k_{loopstart\_23,13}$) / 2))

[Take given term]

[14.0] (p < $k_{loopstart\_23,13}$) ∧ ($i_{loopstart\_23,13} \leq$ p)

→ *[simplify]*

[14.8] (0 < (−p + $k_{loopstart\_23,13}$)) ∧ (-1 < ($-i_{loopstart\_23,13}$ + p))

[Work on sub-term 2 of conjunction in term 14.8]

[15.0] 0 < (−p + $k_{loopstart\_23,13}$)

→ *[from term 12.1, p is equal to ($i_{loopstart\_23,13} + k_{loopstart\_23,13}$) / 2]*

[15.1] 0 < (−(($i_{loopstart\_23,13} + k_{loopstart\_23,13}$) / 2) + $k_{loopstart\_23,13}$)

→ *[simplify]*

[15.11] 0 < ($-i_{loopstart\_23,13} + k_{loopstart\_23,13}$)

[Assume known post-assertion, class invariant or type constraint for term 1.12]

[16.0] $0 \leq i_{loopend}$

→ *[simplify]*

[16.2] -1 < $i_{loopend}$

[Take given term]

[13.0] (¬(**self**.members[p **as** nat] < x) ∧ ($i_{loopstart\_23,13} = i_{loopend}$) ∧ ($k_{loopend}$ = p)) ∨ (($i_{loopend}$ = (>p **as** nat)) ∧ ($k_{loopstart\_23,13} = k_{loopend}$) ∧ (**self**.members[p **as** nat] < x) ∧ ($0 \leq i_{loopend}$))

→ *[simplify]*

[13.16] (¬(**self**.members[p] < x) ∧ (0 = ($-i_{loopend} + i_{loopstart\_23,13}$)) ∧ (0 = (−p + $k_{loopend}$))) ∨ ((1 = (−p + $i_{loopend}$)) ∧ (0 = ($-k_{loopend} + k_{loopstart\_23,13}$)) ∧ (**self**.members[p] < x) ∧ (-1 < $i_{loopend}$))

→ *[from term 12.1, p is equal to ($i_{loopstart\_23,13} + k_{loopstart\_23,13}$) / 2]*

[13.20] (¬(**self**.members[($i_{loopstart\_23,13} + k_{loopstart\_23,13}$) / 2] < x) ∧ (0 = (−(($i_{loopstart\_23,13}$ + $k_{loopstart\_23,13}$) / 2) + $k_{loopend}$)) ∧ (0 = ($-i_{loopend} + i_{loopstart\_23,13}$))) ∨ ((0 = ($-k_{loopend}$ +

$k_{loopstart\_23,13})) \wedge (1 = (-((i_{loopstart\_23,13} + k_{loopstart\_23,13}) / 2) + i_{loopend})) \wedge (-1 < i_{loopend}) \wedge$ $(\mathbf{self}.\text{members}[(i_{loopstart\_23,13} + k_{loopstart\_23,13}) / 2] < x))$

→ *[from term 16.2, literala < $i_{loopend}$ is true whenever -1 < (-1 + −literala)]*

Proof of rule precondition:

[13.20.0] $-1 < (-1 + --1)$

→ *[simplify]*

[13.20.3] **true**

[13.21] $(\neg(\mathbf{self}.\text{members}[(i_{loopstart\_23,13} + k_{loopstart\_23,13}) / 2] < x) \wedge (0 = (-((i_{loopstart\_23,13} + k_{loopstart\_23,13}) / 2) + k_{loopend})) \wedge (0 = (-i_{loopend} + i_{loopstart\_23,13}))) \vee ((0 = (-k_{loopend} + k_{loopstart\_23,13})) \wedge (1 = (-((i_{loopstart\_23,13} + k_{loopstart\_23,13}) / 2) + i_{loopend})) \wedge \mathbf{true} \wedge (\mathbf{self}.\text{members}[(i_{loopstart\_23,13} + k_{loopstart\_23,13}) / 2] < x))$

→ *[simplify]*

[13.22] $(\neg(\mathbf{self}.\text{members}[(i_{loopstart\_23,13} + k_{loopstart\_23,13}) / 2] < x) \wedge (0 = (-((i_{loopstart\_23,13} + k_{loopstart\_23,13}) / 2) + k_{loopend})) \wedge (0 = (-i_{loopend} + i_{loopstart\_23,13}))) \vee ((0 = (-k_{loopend} + k_{loopstart\_23,13})) \wedge (1 = (-((i_{loopstart\_23,13} + k_{loopstart\_23,13}) / 2) + i_{loopend})) \wedge (\mathbf{self}.\text{members}[(i_{loopstart\_23,13} + k_{loopstart\_23,13}) / 2] < x))$

Proof branches here giving 2 sub-goals:

Proof of sub-goal 1:

[Branch on disjunction or conditional in term 13.22 and work on branch 1]

[17.0] $\neg(\mathbf{self}.\text{members}[(i_{loopstart\_23,13} + k_{loopstart\_23,13}) / 2] < x) \wedge (0 = (-((i_{loopstart\_23,13} + k_{loopstart\_23,13}) / 2) + k_{loopend})) \wedge (0 = (-i_{loopend} + i_{loopstart\_23,13}))$

[Work on sub-term 2 of conjunction in term 17.0]

[18.0] $0 = (-((i_{loopstart\_23,13} + k_{loopstart\_23,13}) / 2) + k_{loopend})$

[Work on sub-term 3 of conjunction in term 17.0]

[19.0] $0 = (-i_{loopend} + i_{loopstart\_23,13})$

[Copy term 1.12]

[20.0] $0 < (-k_{loopend} + i_{loopend})$

→ *[from term 18.0, $k_{loopend}$ is equal to $(i_{loopstart\_23,13} + k_{loopstart\_23,13}) / 2$]*

[20.1] $0 < (-((i_{loopstart\_23,13} + k_{loopstart\_23,13}) / 2) + i_{loopend})$

→ *[simplify]*

[20.8] $0 < (-i_{loopstart\_23,13} + -k_{loopstart\_23,13} + (2 * i_{loopend}))$

→ *[from term 19.0, $i_{loopend}$ is equal to $i_{loopstart\_23,13}$]*

[20.9] $0 < (-i_{loopstart\_23,13} + -k_{loopstart\_23,13} + (2 * i_{loopstart\_23,13}))$

→ *[simplify]*

[20.12] $0 < (-k_{loopstart\_23,13} + i_{loopstart\_23,13})$

→ *[from term 15.11, literala < $(-k_{loopstart\_23,13} + i_{loopstart\_23,13})$ is false whenever -2 < (0 + literala)]*

Proof of rule precondition:

[20.12.0] $-2 < (0 + 0)$

→ *[simplify]*

[20.12.2] **true**

[20.13] **false**

Proof of sub-goal 2:

[Work on branch 2 from term 13.22]

[21.0] $(0 = (-k_{loopend} + k_{loopstart\_23,13})) \wedge (1 = (-((i_{loopstart\_23,13} + k_{loopstart\_23,13}) \ / \ 2) + i_{loopend})) \wedge$ (**self**.members[$(i_{loopstart\_23,13} + k_{loopstart\_23,13}) \ / \ 2] < x$)

→ *[separate conjunction and work on first sub-term]*

[21.1] $0 = (-k_{loopend} + k_{loopstart\_23,13})$

[Work on sub-term 2 of conjunction in term 21.0]

[22.0] $1 = (-((i_{loopstart\_23,13} + k_{loopstart\_23,13}) \ / \ 2) + i_{loopend})$

[Copy term 1.12]

[24.0] $0 < (-k_{loopend} + i_{loopend})$

→ *[from term 21.1, $k_{loopend}$ is equal to $k_{loopstart\_23,13}$]*

[24.1] $0 < (-k_{loopstart\_23,13} + i_{loopend})$

→ *[from term 22.0, $i_{loopend}$ is equal to 1 + $((i_{loopstart\_23,13} + k_{loopstart\_23,13}) \ / \ 2)$]*

[24.2] $0 < (-k_{loopstart\_23,13} + (1 + ((i_{loopstart\_23,13} + k_{loopstart\_23,13}) \ / \ 2)))$

→ *[simplify]*

[24.17] -1 $< (-k_{loopstart\_23,13} + i_{loopstart\_23,13})$

→ *[from term 15.11, literala $< (-k_{loopstart\_23,13} + i_{loopstart\_23,13})$ is false whenever -2 < (0 + literala)]*

Proof of rule precondition:

[24.17.0] -2 < (-1 + 0)

→ *[simplify]*

[24.17.2] **true**

[24.18] **false**

Proof of verification condition: Loop body preserves loop invariant

Condition generated at: C:\Program Files\Escher Technologies\Perfect Developer\Examples\Refinement\BinarySearch.pd (36,17)

Condition defined at: C:\Program Files\Escher Technologies\Perfect Developer\Examples\Refinement\BinarySearch.pd (29,24)

To prove: $k_{loopend} \leq$ #**self**.members

Given: **self**.members.isndec, i = 0 **as** nat, $0 \leq$ i, k = #**self**.members **as** **int**, $0 \leq i_{loopstart\_23,13}$, $0 \leq i_{loopstart\_23,13}$, $i_{loopstart\_23,13} \leq k_{loopstart\_23,13}$, $k_{loopstart\_23,13} \leq$ #**self**.members, ∀ z ∈ 0 .. <$i_{loopstart\_23,13}$ • **self**.members[z **as** nat] < x, ∀ z ∈ $k_{loopstart\_23,13}$ .. <(#**self**.members) • ¬(**self**.members[z **as** nat] < x), ∀ $x ∈ $attributeNames(**int**) • **different**($i_{loopstart_23,13}$.$x; $i_{loopstart\_23,13}$) ⇒ i.$x=$i_{loopstart_23,13}$.$x, $\neg(i_{loopstart\_23,13} = k_{loopstart\_23,13})$, $0 \leq (k_{loopstart\_23,13} - (i_{loopstart\_23,13}$ **as int**)), $(k_{loopstart\_23,13} - (i_{loopstart\_23,13}$ **as int**)) $\leq$ (k − (i **as int**)), p = $(i_{loopstart\_23,13} + k_{loopstart\_23,13}) \ / \ 2$, (¬(**self**.members[p **as** nat] < x) ∧ $(i_{loopstart_23,13} = i_{loopend}) \wedge (k_{loopend} = p)) \vee ((i_{loopend} = (>$p **as** nat)) ∧ $(k_{loopstart\_23,13} = k_{loopend})$ ∧ (**self**.members[p **as** nat] < x) ∧ $(0 \leq i_{loopend})$), $(p < k_{loopstart\_23,13}) \wedge (i_{loopstart\_23,13} \leq p)$

Proof:

[Take goal term]

[1.0] $k_{loopend} \leq$ #**self**.members

→ *[simplify]*

[1.7] -1 $< (-k_{loopend} +$ #**self**.members)

→ *[negate goal and search for contradiction]*

[1.8] ¬(-1 $< (-k_{loopend} +$ #**self**.members))

$\rightarrow$ *[simplify]*

[1.12] $0 < (\mathrm{k}_{loopend} + -(\#\mathbf{self}.\mathrm{members}))$

[Take given term]

[7.0] $\mathrm{k}_{loopstart\_23,13} \leq \#\mathbf{self}.\mathrm{members}$

$\rightarrow$ *[simplify]*

[7.7] $-1 < (-\mathrm{k}_{loopstart\_23,13} + \#\mathbf{self}.\mathrm{members})$

[Take given term]

[12.0] $((\mathrm{i}_{loopstart\_23,13} + \mathrm{k}_{loopstart\_23,13})\ /\ 2) = \mathrm{p}$

$\rightarrow$ *[simplify]*

[12.1] $0 = (-\mathrm{p} + ((\mathrm{i}_{loopstart\_23,13} + \mathrm{k}_{loopstart\_23,13})\ /\ 2))$

[Take given term]

[13.0] $(\neg(\mathbf{self}.\mathrm{members}[\mathrm{p}\ \mathbf{as}\ \mathrm{nat}] < \mathrm{x}) \wedge (\mathrm{i}_{loopstart\_23,13} = \mathrm{i}_{loopend}) \wedge (\mathrm{k}_{loopend} = \mathrm{p})) \vee ((\mathrm{i}_{loopend} = (>\mathrm{p}\ \mathbf{as}\ \mathrm{nat})) \wedge (\mathrm{k}_{loopstart\_23,13} = \mathrm{k}_{loopend}) \wedge (\mathbf{self}.\mathrm{members}[\mathrm{p}\ \mathbf{as}\ \mathrm{nat}] < \mathrm{x}) \wedge (0 \leq \mathrm{i}_{loopend}))$

$\rightarrow$ *[simplify]*

[13.16] $(\neg(\mathbf{self}.\mathrm{members}[\mathrm{p}] < \mathrm{x}) \wedge (0 = (-\mathrm{i}_{loopend} + \mathrm{i}_{loopstart\_23,13})) \wedge (0 = (-\mathrm{p} + \mathrm{k}_{loopend}))) \vee ((1 = (-\mathrm{p} + \mathrm{i}_{loopend})) \wedge (0 = (-\mathrm{k}_{loopend} + \mathrm{k}_{loopstart\_23,13})) \wedge (\mathbf{self}.\mathrm{members}[\mathrm{p}] < \mathrm{x}) \wedge (-1 < \mathrm{i}_{loopend}))$

$\rightarrow$ *[from term 12.1, p is equal to* $(i_{loopstart\_23,13} + k_{loopstart\_23,13})\ /\ 2$*]*

[13.20] $(\neg(\mathbf{self}.\mathrm{members}[(\mathrm{i}_{loopstart\_23,13} + \mathrm{k}_{loopstart\_23,13})\ /\ 2] < \mathrm{x}) \wedge (0 = (-((\mathrm{i}_{loopstart\_23,13} + \mathrm{k}_{loopstart\_23,13})\ /\ 2) + \mathrm{k}_{loopend})) \wedge (0 = (-\mathrm{i}_{loopend} + \mathrm{i}_{loopstart\_23,13}))) \vee ((0 = (-\mathrm{k}_{loopend} + \mathrm{k}_{loopstart\_23,13})) \wedge (1 = (-((\mathrm{i}_{loopstart\_23,13} + \mathrm{k}_{loopstart\_23,13})\ /\ 2) + \mathrm{i}_{loopend})) \wedge (-1 < \mathrm{i}_{loopend}) \wedge (\mathbf{self}.\mathrm{members}[(\mathrm{i}_{loopstart\_23,13} + \mathrm{k}_{loopstart\_23,13})\ /\ 2] < \mathrm{x}))$

[Take given term]

[14.0] $(\mathrm{p} < \mathrm{k}_{loopstart\_23,13}) \wedge (\mathrm{i}_{loopstart\_23,13} \leq \mathrm{p})$

$\rightarrow$ *[simplify]*

[14.8] $(0 < (-\mathrm{p} + \mathrm{k}_{loopstart\_23,13})) \wedge (-1 < (-\mathrm{i}_{loopstart\_23,13} + \mathrm{p}))$

[Work on sub-term 2 of conjunction in term 14.8]

[15.0] $0 < (-\mathrm{p} + \mathrm{k}_{loopstart\_23,13})$

$\rightarrow$ *[from term 12.1, p is equal to* $(i_{loopstart\_23,13} + k_{loopstart\_23,13})\ /\ 2$*]*

[15.1] $0 < (-((\mathrm{i}_{loopstart\_23,13} + \mathrm{k}_{loopstart\_23,13})\ /\ 2) + \mathrm{k}_{loopstart\_23,13})$

$\rightarrow$ *[simplify]*

[15.11] $0 < (-\mathrm{i}_{loopstart\_23,13} + \mathrm{k}_{loopstart\_23,13})$

[Create new term from terms 1.12, 7.7 using rule: transitivity 1]

[66.0] $(-1 + 0 + 1) < (-\mathrm{k}_{loopstart\_23,13} + \mathrm{k}_{loopend})$

$\rightarrow$ *[simplify]*

[66.1] $0 < (-\mathrm{k}_{loopstart\_23,13} + \mathrm{k}_{loopend})$

[Create new term from terms 7.7, 15.11 using rule: transitivity 1]

[124.0] $(-1 + 0 + 1) < (-\mathrm{i}_{loopstart\_23,13} + \#\mathbf{self}.\mathrm{members})$

$\rightarrow$ *[simplify]*

[124.1] $0 < (-\mathrm{i}_{loopstart\_23,13} + \#\mathbf{self}.\mathrm{members})$

Proof branches here giving 2 sub-goals:

Proof of sub-goal 1:

[Branch on disjunction or conditional in term 13.20 and work on branch 1]

[16.0] ¬(**self**.members[($i_{loopstart\_23,13}$ + $k_{loopstart\_23,13}$) / 2] < x) ∧ (0 = (−(($i_{loopstart\_23,13}$ + $k_{loopstart\_23,13}$) / 2) + $k_{loopend}$)) ∧ (0 = (−$i_{loopend}$ + $i_{loopstart\_23,13}$))

[Work on sub-term 2 of conjunction in term 16.0]

[18.0] 0 = (−(($i_{loopstart\_23,13}$ + $k_{loopstart\_23,13}$) / 2) + $k_{loopend}$)

[Copy term 1.12]

[20.0] 0 < (−(#**self**.members) + $k_{loopend}$)

→ *[from term 18.0, $k_{loopend}$ is equal to ($i_{loopstart\_23,13}$ + $k_{loopstart\_23,13}$) / 2]*

[20.1] 0 < (−(#**self**.members) + (($i_{loopstart\_23,13}$ + $k_{loopstart\_23,13}$) / 2))

→ *[simplify]*

[20.9] 1 < ((-2 * #**self**.members) + $i_{loopstart\_23,13}$ + $k_{loopstart\_23,13}$)

[Create new term from terms 7.7, 20.9 using rule: transitivity 1]

[120.0] (-1 + 1 + 1) < (((-2 * #**self**.members) + $i_{loopstart\_23,13}$) + #**self**.members)

→ *[simplify]*

[120.5] 1 < (−(#**self**.members) + $i_{loopstart\_23,13}$)

→ *[from term 124.1, literala < (−(#**self**.members) + $i_{loopstart\_23,13}$) is false whenever -2 < (0 + literala)]*

Proof of rule precondition:

[120.5.0] -2 < (0 + 1)

→ *[simplify]*

[120.5.2] **true**

[120.6] **false**

Proof of sub-goal 2:

[Work on branch 2 from term 13.20]

[130.0] (0 = (−$k_{loopend}$ + $k_{loopstart\_23,13}$)) ∧ (1 = (−(($i_{loopstart\_23,13}$ + $k_{loopstart\_23,13}$) / 2) + $i_{loopend}$)) ∧ (-1 < $i_{loopend}$) ∧ (**self**.members[($i_{loopstart\_23,13}$ + $k_{loopstart\_23,13}$) / 2] < x)

→ *[from term 66.1, (−$k_{loopend}$ + $k_{loopstart\_23,13}$) = literala is false whenever -1 < (0 + literala)]*

Proof of rule precondition:

[130.0.0] -1 < (0 + 0)

→ *[simplify]*

[130.0.2] **true**

[130.1] **false** ∧ (1 = (−(($i_{loopstart\_23,13}$ + $k_{loopstart\_23,13}$) / 2) + $i_{loopend}$)) ∧ (-1 < $i_{loopend}$) ∧ (**self**.members[($i_{loopstart\_23,13}$ + $k_{loopstart\_23,13}$) / 2] < x)

→ *[simplify]*

[130.2] **false**

Proof of verification condition: Loop body preserves loop invariant

Condition generated at: C:\Program Files\Escher Technologies\Perfect Developer\Examples\Refinement\BinarySearch.pd (36,17)

Condition defined at: C:\Program Files\Escher Technologies\Perfect Developer\Examples\Refinement\BinarySearch.pd (30,21)

To prove: ∀ z ∈ 0 .. <$i_{loopend}$ • **self**.members[z **as** nat] < x

Given: **self**.members.isndec, i = 0 **as** nat, $0 \leq$ i, k = #**self**.members **as int**, $0 \leq i_{loopstart\_23,13}$, $0 \leq i_{loopstart\_23,13}$, $i_{loopstart\_23,13} \leq k_{loopstart\_23,13}$, $k_{loopstart\_23,13} \leq$ #**self**.members, $\forall$ z $\in$ 0 .. <$i_{loopstart\_23,13}$ $\bullet$ **self**.members[z **as** nat] < x, $\forall$ z $\in$ $k_{loopstart\_23,13}$ .. <(#**self**.members) $\bullet$ $\neg$(**self**.members[z **as** nat] < x), $\forall$ $x $\in$ $attributeNames(**int**) $\bullet$ **different**($i_{loopstart\_23,13}$.$x; $i_{loopstart\_23,13}$) $\Rightarrow$ i.$x=$i_{loopstart_23,13}$.$x, $\neg(i_{loopstart\_23,13} = k_{loopstart\_23,13})$, $0 \leq (k_{loopstart\_23,13} - (i_{loopstart\_23,13}$ **as int**)), $(k_{loopstart\_23,13} - (i_{loopstart\_23,13}$ **as int**)) $\leq$ (k $-$ (i **as int**)), p = $(i_{loopstart\_23,13} + k_{loopstart\_23,13})$ / 2, ($\neg$(**self**.members[p **as** nat] < x) $\wedge$ $(i_{loopstart\_23,13} = i_{loopend})$ $\wedge$ $(k_{loopend} = p)$) $\vee$ (($i_{loopend}$ = (>p **as** nat)) $\wedge$ $(k_{loopstart\_23,13} = k_{loopend})$ $\wedge$ (**self**.members[p **as** nat] < x) $\wedge$ $(0 \leq i_{loopend})$), $(p < k_{loopstart\_23,13}) \wedge (i_{loopstart\_23,13} \leq p)$

Proof:

[Take goal term]

[1.0] $\forall$ z $\in$ 0 .. <$i_{loopend}$ $\bullet$ **self**.members[z **as** nat] < x

$\rightarrow$ *[simplify]*

[1.3] $\forall$ z $\in$ (0 .. (-1 + $i_{loopend}$)).ran $\bullet$ **self**.members[z] < x

$\rightarrow$ *[negate goal and search for contradiction]*

[1.4] $\exists$ z $\in$ (0 .. (-1 + $i_{loopend}$)).ran $\bullet$ $\neg$(**self**.members[z] < x)

$\rightarrow$ *[introduce skolem term and eliminate 'exists']*

[1.5] $\neg$(**self**.members[$a_z] < x)

[Take given term]

[2.0] **self**.members.isndec

[Take given term]

[8.0] $\forall$ z $\in$ 0 .. <$i_{loopstart\_23,13}$ $\bullet$ **self**.members[z **as** nat] < x

$\rightarrow$ *[simplify]*

[8.3] $\forall$ z $\in$ (0 .. (-1 + $i_{loopstart\_23,13}$)).ran $\bullet$ **self**.members[z] < x

$\rightarrow$ *[introduce metavariable and eliminate 'forall']*

[8.4] **self**.members[?b] < x

[Take given term]

[12.0] $((i_{loopstart\_23,13} + k_{loopstart\_23,13}) / 2) = p$

$\rightarrow$ *[simplify]*

[12.1] $0 = (-p + ((i_{loopstart\_23,13} + k_{loopstart\_23,13}) / 2))$

[Create new term from bound when replacing existential quantifier in term 1.4]

[16.0] $a_z **in** (0 .. (-1 + $i_{loopend}$)).ran

$\rightarrow$ *[simplify]*

[16.6] (0 < ($-$$a_z + $i_{loopend}$)) $\wedge$ (-1 < $a_z)

$\rightarrow$ *[separate conjunction and work on first sub-term]*

[16.7] -1 < $a_z

[Assume known post-assertion, class invariant or type constraint for term 16.6]

[17.0] $0 \leq i_{loopend}$

$\rightarrow$ *[simplify]*

[17.2] $-1 < i_{loopend}$

[Take given term]

[13.0] ($\neg$(**self**.members[p **as** nat] < x) $\wedge$ $(i_{loopstart\_23,13} = i_{loopend})$ $\wedge$ $(k_{loopend} = p)$) $\vee$ (($i_{loopend}$ = (>p **as** nat)) $\wedge$ $(k_{loopstart\_23,13} = k_{loopend})$ $\wedge$ (**self**.members[p **as** nat] < x) $\wedge$ $(0 \leq i_{loopend})$)

→ *[simplify]*

[13.16] $(\neg(\textbf{self}.\text{members}[p] < x) \wedge (0 = (-i_{loopend} + i_{loopstart\_23,13})) \wedge (0 = (-p + k_{loopend}))) \vee ((1 = (-p + i_{loopend})) \wedge (0 = (-k_{loopend} + k_{loopstart\_23,13})) \wedge (\textbf{self}.\text{members}[p] < x) \wedge (-1 < i_{loopend}))$

→ *[from term 12.1, p is equal to* $(i_{loopstart\_23,13} + k_{loopstart\_23,13})\ /\ 2$*]*

[13.20] $(\neg(\textbf{self}.\text{members}[(i_{loopstart\_23,13} + k_{loopstart\_23,13})\ /\ 2] < x) \wedge (0 = (-((i_{loopstart\_23,13} + k_{loopstart\_23,13})\ /\ 2) + k_{loopend})) \wedge (0 = (-i_{loopend} + i_{loopstart\_23,13}))) \vee ((0 = (-k_{loopend} + k_{loopstart\_23,13})) \wedge (1 = (-((i_{loopstart\_23,13} + k_{loopstart\_23,13})\ /\ 2) + i_{loopend})) \wedge (-1 < i_{loopend}) \wedge (\textbf{self}.\text{members}[(i_{loopstart\_23,13} + k_{loopstart\_23,13})\ /\ 2] < x))$

→ *[from term 17.2, literala* $< i_{loopend}$ *is true whenever -1 < (-1 + −literala)]*

Proof of rule precondition:

[13.20.0] $-1 < (-1 + --1)$

→ *[simplify]*

[13.20.3] **true**

[13.21] $(\neg(\textbf{self}.\text{members}[(i_{loopstart\_23,13} + k_{loopstart\_23,13})\ /\ 2] < x) \wedge (0 = (-((i_{loopstart\_23,13} + k_{loopstart\_23,13})\ /\ 2) + k_{loopend})) \wedge (0 = (-i_{loopend} + i_{loopstart\_23,13}))) \vee ((0 = (-k_{loopend} + k_{loopstart\_23,13})) \wedge (1 = (-((i_{loopstart\_23,13} + k_{loopstart\_23,13})\ /\ 2) + i_{loopend})) \wedge \textbf{true} \wedge (\textbf{self}.\text{members}[(i_{loopstart\_23,13} + k_{loopstart\_23,13})\ /\ 2] < x))$

→ *[simplify]*

[13.22] $(\neg(\textbf{self}.\text{members}[(i_{loopstart\_23,13} + k_{loopstart\_23,13})\ /\ 2] < x) \wedge (0 = (-((i_{loopstart\_23,13} + k_{loopstart\_23,13})\ /\ 2) + k_{loopend})) \wedge (0 = (-i_{loopend} + i_{loopstart\_23,13}))) \vee ((0 = (-k_{loopend} + k_{loopstart\_23,13})) \wedge (1 = (-((i_{loopstart\_23,13} + k_{loopstart\_23,13})\ /\ 2) + i_{loopend})) \wedge (\textbf{self}.\text{members}[(i_{loopstart\_23,13} + k_{loopstart\_23,13})\ /\ 2] < x))$

[Work on sub-term 2 of conjunction in term 16.6]

[18.0] $0 < (-\$a\_z + i_{loopend})$

[Apply unification ?b → \$a_z to term 8.4]

[25.0] **self**.members[\$a_z] < x

→ *[from term 1.5,* **self**.*members[\$a_z] < x is false]*

[25.1] **false**

Proof branches here giving 2 sub-goals:

Proof of sub-goal 1:

[Branch on disjunction or conditional in term 13.22 and work on branch 1]

[19.0] $\neg(\textbf{self}.\text{members}[(i_{loopstart\_23,13} + k_{loopstart\_23,13})\ /\ 2] < x) \wedge (0 = (-((i_{loopstart\_23,13} + k_{loopstart\_23,13})\ /\ 2) + k_{loopend})) \wedge (0 = (-i_{loopend} + i_{loopstart\_23,13}))$

[Work on sub-term 3 of conjunction in term 19.0]

[21.0] $0 = (-i_{loopend} + i_{loopstart\_23,13})$

[Copy term 18.0]

[22.0] $0 < (-\$a\_z + i_{loopend})$

→ *[from term 21.0,* $i_{loopend}$ *is equal to* $i_{loopstart\_23,13}$*]*

[22.1] $0 < (-\$a\_z + i_{loopstart\_23,13})$

[Work on branch 2 from term 8.3]

[27.0] $\neg(\$a\_z\ \textbf{in}\ (0\ ..\ (-1 + i_{loopstart\_23,13})).\text{ran})$

→ *[simplify]*

[27.6] $\neg((0 < (-\$a\_z + i_{loopstart\_23,13})) \wedge (-1 < \$a\_z))$

$\rightarrow$ *[from term 16.7, literala < \$a_z is true whenever -1 < (-1 + −literala)]*

Proof of rule precondition:

[27.6.0] -1 < (-1 + −-1)

$\rightarrow$ *[simplify]*

[27.6.3] **true**

[27.7] $\neg((0 < (-\$a\_z + i_{loopstart\_23,13})) \wedge$ **true**)

$\rightarrow$ *[simplify]*

[27.12] $-1 < (\$a\_z + -i_{loopstart\_23,13})$

[Copy term 22.1]

[28.0] $0 < (-\$a\_z + i_{loopstart\_23,13})$

$\rightarrow$ *[from term 27.12, literala <* $(-\$a\_z + i_{loopstart\_23,13})$ *is false whenever -2 < (-1 + literala)]*

Proof of rule precondition:

[28.0.0] -2 < (-1 + 0)

$\rightarrow$ *[simplify]*

[28.0.2] **true**

[28.1] **false**

Proof of sub-goal 2:

[Work on branch 2 from term 13.22]

[29.0] $(0 = (-k_{loopend} + k_{loopstart\_23,13})) \wedge (1 = (-((i_{loopstart\_23,13} + k_{loopstart\_23,13}) / 2) + i_{loopend})) \wedge$ (**self**.members$[(i_{loopstart_23,13} + k_{loopstart_23,13}) / 2] < x)$

[Work on sub-term 2 of conjunction in term 29.0]

[30.0] $1 = (-((i_{loopstart\_23,13} + k_{loopstart\_23,13}) / 2) + i_{loopend})$

[Work on sub-term 3 of conjunction in term 29.0]

[31.0] **self**.members$[(i_{loopstart_23,13} + k_{loopstart_23,13}) / 2] < x$

[Copy term 18.0]

[33.0] $0 < (-\$a\_z + i_{loopend})$

$\rightarrow$ *[from term 30.0,* $i_{loopend}$ *is equal to* $1 + ((i_{loopstart\_23,13} + k_{loopstart\_23,13}) / 2)$*]*

[33.1] $0 < (-\$a\_z + (1 + ((i_{loopstart\_23,13} + k_{loopstart\_23,13}) / 2)))$

$\rightarrow$ *[simplify]*

[33.13] $-1 < ((-2 * \$a\_z) + i_{loopstart\_23,13} + k_{loopstart\_23,13})$

[Create new term from terms 1.5, 31.0 using rule: transitivity]

[69.0] **self**.members$[(i_{loopstart_23,13} + k_{loopstart_23,13}) / 2] <$ **self**.members[\$a_z]

[Create new term from terms 2.0, 69.0 using rule: compare elements of non-decreasing sequence]

[96.0] $((i_{loopstart\_23,13} + k_{loopstart\_23,13}) / 2) < \$a\_z$

$\rightarrow$ *[simplify]*

[96.8] $0 < (-i_{loopstart\_23,13} + -k_{loopstart\_23,13} + (2 * \$a\_z))$

$\rightarrow$ *[from term 33.13, literala <* $(-i_{loopstart\_23,13} + -k_{loopstart\_23,13} + (2 * \$a\_z))$ *is false whenever -2 < (-1 + literala)]*

Proof of rule precondition:

[96.8.0] -2 < (-1 + 0)

$\rightarrow$ *[simplify]*

[96.8.2] **true**

[96.9] **false**

Proof of verification condition: Loop body preserves loop invariant

Condition generated at: C:\Program Files\Escher Technologies\Perfect Developer\Examples\Refinement\BinarySearch.pd (36,17)

Condition defined at: C:\Program Files\Escher Technologies\Perfect Developer\Examples\Refinement\BinarySearch.pd (31,21)

To prove: $\forall$ z $\in$ $k_{loopend}$ .. <(#**self**.members) $\bullet$ $\neg$(**self**.members[z **as** nat] < x)

Given: **self**.members.isndec, i = 0 **as** nat, 0 $\leq$ i, k = #**self**.members **as** **int**, 0 $\leq$ $i_{loopstart\_23,13}$, 0 $\leq$ $i_{loopstart\_23,13}$, $i_{loopstart\_23,13} \leq k_{loopstart\_23,13}$, $k_{loopstart\_23,13} \leq$ #**self**.members, $\forall$ z $\in$ 0 .. <$i_{loopstart\_23,13}$ $\bullet$ **self**.members[z **as** nat] < x, $\forall$ z $\in$ $k_{loopstart\_23,13}$ .. <(#**self**.members) $\bullet$ $\neg$(**self**.members[z **as** nat] < x), $\forall$ $x $\in$ $attributeNames(**int**) $\bullet$ **different**($i_{loopstart\_23,13}$.$x; $i_{loopstart\_23,13}$) $\Rightarrow$ i.$x=$i_{loopstart_23,13}$.$x, $\neg$($i_{loopstart\_23,13} = k_{loopstart\_23,13}$), 0 $\leq$ ($k_{loopstart\_23,13}$ $-$ ($i_{loopstart\_23,13}$ **as int**)), ($k_{loopstart\_23,13}$ $-$ ($i_{loopstart\_23,13}$ **as int**)) $\leq$ (k $-$ (i **as int**)), p = ($i_{loopstart\_23,13} + k_{loopstart\_23,13}$) / 2, ($\neg$(**self**.members[p **as** nat] < x) $\wedge$ ($i_{loopstart\_23,13} = i_{loopend}$) $\wedge$ ($k_{loopend}$ = p)) $\vee$ (($i_{loopend}$ = (>p **as** nat)) $\wedge$ ($k_{loopstart\_23,13} = k_{loopend}$) $\wedge$ (**self**.members[p **as** nat] < x) $\wedge$ (0 $\leq$ $i_{loopend}$)), (p < $k_{loopstart\_23,13}$) $\wedge$ ($i_{loopstart\_23,13} \leq$ p)

Proof:

[Take goal term]

[1.0] $\forall$ z $\in$ $k_{loopend}$ .. <(#**self**.members) $\bullet$ $\neg$(**self**.members[z **as** nat] < x)

$\rightarrow$ *[simplify]*

[1.3] $\forall$ z $\in$ ($k_{loopend}$ .. (-1 + #**self**.members)).ran $\bullet$ $\neg$(**self**.members[z] < x)

$\rightarrow$ *[negate goal and search for contradiction]*

[1.4] $\exists$ z $\in$ ($k_{loopend}$ .. (-1 + #**self**.members)).ran $\bullet$ **self**.members[z] < x

$\rightarrow$ *[introduce skolem term and eliminate 'exists']*

[1.5] **self**.members[$a_z] < x

[Take given term]

[2.0] **self**.members.isndec

[Take given term]

[9.0] $\forall$ z $\in$ $k_{loopstart\_23,13}$ .. <(#**self**.members) $\bullet$ $\neg$(**self**.members[z **as** nat] < x)

$\rightarrow$ *[simplify]*

[9.3] $\forall$ z $\in$ ($k_{loopstart\_23,13}$ .. (-1 + #**self**.members)).ran $\bullet$ $\neg$(**self**.members[z] < x)

$\rightarrow$ *[introduce metavariable and eliminate 'forall']*

[9.4] $\neg$(**self**.members[?c] < x)

[Take given term]

[12.0] (($i_{loopstart\_23,13} + k_{loopstart\_23,13}$) / 2) = p

$\rightarrow$ *[simplify]*

[12.1] 0 = ($-$p + (($i_{loopstart\_23,13} + k_{loopstart\_23,13}$) / 2))

[Take given term]

[13.0] ($\neg$(**self**.members[p **as** nat] < x) $\wedge$ ($i_{loopstart\_23,13} = i_{loopend}$) $\wedge$ ($k_{loopend}$ = p)) $\vee$ (($i_{loopend}$ = (>p **as** nat)) $\wedge$ ($k_{loopstart\_23,13} = k_{loopend}$) $\wedge$ (**self**.members[p **as** nat] < x) $\wedge$ (0 $\leq$ $i_{loopend}$))

$\rightarrow$ *[simplify]*

[13.16] (¬(**self**.members[p] < x) ∧ (0 = (−$i_{loopend}$ + $i_{loopstart\_23,13}$)) ∧ (0 = (−p + $k_{loopend}$))) ∨ ((1 = (−p + $i_{loopend}$)) ∧ (0 = (−$k_{loopend}$ + $k_{loopstart\_23,13}$)) ∧ (**self**.members[p] < x) ∧ (-1 < $i_{loopend}$))

→ *[from term 12.1, p is equal to ($i_{loopstart\_23,13}$ + $k_{loopstart\_23,13}$) / 2]*

[13.20] (¬(**self**.members[($i_{loopstart\_23,13}$ + $k_{loopstart\_23,13}$) / 2] < x) ∧ (0 = (−(($i_{loopstart\_23,13}$ + $k_{loopstart\_23,13}$) / 2) + $k_{loopend}$)) ∧ (0 = (−$i_{loopend}$ + $i_{loopstart\_23,13}$))) ∨ ((0 = (−$k_{loopend}$ + $k_{loopstart\_23,13}$)) ∧ (1 = (−(($i_{loopstart\_23,13}$ + $k_{loopstart\_23,13}$) / 2) + $i_{loopend}$)) ∧ (-1 < $i_{loopend}$) ∧ (**self**.members[($i_{loopstart\_23,13}$ + $k_{loopstart\_23,13}$) / 2] < x))

[Create new term from bound when replacing existential quantifier in term 1.4]

[16.0] $a_z **in** ($k_{loopend}$ .. (-1 + #**self**.members)).ran

→ *[simplify]*

[16.4] (0 < (−$a_z + #**self**.members)) ∧ (-1 < (−$k_{loopend}$ + $a_z))

→ *[separate conjunction and work on first sub-term]*

[16.5] -1 < (−$k_{loopend}$ + $a_z)

[Work on sub-term 2 of conjunction in term 16.4]

[17.0] 0 < (−$a_z + #**self**.members)

[Apply unification ?c → $a_z to term 9.4]

[28.0] ¬(**self**.members[$a_z] < x)

→ *[from term 1.5,* **self***.members[$a_z] < x is true]*

[28.1] ¬**true**

→ *[simplify]*

[28.2] **false**

Proof branches here giving 2 sub-goals:

Proof of sub-goal 1:

[Branch on disjunction or conditional in term 13.20 and work on branch 1]

[18.0] ¬(**self**.members[($i_{loopstart\_23,13}$ + $k_{loopstart\_23,13}$) / 2] < x) ∧ (0 = (−(($i_{loopstart\_23,13}$ + $k_{loopstart\_23,13}$) / 2) + $k_{loopend}$)) ∧ (0 = (−$i_{loopend}$ + $i_{loopstart\_23,13}$))

→ *[separate conjunction and work on first sub-term]*

[18.1] ¬(**self**.members[($i_{loopstart\_23,13}$ + $k_{loopstart\_23,13}$) / 2] < x)

[Work on sub-term 2 of conjunction in term 18.0]

[20.0] 0 = (−(($i_{loopstart\_23,13}$ + $k_{loopstart\_23,13}$) / 2) + $k_{loopend}$)

[Copy term 16.5]

[22.0] -1 < (−$k_{loopend}$ + $a_z)

→ *[from term 20.0, $k_{loopend}$ is equal to ($i_{loopstart\_23,13}$ + $k_{loopstart\_23,13}$) / 2]*

[22.1] -1 < (−(($i_{loopstart\_23,13}$ + $k_{loopstart\_23,13}$) / 2) + $a_z)

→ *[simplify]*

[22.12] -2 < (−$i_{loopstart\_23,13}$ + −$k_{loopstart\_23,13}$ + (2 * $a_z))

[Create new term from terms 1.5, 18.1 using rule: transitivity]

[50.0] **self**.members[$a_z] < **self**.members[($i_{loopstart_23,13}$ + $k_{loopstart\_23,13}$) / 2]

[Create new term from terms 2.0, 50.0 using rule: compare elements of non-decreasing sequence]

[64.0] $a_z < (($i_{loopstart_23,13}$ + $k_{loopstart\_23,13}$) / 2)

→ *[simplify]*

[64.9] $1 < ((-2 * \$a\_z) + i_{loopstart\_23,13} + k_{loopstart\_23,13})$

→ *[from term 22.12, literala < ((-2 * $a_z) + $i_{loopstart\_23,13}$ + $k_{loopstart\_23,13}$) is false whenever -2 < (-2 + literala)]*

Proof of rule precondition:

[64.9.0] -2 < (-2 + 1)

→ *[simplify]*

[64.9.2] **true**

[64.10] **false**

Proof of sub-goal 2:

[Work on branch 2 from term 13.20]

[72.0] $(0 = (-k_{loopend} + k_{loopstart\_23,13})) \wedge (1 = (-((i_{loopstart\_23,13} + k_{loopstart\_23,13}) / 2) + i_{loopend})) \wedge (-1 < i_{loopend}) \wedge (\mathbf{self}.\text{members}[(i_{loopstart\_23,13} + k_{loopstart\_23,13}) / 2] < x)$

→ *[separate conjunction and work on first sub-term]*

[72.1] $0 = (-k_{loopend} + k_{loopstart\_23,13})$

[Work on branch 2 from term 9.3]

[32.0] ¬($a_z **in** ($k_{loopstart_23,13}$ .. (-1 + #**self**.members)).ran)

→ *[simplify]*

[32.4] ¬((0 < (−$a_z + #**self**.members)) ∧ (-1 < ($-k_{loopstart_23,13}$ + $a_z)))

→ *[from term 17.0, literala < (−$a_z + #**self**.members) is true whenever -1 < (0 + −literala)]*

Proof of rule precondition:

[32.4.0] -1 < (0 + −0)

→ *[simplify]*

[32.4.3] **true**

[32.5] ¬(**true** ∧ (-1 < ($-k_{loopstart\_23,13}$ + $a_z)))

→ *[simplify]*

[32.10] 0 < ($k_{loopstart\_23,13}$ + −$a_z)

[Copy term 16.5]

[97.0] -1 < ($-k_{loopend}$ + $a_z)

→ *[from term 72.1, $k_{loopend}$ is equal to $k_{loopstart\_23,13}$]*

[97.1] -1 < ($-k_{loopstart\_23,13}$ + $a_z)

→ *[from term 32.10, literala < ($-k_{loopstart\_23,13}$ + $a_z) is false whenever -2 < (0 + literala)]*

Proof of rule precondition:

[97.1.0] -2 < (-1 + 0)

→ *[simplify]*

[97.1.2] **true**

[97.2] **false**

Proof of verification condition: Loop body only modifies objects in 'change' list

Condition generated at: C:\Program Files\Escher Technologies\Perfect Developer\Examples\Refinement\BinarySearch.pd (36,17)

Condition defined at: C:\Program Files\Escher Technologies\Perfect Developer\Examples\Refinement\BinarySearch.pd (25,24)

To prove: $\forall\ \$x \in \$attributeNames(\mathbf{int}) \bullet \mathbf{different}(i_{loopend}.\$x;\ i_{loopend}) \Rightarrow i_{loopstart\_23,13}.\$x = i_{loopend}.\$x$

Given: $\mathbf{self}.members.isndec,\ i = 0\ \mathbf{as}\ nat,\ 0 \leq i,\ k = \#\mathbf{self}.members\ \mathbf{as}\ \mathbf{int},\ 0 \leq i_{loopstart\_23,13},\ 0 \leq i_{loopstart\_23,13},\ i_{loopstart\_23,13} \leq k_{loopstart\_23,13},\ k_{loopstart\_23,13} \leq \#\mathbf{self}.members,\ \forall\ z \in 0\ ..\ <i_{loopstart\_23,13} \bullet \mathbf{self}.members[z\ \mathbf{as}\ nat] < x,\ \forall\ z \in k_{loopstart\_23,13}\ ..\ <(\#\mathbf{self}.members) \bullet \neg(\mathbf{self}.members[z\ \mathbf{as}\ nat] < x),\ \forall\ \$x \in \$attributeNames(\mathbf{int}) \bullet \mathbf{different}(i_{loopstart\_23,13}.\$x;\ i_{loopstart\_23,13}) \Rightarrow i.\$x = i_{loopstart\_23,13}.\$x,\ \neg(i_{loopstart\_23,13} = k_{loopstart\_23,13}),\ 0 \leq (k_{loopstart\_23,13} - (i_{loopstart\_23,13}\ \mathbf{as}\ \mathbf{int})),\ (k_{loopstart\_23,13} - (i_{loopstart\_23,13}\ \mathbf{as}\ \mathbf{int})) \leq (k - (i\ \mathbf{as}\ \mathbf{int})),\ p = (i_{loopstart\_23,13} + k_{loopstart\_23,13})\ /\ 2,\ (\neg(\mathbf{self}.members[p\ \mathbf{as}\ nat] < x) \wedge (i_{loopstart\_23,13} = i_{loopend}) \wedge (k_{loopend} = p)) \vee ((i_{loopend} = (>p\ \mathbf{as}\ nat)) \wedge (k_{loopstart\_23,13} = k_{loopend}) \wedge (\mathbf{self}.members[p\ \mathbf{as}\ nat] < x) \wedge (0 \leq i_{loopend})),\ (p < k_{loopstart\_23,13}) \wedge (i_{loopstart\_23,13} \leq p)$

Proof:

[Take goal term]

[1.0] $\forall\ \$x \in \$attributeNames(\mathbf{int}) \bullet \mathbf{different}(i_{loopend}.\$x;\ i_{loopend}) \Rightarrow i_{loopstart\_23,13}.\$x = i_{loopend}.\$x$

$\rightarrow$ *[simplify]*

[1.1] **true**

Proof of verification condition: Precondition of '/' satisfied

Condition generated at: C:\Program Files\Escher Technologies\Perfect Developer\Examples\Refinement\BinarySearch.pd (36,33)

Condition defined at: built in declaration

To prove: $0 < 2$

Given: $\mathbf{self}.members.isndec,\ i = 0\ \mathbf{as}\ nat,\ 0 \leq i,\ k = \#\mathbf{self}.members\ \mathbf{as}\ \mathbf{int},\ 0 \leq i_{loopstart\_23,13},\ 0 \leq i_{loopstart\_23,13},\ i_{loopstart\_23,13} \leq k_{loopstart\_23,13},\ k_{loopstart\_23,13} \leq \#\mathbf{self}.members,\ \forall\ z \in 0\ ..\ <i_{loopstart\_23,13} \bullet \mathbf{self}.members[z\ \mathbf{as}\ nat] < x,\ \forall\ z \in k_{loopstart\_23,13}\ ..\ <(\#\mathbf{self}.members) \bullet \neg(\mathbf{self}.members[z\ \mathbf{as}\ nat] < x),\ \forall\ \$x \in \$attributeNames(\mathbf{int}) \bullet \mathbf{different}(i_{loopstart\_23,13}.\$x;\ i_{loopstart\_23,13}) \Rightarrow i.\$x = i_{loopstart\_23,13}.\$x,\ \neg(i_{loopstart\_23,13} = k_{loopstart\_23,13}),\ 0 \leq (k_{loopstart\_23,13} - (i_{loopstart\_23,13}\ \mathbf{as}\ \mathbf{int})),\ (k_{loopstart\_23,13} - (i_{loopstart\_23,13}\ \mathbf{as}\ \mathbf{int})) \leq (k - (i\ \mathbf{as}\ \mathbf{int}))$

Proof:

[Take goal term]

[1.0] $0 < 2$

$\rightarrow$ *[simplify]*

[1.1] **true**

Proof of verification condition: Assertion valid

Condition generated at: C:\Program Files\Escher Technologies\Perfect Developer\Examples\Refinement\BinarySearch.pd (37,26)

To prove: $(p < k_{loopstart\_23,13}) \wedge (i_{loopstart\_23,13} \leq p)$

Given: $\mathbf{self}.members.isndec,\ i = 0\ \mathbf{as}\ nat,\ 0 \leq i,\ k = \#\mathbf{self}.members\ \mathbf{as}\ \mathbf{int},\ 0 \leq i_{loopstart\_23,13},\ 0 \leq i_{loopstart\_23,13},\ i_{loopstart\_23,13} \leq k_{loopstart\_23,13},\ k_{loopstart\_23,13} \leq \#\mathbf{self}.members,\ \forall\ z \in 0\ ..\ <i_{loopstart\_23,13} \bullet \mathbf{self}.members[z\ \mathbf{as}\ nat] < x,\ \forall\ z \in k_{loopstart\_23,13}\ ..\ <(\#\mathbf{self}.members) \bullet \neg(\mathbf{self}.members[z\ \mathbf{as}\ nat] < x),\ \forall\ \$x \in \$attributeNames(\mathbf{int}) \bullet \mathbf{different}(i_{loopstart\_23,13}.\$x;\ i_{loopstart\_23,13}) \Rightarrow i.\$x = i_{loopstart\_23,13}.\$x,\ \neg(i_{loopstart\_23,13} = k_{loopstart\_23,13}),\ 0 \leq (k_{loopstart\_23,13} - (i_{loopstart\_23,13}\ \mathbf{as}\ \mathbf{int})),\ (k_{loopstart\_23,13} - (i_{loopstart\_23,13}\ \mathbf{as}\ \mathbf{int})) \leq (k - (i\ \mathbf{as}\ \mathbf{int})),\ p = (i_{loopstart\_23,13} + k_{loopstart\_23,13})\ /\ 2$

Proof:

[Take given term]

[6.0] $\mathrm{i}_{loopstart\_23,13} \leq \mathrm{k}_{loopstart\_23,13}$

$\rightarrow$ *[simplify]*

[6.7] -1 $< (-\mathrm{i}_{loopstart\_23,13} + \mathrm{k}_{loopstart\_23,13})$

[Take given term]

[12.0] $((\mathrm{i}_{loopstart\_23,13} + \mathrm{k}_{loopstart\_23,13}) / 2) = \mathrm{p}$

$\rightarrow$ *[simplify]*

[12.1] $0 = (-\mathrm{p} + ((\mathrm{i}_{loopstart\_23,13} + \mathrm{k}_{loopstart\_23,13}) / 2))$

[Take goal term]

[1.0] $(\mathrm{p} < \mathrm{k}_{loopstart\_23,13}) \wedge (\mathrm{i}_{loopstart\_23,13} \leq \mathrm{p})$

$\rightarrow$ *[from term 12.1, p is equal to ($i_{loopstart\_23,13} + k_{loopstart\_23,13}$) / 2]*

[1.1] $(((\mathrm{i}_{loopstart\_23,13} + \mathrm{k}_{loopstart\_23,13}) / 2) < \mathrm{k}_{loopstart\_23,13}) \wedge (\mathrm{i}_{loopstart\_23,13} \leq \mathrm{p})$

$\rightarrow$ *[simplify]*

[1.12] $(0 < (-\mathrm{i}_{loopstart\_23,13} + \mathrm{k}_{loopstart\_23,13})) \wedge (\mathrm{i}_{loopstart\_23,13} \leq \mathrm{p})$

$\rightarrow$ *[from term 12.1, p is equal to ($i_{loopstart\_23,13} + k_{loopstart\_23,13}$) / 2]*

[1.13] $(0 < (-\mathrm{i}_{loopstart\_23,13} + \mathrm{k}_{loopstart\_23,13})) \wedge (\mathrm{i}_{loopstart\_23,13} \leq ((\mathrm{i}_{loopstart\_23,13} + \mathrm{k}_{loopstart\_23,13}) / 2))$

$\rightarrow$ *[simplify]*

[1.32] $(0 < (-\mathrm{i}_{loopstart\_23,13} + \mathrm{k}_{loopstart\_23,13})) \wedge (-1 < (-\mathrm{i}_{loopstart\_23,13} + \mathrm{k}_{loopstart\_23,13}))$

$\rightarrow$ *[from other term in conjunction, literala < ($-i_{loopstart\_23,13} + k_{loopstart\_23,13}$) is true whenever -1 < (0 + $-$literala)]*

Proof of rule precondition:

[1.32.0] -1 < (0 + −-1)

$\rightarrow$ *[simplify]*

[1.32.3] **true**

[1.33] $(0 < (-\mathrm{i}_{loopstart\_23,13} + \mathrm{k}_{loopstart\_23,13})) \wedge$ **true**

$\rightarrow$ *[simplify]*

[1.34] $0 < (-\mathrm{i}_{loopstart\_23,13} + \mathrm{k}_{loopstart\_23,13})$

$\rightarrow$ *[negate goal and search for contradiction]*

[1.35] $\neg(0 < (-\mathrm{i}_{loopstart\_23,13} + \mathrm{k}_{loopstart\_23,13}))$

$\rightarrow$ *[simplify]*

[1.39] $-1 < (\mathrm{i}_{loopstart\_23,13} + -\mathrm{k}_{loopstart\_23,13})$

$\rightarrow$ *[from term 6.7, -1 < ($-k_{loopstart\_23,13} + i_{loopstart\_23,13}$) is true if and only if 0 = ($-i_{loopstart\_23,13} + k_{loopstart\_23,13}$)]*

[1.40] $0 = (-\mathrm{i}_{loopstart\_23,13} + \mathrm{k}_{loopstart\_23,13})$

[Take given term]

[10.0] $\neg(\mathrm{i}_{loopstart\_23,13} = \mathrm{k}_{loopstart\_23,13})$

$\rightarrow$ *[simplify]*

[10.1] $\neg(0 = (-\mathrm{k}_{loopstart\_23,13} + \mathrm{i}_{loopstart\_23,13}))$

$\rightarrow$ *[from term 1.40, $-k_{loopstart\_23,13} + i_{loopstart\_23,13}$ is equal to 0]*

[10.2] $\neg(0 = 0)$

$\rightarrow$ *[simplify]*

[10.4] **false**

Proof of verification condition: Type constraint satisfied

Condition generated at: C:\Program Files\Escher Technologies\Perfect Developer\Examples\Refinement\BinarySearch.pd (40,26)

Condition defined at: built in declaration

To prove: $0 \leq >$p

Given: **self**.members.isndec, i = 0 **as** nat, $0 \leq$ i, k = #**self**.members **as** **int**, $0 \leq i_{loopstart\_23,13}$, $0 \leq i_{loopstart\_23,13}$, $i_{loopstart\_23,13} \leq k_{loopstart\_23,13}$, $k_{loopstart\_23,13} \leq$ #**self**.members, $\forall$ z $\in$ 0 .. $< i_{loopstart\_23,13}$ • **self**.members[z **as** nat] < x, $\forall$ z $\in k_{loopstart\_23,13}$ .. <(#**self**.members) • ¬(**self**.members[z **as** nat] < x), $\forall$ \$x $\in$ \$attributeNames(**int**) • **different**($i_{loopstart\_23,13}$.\$x; $i_{loopstart\_23,13}$) $\Rightarrow$ i.\$x=$i_{loopstart\_23,13}$.\$x, ¬($i_{loopstart\_23,13} = k_{loopstart\_23,13}$), $0 \leq$ ($k_{loopstart\_23,13}$ − ($i_{loopstart\_23,13}$ **as int**)), ($k_{loopstart\_23,13}$ − ($i_{loopstart\_23,13}$ **as int**)) $\leq$ (k − (i **as int**)), p = ($i_{loopstart\_23,13} + k_{loopstart\_23,13}$) / 2, (p < $k_{loopstart\_23,13}$) ∧ ($i_{loopstart\_23,13} \leq$ p), **self**.members[p **as** nat] < x

Proof:

[Take given term]

[12.0] (($i_{loopstart\_23,13} + k_{loopstart\_23,13}$) / 2) = p

→ *[simplify]*

[12.1] 0 = (−p + (($i_{loopstart\_23,13} + k_{loopstart\_23,13}$) / 2))

[Take goal term]

[1.0] $0 \leq >$p

→ *[from term 12.1, p is equal to ($i_{loopstart\_23,13} + k_{loopstart\_23,13}$) / 2]*

[1.1] $0 \leq >$(($i_{loopstart\_23,13} + k_{loopstart\_23,13}$) / 2)

→ *[simplify]*

[1.12] -3 < ($i_{loopstart\_23,13} + k_{loopstart\_23,13}$)

→ *[negate goal and search for contradiction]*

[1.13] ¬(-3 < ($i_{loopstart\_23,13} + k_{loopstart\_23,13}$))

→ *[simplify]*

[1.16] 2 < ($-i_{loopstart\_23,13} + -k_{loopstart\_23,13}$)

[Take given term]

[13.0] (p < $k_{loopstart\_23,13}$) ∧ ($i_{loopstart\_23,13} \leq$ p)

→ *[from term 12.1, p is equal to ($i_{loopstart\_23,13} + k_{loopstart\_23,13}$) / 2]*

[13.1] ((($i_{loopstart\_23,13} + k_{loopstart\_23,13}$) / 2) < $k_{loopstart\_23,13}$) ∧ ($i_{loopstart\_23,13} \leq$ p)

→ *[simplify]*

[13.12] (0 < ($-i_{loopstart\_23,13} + k_{loopstart\_23,13}$)) ∧ ($i_{loopstart\_23,13} \leq$ p)

→ *[from term 12.1, p is equal to ($i_{loopstart\_23,13} + k_{loopstart\_23,13}$) / 2]*

[13.13] (0 < ($-i_{loopstart\_23,13} + k_{loopstart\_23,13}$)) ∧ ($i_{loopstart\_23,13} \leq$ (($i_{loopstart\_23,13} + k_{loopstart\_23,13}$) / 2))

→ *[simplify]*

[13.32] (0 < ($-i_{loopstart\_23,13} + k_{loopstart\_23,13}$)) ∧ (-1 < ($-i_{loopstart\_23,13} + k_{loopstart\_23,13}$))

→ *[from other term in conjunction, literala < ($-i_{loopstart\_23,13} + k_{loopstart\_23,13}$) is true whenever -1 < (0 + −literala)]*

Proof of rule precondition:

[13.32.0] -1 < (0 + −-1)

→ *[simplify]*

[13.32.3] **true**

[13.33] $(0 < (-\mathrm{i}_{loopstart\_23,13} + \mathrm{k}_{loopstart\_23,13})) \wedge$ **true**

→ *[simplify]*

[13.34] $0 < (-\mathrm{i}_{loopstart\_23,13} + \mathrm{k}_{loopstart\_23,13})$

[Create new term from terms 1.16, 13.34 using rule: transitivity 1]

[55.0] $(0 + 1 + 2) < (-\mathrm{i}_{loopstart\_23,13} + -\mathrm{i}_{loopstart\_23,13})$

→ *[simplify]*

[55.5] ([-2 < 0]: $(3 / 2) < -\mathrm{i}_{loopstart\_23,13}$, [0 < -2]: $(3 / \text{-}2) < \mathrm{i}_{loopstart\_23,13}$, [-2 = 0]: 3 < 0)

→ *[explicitly assert falsehood of skipped guards in subsequent guards]*

[55.6] ([-2 < 0]: $(3 / 2) < -\mathrm{i}_{loopstart\_23,13}$, [¬(-2 < 0) ∧ (0 < -2)]: $(3 / \text{-}2) < \mathrm{i}_{loopstart\_23,13}$, [¬(-2 < 0) ∧ ¬(0 < -2) ∧ (-2 = 0)]: 3 < 0)

→ *[simplify]*

[55.9] $1 < -\mathrm{i}_{loopstart\_23,13}$

[Take given term]

[5.0] $0 \leq \mathrm{i}_{loopstart\_23,13}$

→ *[simplify]*

[5.2] $\text{-}1 < \mathrm{i}_{loopstart\_23,13}$

→ *[from term 55.9, literala < $\mathrm{i}_{loopstart\_23,13}$ is false whenever -2 < (1 + literala)]*

Proof of rule precondition:

[5.2.0] -2 < (-1 + 1)

→ *[simplify]*

[5.2.2] **true**

[5.3] **false**

Proof of verification condition: Type constraint satisfied

Condition generated at: C:\Program Files\Escher Technologies\Perfect Developer\Examples\Refinement\BinarySearch.pd (39,30)

Condition defined at: built in declaration

To prove: $0 \leq \mathrm{p}$

Given: **self**.members.isndec, i = 0 **as** nat, 0 ≤ i, k = #**self**.members **as** **int**, $0 \leq \mathrm{i}_{loopstart\_23,13}$, $0 \leq \mathrm{i}_{loopstart\_23,13}$, $\mathrm{i}_{loopstart\_23,13} \leq \mathrm{k}_{loopstart\_23,13}$, $\mathrm{k}_{loopstart\_23,13} \leq$ #**self**.members, ∀ z ∈ 0 .. <$\mathrm{i}_{loopstart\_23,13}$ • **self**.members[z **as** nat] < x, ∀ z ∈ $\mathrm{k}_{loopstart\_23,13}$ .. <(#**self**.members) • ¬(**self**.members[z **as** nat] < x), ∀ $x ∈ $attributeNames(**int**) • **different**($\mathrm{i}_{loopstart_23,13}$.$x; $\mathrm{i}_{loopstart\_23,13}$) ⇒ i.$x=$\mathrm{i}_{loopstart_23,13}$.$x, ¬($\mathrm{i}_{loopstart_23,13} = \mathrm{k}_{loopstart_23,13}$), 0 ≤ ($\mathrm{k}_{loopstart_23,13}$ − ($\mathrm{i}_{loopstart\_23,13}$ **as int**)), ($\mathrm{k}_{loopstart\_23,13}$ − ($\mathrm{i}_{loopstart\_23,13}$ **as int**)) ≤ (k − (i **as int**)), p = ($\mathrm{i}_{loopstart\_23,13} + \mathrm{k}_{loopstart\_23,13}$) / 2, (p < $\mathrm{k}_{loopstart\_23,13}$) ∧ ($\mathrm{i}_{loopstart\_23,13} \leq$ p)

Proof:

[Take given term]

[12.0] $((\mathrm{i}_{loopstart\_23,13} + \mathrm{k}_{loopstart\_23,13}) / 2) = \mathrm{p}$

→ *[simplify]*

[12.1] $0 = (-\mathrm{p} + ((\mathrm{i}_{loopstart\_23,13} + \mathrm{k}_{loopstart\_23,13}) / 2))$

[Take goal term]

[1.0] $0 \leq \mathrm{p}$

$\rightarrow$ *[from term 12.1, p is equal to* $(i_{loopstart\_23,13} + k_{loopstart\_23,13})$ */ 2]*

[1.1] $0 \leq ((\mathrm{i}_{loopstart\_23,13} + \mathrm{k}_{loopstart\_23,13})\ /\ 2)$

$\rightarrow$ *[simplify]*

[1.9] $-1 < (\mathrm{i}_{loopstart\_23,13} + \mathrm{k}_{loopstart\_23,13})$

$\rightarrow$ *[negate goal and search for contradiction]*

[1.10] $\neg(-1 < (\mathrm{i}_{loopstart\_23,13} + \mathrm{k}_{loopstart\_23,13}))$

$\rightarrow$ *[simplify]*

[1.13] $0 < (-\mathrm{i}_{loopstart\_23,13} + -\mathrm{k}_{loopstart\_23,13})$

[Take given term]

[13.0] $(\mathrm{p} < \mathrm{k}_{loopstart\_23,13}) \wedge (\mathrm{i}_{loopstart\_23,13} \leq \mathrm{p})$

$\rightarrow$ *[from term 12.1, p is equal to* $(i_{loopstart\_23,13} + k_{loopstart\_23,13})$ */ 2]*

[13.1] $(((\mathrm{i}_{loopstart\_23,13} + \mathrm{k}_{loopstart\_23,13})\ /\ 2) < \mathrm{k}_{loopstart\_23,13}) \wedge (\mathrm{i}_{loopstart\_23,13} \leq \mathrm{p})$

$\rightarrow$ *[simplify]*

[13.12] $(0 < (-\mathrm{i}_{loopstart\_23,13} + \mathrm{k}_{loopstart\_23,13})) \wedge (\mathrm{i}_{loopstart\_23,13} \leq \mathrm{p})$

$\rightarrow$ *[from term 12.1, p is equal to* $(i_{loopstart\_23,13} + k_{loopstart\_23,13})$ */ 2]*

[13.13] $(0 < (-\mathrm{i}_{loopstart\_23,13} + \mathrm{k}_{loopstart\_23,13})) \wedge (\mathrm{i}_{loopstart\_23,13} \leq ((\mathrm{i}_{loopstart\_23,13} + \mathrm{k}_{loopstart\_23,13})\ /\ 2))$

$\rightarrow$ *[simplify]*

[13.32] $(0 < (-\mathrm{i}_{loopstart\_23,13} + \mathrm{k}_{loopstart\_23,13})) \wedge (-1 < (-\mathrm{i}_{loopstart\_23,13} + \mathrm{k}_{loopstart\_23,13}))$

$\rightarrow$ *[from other term in conjunction, literala <* $(-i_{loopstart\_23,13} + k_{loopstart\_23,13})$ *is true whenever -1 < (0 +* $-$*literala)]*

Proof of rule precondition:

[13.32.0] $-1 < (0 + --1)$

$\rightarrow$ *[simplify]*

[13.32.3] **true**

[13.33] $(0 < (-\mathrm{i}_{loopstart\_23,13} + \mathrm{k}_{loopstart\_23,13})) \wedge$ **true**

$\rightarrow$ *[simplify]*

[13.34] $0 < (-\mathrm{i}_{loopstart\_23,13} + \mathrm{k}_{loopstart\_23,13})$

[Create new term from terms 1.13, 13.34 using rule: transitivity 1]

[51.0] $(0 + 0 + 1) < (-\mathrm{i}_{loopstart\_23,13} + -\mathrm{i}_{loopstart\_23,13})$

$\rightarrow$ *[simplify]*

[51.5] ([-2 < 0]: $(1\ /\ 2) < -\mathrm{i}_{loopstart\_23,13}$, [0 < -2]: $(1\ /\ {-2}) < \mathrm{i}_{loopstart\_23,13}$, [-2 = 0]: $1 < 0$)

$\rightarrow$ *[explicitly assert falsehood of skipped guards in subsequent guards]*

[51.6] ([-2 < 0]: $(1\ /\ 2) < -\mathrm{i}_{loopstart\_23,13}$, [¬(-2 < 0) ∧ (0 < -2)]: $(1\ /\ {-2}) < \mathrm{i}_{loopstart\_23,13}$, [¬(-2 < 0) ∧ ¬(0 < -2) ∧ (-2 = 0)]: $1 < 0$)

$\rightarrow$ *[simplify]*

[51.9] $0 < -\mathrm{i}_{loopstart\_23,13}$

[Take given term]

[5.0] $0 \leq \mathrm{i}_{loopstart\_23,13}$

$\rightarrow$ *[simplify]*

[5.2] $-1 < \mathrm{i}_{loopstart\_23,13}$

→ *[from term 51.9, literala < $i_{loopstart\_23,13}$ is false whenever -2 < (0 + literala)]*

Proof of rule precondition:

[5.2.0] -2 < (-1 + 0)

→ *[simplify]*

[5.2.2] **true**

[5.3] **false**

Proof of verification condition: Precondition of '[]' satisfied

Condition generated at: C:\Program Files\Escher Technologies\Perfect Developer\Examples\Refinement\BinarySearch.pd (39,29)

Condition defined at: built in declaration

To prove: (p **as** nat) < #**self**.members

Given: **self**.members.isndec, i = 0 **as** nat, $0 \leq$ i, k = #**self**.members **as int**, $0 \leq i_{loopstart\_23,13}$, $0 \leq i_{loopstart\_23,13}$, $i_{loopstart\_23,13} \leq k_{loopstart\_23,13}$, $k_{loopstart\_23,13} \leq$ #**self**.members, ∀ z ∈ 0 .. <$i_{loopstart\_23,13}$ • **self**.members[z **as** nat] < x, ∀ z ∈ $k_{loopstart\_23,13}$ .. <(#**self**.members) • ¬(**self**.members[z **as** nat] < x), ∀ \$x ∈ \$attributeNames(**int**) • **different**($i_{loopstart\_23,13}$.\$x; $i_{loopstart\_23,13}$) ⇒ i.\$x=$i_{loopstart\_23,13}$.\$x, ¬($i_{loopstart\_23,13} = k_{loopstart\_23,13}$), $0 \leq$ ($k_{loopstart\_23,13}$ − ($i_{loopstart\_23,13}$ **as int**)), ($k_{loopstart\_23,13}$ − ($i_{loopstart\_23,13}$ **as int**)) ≤ (k − (i **as int**)), p = ($i_{loopstart\_23,13} + k_{loopstart\_23,13}$) / 2, (p < $k_{loopstart\_23,13}$) ∧ ($i_{loopstart\_23,13} \leq$ p)

Proof:

[Take given term]

[7.0] $k_{loopstart\_23,13} \leq$ #**self**.members

→ *[simplify]*

[7.7] -1 < ($-k_{loopstart\_23,13}$ + #**self**.members)

[Take given term]

[12.0] (($i_{loopstart\_23,13} + k_{loopstart\_23,13}$) / 2) = p

→ *[simplify]*

[12.1] 0 = (−p + (($i_{loopstart\_23,13} + k_{loopstart\_23,13}$) / 2))

[Take goal term]

[1.0] (p **as** nat) < #**self**.members

→ *[from term 12.1, p is equal to ($i_{loopstart\_23,13} + k_{loopstart\_23,13}$) / 2]*

[1.1] (($i_{loopstart\_23,13} + k_{loopstart\_23,13}$) / 2 **as** nat) < #**self**.members

→ *[simplify]*

[1.10] 0 < ($-i_{loopstart\_23,13} + -k_{loopstart\_23,13}$ + (2 * #**self**.members))

→ *[negate goal and search for contradiction]*

[1.11] ¬(0 < ($-i_{loopstart\_23,13} + -k_{loopstart\_23,13}$ + (2 * #**self**.members)))

→ *[simplify]*

[1.19] -1 < ((-2 * #**self**.members) + $i_{loopstart\_23,13} + k_{loopstart\_23,13}$)

[Take given term]

[13.0] (p < $k_{loopstart\_23,13}$) ∧ ($i_{loopstart\_23,13} \leq$ p)

→ *[from term 12.1, p is equal to ($i_{loopstart\_23,13} + k_{loopstart\_23,13}$) / 2]*

[13.1] ((($i_{loopstart\_23,13} + k_{loopstart\_23,13}$) / 2) < $k_{loopstart\_23,13}$) ∧ ($i_{loopstart\_23,13} \leq$ p)

→ *[simplify]*

[13.12] $(0 < (-\mathrm{i}_{loopstart\_23,13} + \mathrm{k}_{loopstart\_23,13})) \wedge (\mathrm{i}_{loopstart\_23,13} \leq \mathrm{p})$

→ *[from term 12.1, p is equal to* $(\mathrm{i}_{loopstart\_23,13} + \mathrm{k}_{loopstart\_23,13})$ */ 2]*

[13.13] $(0 < (-\mathrm{i}_{loopstart\_23,13} + \mathrm{k}_{loopstart\_23,13})) \wedge (\mathrm{i}_{loopstart\_23,13} \leq ((\mathrm{i}_{loopstart\_23,13} + \mathrm{k}_{loopstart\_23,13})\ /\ 2))$

→ *[simplify]*

[13.32] $(0 < (-\mathrm{i}_{loopstart\_23,13} + \mathrm{k}_{loopstart\_23,13})) \wedge (-1 < (-\mathrm{i}_{loopstart\_23,13} + \mathrm{k}_{loopstart\_23,13}))$

→ *[from other term in conjunction, literala <* $(-\mathrm{i}_{loopstart\_23,13} + \mathrm{k}_{loopstart\_23,13})$ *is true whenever -1 < (0 + −literala)]*

Proof of rule precondition:

[13.32.0] -1 < (0 + −-1)

→ *[simplify]*

[13.32.3] **true**

[13.33] $(0 < (-\mathrm{i}_{loopstart\_23,13} + \mathrm{k}_{loopstart\_23,13})) \wedge$ **true**

→ *[simplify]*

[13.34] $0 < (-\mathrm{i}_{loopstart\_23,13} + \mathrm{k}_{loopstart\_23,13})$

[Create new term from terms 1.19, 13.34 using rule: transitivity 1]

[24.0] (-1 + 0 + 1) < (((-2 * #**self**.members) + $\mathrm{k}_{loopstart\_23,13}$) + $\mathrm{k}_{loopstart\_23,13}$)

→ *[simplify]*

[24.12] 0 < (−(#**self**.members) + $\mathrm{k}_{loopstart\_23,13}$)

→ *[from term 7.7, literala < (−(#**self**.members) +* $\mathrm{k}_{loopstart\_23,13}$*) is false whenever -2 < (-1 + literala)]*

Proof of rule precondition:

[24.12.0] -2 < (-1 + 0)

→ *[simplify]*

[24.12.2] **true**

[24.13] **false**

Proof of verification condition: Type constraint satisfied

Condition generated at: C:\Program Files\Escher Technologies\Perfect Developer\Examples\Refinement\BinarySearch.pd (31,62)

Condition defined at: built in declaration

To prove: $0 \leq z$

Given: **self**.members.isndec, i = 0 **as** nat, $0 \leq \mathrm{i}$, k = #**self**.members **as** **int**, $0 \leq \mathrm{i}_{loopstart\_23,13}$, $0 \leq \mathrm{i}_{loopstart\_23,13}$, $\mathrm{i}_{loopstart\_23,13} \leq \mathrm{k}_{loopstart\_23,13}$, $\mathrm{k}_{loopstart\_23,13} \leq$ #**self**.members, ∀ z ∈ 0 .. <$\mathrm{i}_{loopstart\_23,13}$ • **self**.members[z **as** nat] < x, z **in** ($\mathrm{k}_{loopstart\_23,13}$ .. <(#**self**.members))

Proof:

[Take goal term]

[1.0] $0 \leq z$

→ *[simplify]*

[1.2] -1 < z

→ *[negate goal and search for contradiction]*

[1.3] ¬(-1 < z)

→ *[simplify]*

[1.5] $0 < -\mathrm{z}$

[Take given term]

[5.0] $0 \leq \mathrm{i}_{loopstart\_23,13}$

→ *[simplify]*

[5.2] $-1 < \mathrm{i}_{loopstart\_23,13}$

[Take given term]

[6.0] $\mathrm{i}_{loopstart\_23,13} \leq \mathrm{k}_{loopstart\_23,13}$

→ *[simplify]*

[6.7] $-1 < (-\mathrm{i}_{loopstart\_23,13} + \mathrm{k}_{loopstart\_23,13})$

[Take given term]

[9.0] z **in** ($\mathrm{k}_{loopstart\_23,13}$ .. <(#**self**.members))

→ *[simplify]*

[9.5] (0 < (−z + #**self**.members)) ∧ (-1 < ($-\mathrm{k}_{loopstart\_23,13}$ + z))

→ *[separate conjunction and work on first sub-term]*

[9.6] $-1 < (-\mathrm{k}_{loopstart\_23,13} + \mathrm{z})$

[Create new term from terms 1.5, 9.6 using rule: transitivity 3]

[13.0] $(-1 + 0 + 1) < -\mathrm{k}_{loopstart\_23,13}$

→ *[simplify]*

[13.1] $0 < -\mathrm{k}_{loopstart\_23,13}$

[Create new term from terms 6.7, 13.1 using rule: transitivity 2]

[16.0] $(-1 + 0 + 1) < -\mathrm{i}_{loopstart\_23,13}$

→ *[simplify]*

[16.1] $0 < -\mathrm{i}_{loopstart\_23,13}$

→ *[from term 5.2, literala < $-\mathrm{i}_{loopstart\_23,13}$ is false whenever -2 < (-1 + literala)]*

Proof of rule precondition:

[16.1.0] -2 < (-1 + 0)

→ *[simplify]*

[16.1.2] **true**

[16.2] **false**

Proof of verification condition: Precondition of '[]' satisfied

Condition generated at: C:\Program Files\Escher Technologies\Perfect Developer\Examples\Refinement\BinarySearch.pd (31,61)

Condition defined at: built in declaration

To prove: (z **as** nat) < #**self**.members

Given: **self**.members.isndec, i = 0 **as** nat, 0 ≤ i, k = #**self**.members **as int**, 0 ≤ $\mathrm{i}_{loopstart\_23,13}$, 0 ≤ $\mathrm{i}_{loopstart\_23,13}$, $\mathrm{i}_{loopstart\_23,13} \leq \mathrm{k}_{loopstart\_23,13}$, $\mathrm{k}_{loopstart\_23,13}$ ≤ #**self**.members, ∀ z ∈ 0 .. <$\mathrm{i}_{loopstart\_23,13}$ • **self**.members[z **as** nat] < x, z **in** ($\mathrm{k}_{loopstart\_23,13}$ .. <(#**self**.members))

Proof:

[Take given term]

[9.0] z **in** ($k_{loopstart\_23,13}$ .. <(#**self**.members))

→ *[simplify]*

[9.5] (0 < (−z + #**self**.members)) ∧ (-1 < ($-k_{loopstart\_23,13}$ + z))

[Work on sub-term 2 of conjunction in term 9.5]

[10.0] 0 < (−z + #**self**.members)

[Take goal term]

[1.0] (z **as** nat) < #**self**.members

→ *[simplify]*

[1.2] 0 < (−z + #**self**.members)

→ *[from term 10.0, literala < (−z + #**self**.members) is true whenever -1 < (0 + −literala)]*

Proof of rule precondition:

[1.2.0] -1 < (0 + −0)

→ *[simplify]*

[1.2.3] **true**

[1.3] **true**

Proof of verification condition: Type constraint satisfied

Condition generated at: C:\Program Files\Escher Technologies\Perfect Developer\Examples\Refinement\BinarySearch.pd (30,53)

Condition defined at: built in declaration

To prove: 0 ≤ z

Given: **self**.members.isndec, i = 0 **as** nat, 0 ≤ i, k = #**self**.members **as int**, 0 ≤ $i_{loopstart\_23,13}$, 0 ≤ $i_{loopstart\_23,13}$, $i_{loopstart\_23,13}$ ≤ $k_{loopstart\_23,13}$, $k_{loopstart\_23,13}$ ≤ #**self**.members, z **in** (0 .. <$i_{loopstart\_23,13}$)

Proof:

[Take given term]

[8.0] z **in** (0 .. <$i_{loopstart\_23,13}$)

→ *[simplify]*

[8.7] (0 < (−z + $i_{loopstart\_23,13}$)) ∧ (-1 < z)

→ *[separate conjunction and work on first sub-term]*

[8.8] -1 < z

[Take goal term]

[1.0] 0 ≤ z

→ *[simplify]*

[1.2] -1 < z

→ *[from term 8.8, literala < z is true whenever -1 < (-1 + −literala)]*

Proof of rule precondition:

[1.2.0] -1 < (-1 + −-1)

→ *[simplify]*

[1.2.3] **true**

[1.3] **true**

Proof of verification condition: Precondition of '[]' satisfied

Condition generated at: C:\Program Files\Escher Technologies\Perfect Developer\Examples\Refinement\BinarySearch.pd (30,52)

Condition defined at: built in declaration

To prove: (z **as** nat) < #**self**.members

Given: **self**.members.isndec, i = 0 **as** nat, $0 \le$ i, k = #**self**.members **as int**, $0 \le i_{loopstart\_23,13}$, $0 \le i_{loopstart\_23,13}$, $i_{loopstart\_23,13} \le k_{loopstart\_23,13}$, $k_{loopstart\_23,13} \le$ #**self**.members, z **in** (0 .. <$i_{loopstart\_23,13}$)

Proof:

[Take goal term]

[1.0] (z **as** nat) < #**self**.members

→ *[simplify]*

[1.2] 0 < (−z + #**self**.members)

→ *[negate goal and search for contradiction]*

[1.3] ¬(0 < (−z + #**self**.members))

→ *[simplify]*

[1.7] -1 < (z + −(#**self**.members))

[Take given term]

[6.0] $i_{loopstart\_23,13} \le k_{loopstart\_23,13}$

→ *[simplify]*

[6.7] -1 < ($-i_{loopstart\_23,13} + k_{loopstart\_23,13}$)

[Take given term]

[7.0] $k_{loopstart\_23,13} \le$ #**self**.members

→ *[simplify]*

[7.7] -1 < ($-k_{loopstart\_23,13}$ + #**self**.members)

[Take given term]

[8.0] z **in** (0 .. <$i_{loopstart\_23,13}$)

→ *[simplify]*

[8.7] (0 < ($-z + i_{loopstart\_23,13}$)) ∧ (-1 < z)

[Work on sub-term 2 of conjunction in term 8.7]

[9.0] 0 < ($-z + i_{loopstart\_23,13}$)

[Create new term from terms 1.7, 9.0 using rule: transitivity 1]

[11.0] (-1 + 0 + 1) < (−(#**self**.members) + $i_{loopstart\_23,13}$)

→ *[simplify]*

[11.1] 0 < (−(#**self**.members) + $i_{loopstart\_23,13}$)

[Create new term from terms 6.7, 11.1 using rule: transitivity 1]

[15.0] (-1 + 0 + 1) < (−(#**self**.members) + $k_{loopstart\_23,13}$)

→ *[simplify]*

[15.1] 0 < (−(#**self**.members) + $k_{loopstart\_23,13}$)

→ *[from term 7.7, literala < (−(#**self**.members) + $k_{loopstart\_23,13}$) is false whenever -2 < (-1 + literala)]*

Proof of rule precondition:

[15.1.0] -2 < (-1 + 0)

→ *[simplify]*

[15.1.2] **true**

[15.2] **false**

Proof of verification condition: Loop initialisation establishes loop invariant

Condition generated at: C:\Program Files\Escher Technologies\Perfect Developer\Examples\Refinement\BinarySearch.pd (23,13)

Condition defined at: C:\Program Files\Escher Technologies\Perfect Developer\Examples\Refinement\BinarySearch.pd (27,23)

To prove: $0 \leq$ i

Given: **self**.members.isndec, i = 0 **as** nat, $0 \leq$ i, k = #**self**.members **as int**

Proof:

[Take given term]

[3.0] i = (0 **as** nat)

→ *[simplify]*

[3.1] i = 0

[Take goal term]

[1.0] $0 \leq$ i

→ *[from term 3.1, i is equal to 0]*

[1.1] $0 \leq 0$

→ *[simplify]*

[1.2] **true**

Proof of verification condition: Loop initialisation establishes loop invariant

Condition generated at: C:\Program Files\Escher Technologies\Perfect Developer\Examples\Refinement\BinarySearch.pd (23,13)

Condition defined at: C:\Program Files\Escher Technologies\Perfect Developer\Examples\Refinement\BinarySearch.pd (28,24)

To prove: i $\leq$ k

Given: **self**.members.isndec, i = 0 **as** nat, $0 \leq$ i, k = #**self**.members **as int**

Proof:

[Take given term]

[3.0] i = (0 **as** nat)

→ *[simplify]*

[3.1] i = 0

[Take given term]

[4.0] k = (#**self**.members **as int**)

→ *[simplify]*

[4.2] 0 = (−k + #**self**.members)

[Take goal term]

[1.0] i $\leq$ k

→ *[from term 3.1, i is equal to 0]*

[1.1] $0 \leq$ k

$\rightarrow$ *[from term 4.2, k is equal to #**self**.members]*

[1.2] $0 \leq$ #**self**.members

$\rightarrow$ *[simplify]*

Proof of rule precondition:

[1.4.0] -1 < 0

$\rightarrow$ *[simplify]*

[1.4.1] **true**

[1.5] **true**

Proof of verification condition: Loop initialisation establishes loop invariant

Condition generated at: C:\Program Files\Escher Technologies\Perfect Developer\Examples\Refinement\BinarySearch.pd (23,13)

Condition defined at: C:\Program Files\Escher Technologies\Perfect Developer\Examples\Refinement\BinarySearch.pd (29,24)

To prove: k $\leq$ #**self**.members

Given: **self**.members.isndec, i = 0 **as** nat, $0 \leq$ i, k = #**self**.members **as int**

Proof:

[Take given term]

[4.0] k = (#**self**.members **as int**)

$\rightarrow$ *[simplify]*

[4.2] 0 = ($-$k + #**self**.members)

[Take goal term]

[1.0] k $\leq$ #**self**.members

$\rightarrow$ *[from term 4.2, k is equal to #**self**.members]*

[1.1] #**self**.members $\leq$ #**self**.members

$\rightarrow$ *[simplify]*

[1.9] **true**

Proof of verification condition: Loop initialisation establishes loop invariant

Condition generated at: C:\Program Files\Escher Technologies\Perfect Developer\Examples\Refinement\BinarySearch.pd (23,13)

Condition defined at: C:\Program Files\Escher Technologies\Perfect Developer\Examples\Refinement\BinarySearch.pd (30,21)

To prove: $\forall$ z $\in$ 0 .. <i $\bullet$ **self**.members[z **as** nat] < x

Given: **self**.members.isndec, i = 0 **as** nat, $0 \leq$ i, k = #**self**.members **as int**

Proof:

[Take given term]

[3.0] i = (0 **as** nat)

$\rightarrow$ *[simplify]*

[3.1] i = 0

[Take goal term]

[1.0] $\forall$ z $\in$ 0 .. <i • **self**.members[z **as** nat] < x

$\rightarrow$ *[from term 3.1, i is equal to 0]*

[1.1] $\forall$ z $\in$ 0 .. <0 • **self**.members[z **as** nat] < x

$\rightarrow$ *[simplify]*

[1.5] $\forall$ z $\in$ **seq of int**{} • **self**.members[z] < x

$\rightarrow$ *[expand literal quantifier]*

[1.6] **true**

Proof of verification condition: Loop initialisation establishes loop invariant

Condition generated at: C:\Program Files\Escher Technologies\Perfect Developer\Examples\Refinement\BinarySearch.pd (23,13)

Condition defined at: C:\Program Files\Escher Technologies\Perfect Developer\Examples\Refinement\BinarySearch.pd (31,21)

To prove: $\forall$ z $\in$ k .. <(#**self**.members) • $\neg$(**self**.members[z **as** nat] < x)

Given: **self**.members.isndec, i = 0 **as** nat, 0 $\leq$ i, k = #**self**.members **as int**

Proof:

[Take given term]

[4.0] k = (#**self**.members **as int**)

$\rightarrow$ *[simplify]*

[4.2] 0 = ($-$k + #**self**.members)

[Take goal term]

[1.0] $\forall$ z $\in$ k .. <(#**self**.members) • $\neg$(**self**.members[z **as** nat] < x)

$\rightarrow$ *[from term 4.2, k is equal to #**self**.members]*

[1.1] $\forall$ z $\in$ #**self**.members .. <(#**self**.members) • $\neg$(**self**.members[z **as** nat] < x)

$\rightarrow$ *[simplify]*

[1.2] $\forall$ z $\in$ #**self**.members .. (-1 + #**self**.members) • $\neg$(**self**.members[z **as** nat] < x)

$\rightarrow$ *[empty range]*

Proof of rule precondition:

[1.2.0] (-1 + #**self**.members) < #**self**.members

$\rightarrow$ *[simplify]*

[1.2.7] **true**

[1.3] $\forall$ z $\in$ **seq of int**{} • $\neg$(**self**.members[z **as** nat] < x)

$\rightarrow$ *[simplify]*

[1.4] $\forall$ z $\in$ **seq of int**{} • $\neg$(**self**.members[z] < x)

$\rightarrow$ *[expand literal quantifier]*

[1.5] **true**

Proof of verification condition: Return value satisfies specification

Condition generated at: C:\Program Files\Escher Technologies\Perfect Developer\Examples\Refinement\BinarySearch.pd (47,13)

Condition defined at: C:\Program Files\Escher Technologies\Perfect Developer\Examples\Refinement\BinarySearch.pd (17,24)

To prove: $i_{23,13} \le$ #**self**.members

Given: **self**.members.isndec, i = 0 **as** nat, $0 \le i$, $0 \le i_{23,13}$, $i_{23,13} \le k'$, $k' \le$ #**self**.members, $\forall$ z $\in$ 0 .. <$i_{23,13}$ • **self**.members[z **as** nat] < x, $\forall$ z $\in$ k′ .. <(#**self**.members) • ¬(**self**.members[z **as** nat] < x), $\forall$ $x $\in$ $attributeNames(**int**) • **different**($i_{23,13}$.$x; $i_{23,13}$) ⇒ i.$x=$i_{23,13}$.$x, $i_{23,13} = k'$, $0 \le i_{23,13}$

Proof:

[Take given term]

[9.0] $i_{23,13} = k'$

→ *[simplify]*

[9.1] $0 = (-k' + i_{23,13})$

[Take given term]

[6.0] $k' \le$ #**self**.members

→ *[simplify]*

[6.7] -1 < ($-k'$ + #**self**.members)

→ *[from term 9.1, k' is equal to $i_{23,13}$]*

[6.8] -1 < ($-i_{23,13}$ + #**self**.members)

[Take goal term]

[1.0] $i_{23,13} \le$ #**self**.members

→ *[simplify]*

[1.7] -1 < ($-i_{23,13}$ + #**self**.members)

→ *[from term 6.8, literala < ($-i_{23,13}$ + #**self**.members) is true whenever -1 < (-1 + −literala)]*

Proof of rule precondition:

[1.7.0] -1 < (-1 + −-1)

→ *[simplify]*

[1.7.3] **true**

[1.8] **true**

Proof of verification condition: Return value satisfies specification

Condition generated at: C:\Program Files\Escher Technologies\Perfect Developer\Examples\Refinement\BinarySearch.pd (47,13)

Condition defined at: C:\Program Files\Escher Technologies\Perfect Developer\Examples\Refinement\BinarySearch.pd (18,13)

To prove: $\forall$ z $\in$ 0 .. <$i_{23,13}$ • **self**.members[z **as** nat] < x

Given: **self**.members.isndec, i = 0 **as** nat, $0 \le i$, $0 \le i_{23,13}$, $i_{23,13} \le k'$, $k' \le$ #**self**.members, $\forall$ z $\in$ 0 .. <$i_{23,13}$ • **self**.members[z **as** nat] < x, $\forall$ z $\in$ k′ .. <(#**self**.members) • ¬(**self**.members[z **as** nat] < x), $\forall$ $x $\in$ $attributeNames(**int**) • **different**($i_{23,13}$.$x; $i_{23,13}$) ⇒ i.$x=$i_{23,13}$.$x, $i_{23,13} = k'$, $0 \le i_{23,13}$

Proof:

[Take given term]

[7.0] $\forall$ z $\in$ 0 .. <$i_{23,13}$ • **self**.members[z **as** nat] < x

→ *[simplify]*

[7.3] $\forall$ z $\in$ (0 .. (-1 + $i_{23,13}$)).ran • **self**.members[z] < x

[Take goal term]

[1.0] $\forall$ z $\in$ 0 .. <$i_{23,13}$ • **self**.members[z **as** nat] < x

→ *[simplify]*

[1.3] $\forall$ z $\in$ (0 .. (-1 + $i_{23,13}$)).ran • **self**.members[z] < x

→ *[from term 7.3, $\forall$ z $\in$ (0 .. (-1 + $i_{23,13}$)).ran •* **self***.members[z] < x is true]*

[1.4] **true**

Proof of verification condition: Return value satisfies specification

Condition generated at: C:\Program Files\Escher Technologies\Perfect Developer\Examples\Refinement\BinarySearch.pd (47,13)

Condition defined at: C:\Program Files\Escher Technologies\Perfect Developer\Examples\Refinement\BinarySearch.pd (19,13)

To prove: $\forall$ z $\in$ $i_{23,13}$ .. <(#**self**.members) • ¬(**self**.members[z **as** nat] < x)

Given: **self**.members.isndec, i = 0 **as** nat, 0 ≤ i, 0 ≤ $i_{23,13}$, $i_{23,13}$ ≤ k′, k′ ≤ #**self**.members, $\forall$ z $\in$ 0 .. <$i_{23,13}$ • **self**.members[z **as** nat] < x, $\forall$ z $\in$ k′ .. <(#**self**.members) • ¬(**self**.members[z **as** nat] < x), $\forall$ $x $\in$ $attributeNames(**int**) • **different**($i_{23,13}$.$x; $i_{23,13}$) ⇒ i.$x=$i_{23,13}$.$x, $i_{23,13}$ = k′, 0 ≤ $i_{23,13}$

Proof:

[Take given term]

[9.0] $i_{23,13}$ = k′

→ *[simplify]*

[9.1] 0 = (−k′ + $i_{23,13}$)

[Take given term]

[8.0] $\forall$ z $\in$ k′ .. <(#**self**.members) • ¬(**self**.members[z **as** nat] < x)

→ *[simplify]*

[8.3] $\forall$ z $\in$ (k′ .. (-1 + #**self**.members)).ran • ¬(**self**.members[z] < x)

→ *[from term 9.1, k′ is equal to $i_{23,13}$]*

[8.4] $\forall$ z $\in$ ($i_{23,13}$ .. (-1 + #**self**.members)).ran • ¬(**self**.members[z] < x)

[Take goal term]

[1.0] $\forall$ z $\in$ $i_{23,13}$ .. <(#**self**.members) • ¬(**self**.members[z **as** nat] < x)

→ *[simplify]*

[1.3] $\forall$ z $\in$ ($i_{23,13}$ .. (-1 + #**self**.members)).ran • ¬(**self**.members[z] < x)

→ *[from term 8.4, $\forall$ z $\in$ ($i_{23,13}$ .. (-1 + #***self***.members)).ran • ¬(***self***.members[z] < x) is true]*

[1.4] **true**

Proof of verification condition: Type constraint satisfied

Condition generated at: C:\Program Files\Escher Technologies\Perfect Developer\Examples\Refinement\BinarySearch.pd (19,58)

Condition defined at: built in declaration

To prove: 0 ≤ z

Given: **self**.members.isndec, 0 ≤ **result′**, **result′** ≤ #**self**.members, $\forall$ z $\in$ 0 .. <**result′** • **self**.members[z **as** nat] < x, z **in** (**result′** .. <(#**self**.members))

Proof:

[Take goal term]

[1.0] 0 ≤ z

→ *[simplify]*

[1.2] -1 < z

→ *[negate goal and search for contradiction]*

[1.3] ¬(-1 < z)

→ *[simplify]*

[1.5] 0 < −z

[Take given term]

[3.0] 0 ≤ **result′**

→ *[simplify]*

[3.2] -1 < **result′**

[Take given term]

[6.0] z **in** (**result′** .. <(#**self**.members))

→ *[simplify]*

[6.5] (0 < (−z + #**self**.members)) ∧ (-1 < (−**result′** + z))

→ *[separate conjunction and work on first sub-term]*

[6.6] -1 < (−**result′** + z)

[Create new term from terms 1.5, 6.6 using rule: transitivity 3]

[10.0] (-1 + 0 + 1) < −**result′**

→ *[simplify]*

[10.1] 0 < −**result′**

→ *[from term 3.2, literala < −**result′** is false whenever -2 < (-1 + literala)]*

Proof of rule precondition:

[10.1.0] -2 < (-1 + 0)

→ *[simplify]*

[10.1.2] **true**

[10.2] **false**

Proof of verification condition: Precondition of '[]' satisfied

Condition generated at: C:\Program Files\Escher Technologies\Perfect Developer\Examples\Refinement\BinarySearch.pd (19,57)

Condition defined at: built in declaration

To prove: (z **as** nat) < #**self**.members

Given: **self**.members.isndec, 0 ≤ **result′**, **result′** ≤ #**self**.members, ∀ z ∈ 0 .. <**result′** • **self**.members[z **as** nat] < x, z **in** (**result′** .. <(#**self**.members))

Proof:

[Take given term]

[6.0] z **in** (**result′** .. <(#**self**.members))

→ *[simplify]*

[6.5] (0 < (−z + #**self**.members)) ∧ (-1 < (−**result′** + z))

[Work on sub-term 2 of conjunction in term 6.5]

[7.0] 0 < (−z + #**self**.members)

[Take goal term]

[1.0] (z **as** nat) $<$ #**self**.members

$\rightarrow$ *[simplify]*

[1.2] 0 $<$ ($-$z + #**self**.members)

$\rightarrow$ *[from term 7.0, literala $<$ ($-$z + #**self**.members) is true whenever -1 $<$ (0 + $-$literala)]*

Proof of rule precondition:

[1.2.0] -1 $<$ (0 + $-$0)

$\rightarrow$ *[simplify]*

[1.2.3] **true**

[1.3] **true**

Proof of verification condition: Type constraint satisfied

Condition generated at: C:\Program Files\Escher Technologies\Perfect Developer\Examples\Refinement\BinarySearch.pd (18,49)

Condition defined at: built in declaration

To prove: 0 $\leq$ z

Given: **self**.members.isndec, 0 $\leq$ **result**$'$, **result**$'$ $\leq$ #**self**.members, z **in** (0 .. $<$**result**$'$)

Proof:

[Take given term]

[5.0] z **in** (0 .. $<$**result**$'$)

$\rightarrow$ *[simplify]*

[5.7] (0 $<$ ($-$z + **result**$'$)) $\wedge$ (-1 $<$ z)

$\rightarrow$ *[separate conjunction and work on first sub-term]*

[5.8] -1 $<$ z

[Take goal term]

[1.0] 0 $\leq$ z

$\rightarrow$ *[simplify]*

[1.2] -1 $<$ z

$\rightarrow$ *[from term 5.8, literala $<$ z is true whenever -1 $<$ (-1 + $-$literala)]*

Proof of rule precondition:

[1.2.0] -1 $<$ (-1 + $-$-1)

$\rightarrow$ *[simplify]*

[1.2.3] **true**

[1.3] **true**

Proof of verification condition: Precondition of '[]' satisfied

Condition generated at: C:\Program Files\Escher Technologies\Perfect Developer\Examples\Refinement\BinarySearch.pd (18,48)

Condition defined at: built in declaration

To prove: (z **as** nat) $<$ #**self**.members

Given: **self**.members.isndec, 0 $\leq$ **result**$'$, **result**$'$ $\leq$ #**self**.members, z **in** (0 .. $<$**result**$'$)

Proof:

[Take goal term]

[1.0] $(\text{z}\ \textbf{as}\ \text{nat}) < \#\textbf{self}.\text{members}$

→ *[simplify]*

[1.2] $0 < (-\text{z} + \#\textbf{self}.\text{members})$

→ *[negate goal and search for contradiction]*

[1.3] $\neg(0 < (-\text{z} + \#\textbf{self}.\text{members}))$

→ *[simplify]*

[1.7] $\text{-}1 < (\text{z} + -(\#\textbf{self}.\text{members}))$

[Take given term]

[4.0] $\textbf{result}' \leq \#\textbf{self}.\text{members}$

→ *[simplify]*

[4.7] $\text{-}1 < (-\textbf{result}' + \#\textbf{self}.\text{members})$

[Take given term]

[5.0] $\text{z}\ \textbf{in}\ (0\ ..\ <\textbf{result}')$

→ *[simplify]*

[5.7] $(0 < (-\text{z} + \textbf{result}')) \land (\text{-}1 < \text{z})$

[Work on sub-term 2 of conjunction in term 5.7]

[6.0] $0 < (-\text{z} + \textbf{result}')$

[Create new term from terms 1.7, 4.7 using rule: transitivity 1]

[7.0] $(\text{-}1 + \text{-}1 + 1) < (-\textbf{result}' + \text{z})$

→ *[simplify]*

[7.1] $\text{-}1 < (-\textbf{result}' + \text{z})$

→ *[from term 6.0, literala* $< (-\textbf{result}' + z)$ *is false whenever -2* $<$ *(0 + literala)]*

Proof of rule precondition:

[7.1.0] $\text{-}2 < (\text{-}1 + 0)$

→ *[simplify]*

[7.1.2] **true**

[7.2] **false**

Proof of verification condition: Precondition of '[]' satisfied

Condition generated at: C:\Program Files\Escher Technologies\Perfect Developer\Examples\Refinement\BinarySearch.pd (57,19)

Condition defined at: built in declaration

To prove: $\text{index} < \#\textbf{self}.\text{members}$

Given: **self**.members.isndec, $0 \leq \text{index}$, $\text{index} < \#\textbf{self}$

Proof:

[Take given term]

[4.0] $\text{index} < \#\textbf{self}$

→ *[expand operator]*

[4.1] $\text{index} < \#\textbf{self}.\text{members}$

→ *[simplify]*

[4.2] $0 < (-\text{index} + \#\textbf{self}.\text{members})$

[Take goal term]

[1.0] index < #**self**.members

→ *[simplify]*

[1.1] $0 < (-\text{index} + \#\textbf{self}.\text{members})$

→ *[from term 4.2, literala < (−index + #**self**.members) is true whenever -1 < (0 + −literala)]*

Proof of rule precondition:

[1.1.0] $-1 < (0 + -0)$

→ *[simplify]*

[1.1.3] **true**

[1.2] **true**

Proof of verification condition: Class invariant satisfied

Condition generated at: C:\Program Files\Escher Technologies\Perfect Developer\Examples\Refinement\BinarySearch.pd (61,14)

Condition defined at: C:\Program Files\Escher Technologies\Perfect Developer\Examples\Refinement\BinarySearch.pd (12,23)

To prove: **self′**.members.isndec

Given: **self** ≈ (**anything**{} **to** Table **of** X), **self′**.members = mem.permndec, ∀ \$x ∈ \$attributeNames(Table **of** X) • **different**(**self′**.\$x; self′.members) ⇒ **self**.\$x=**self′**.\$x

Proof:

[Take given term]

[3.0] **self′**.members = mem.permndec

[Take goal term]

[1.0] **self′**.members.isndec

→ *[from term 3.0,* **self′** *.members is equal to mem.permndec]*

[1.1] mem.permndec.isndec

→ *[negate goal and search for contradiction]*

[1.2] ¬mem.permndec.isndec

→ *[introduce variable 'temp_a' defined as mem.permndec]*

[1.3] ¬temp_a.isndec

[From definition temp_a = mem.permndec]

[5.0] temp_a.isndec

→ *[from term 1.3, temp_a.isndec is false]*

[5.1] **false**

End of proofs for file C:\Program Files\Escher Technologies\Perfect Developer\Examples\Refinement\BinarySearch.pd