

Wikiprint Book

Title: projects/jpf-mango/examples/ltsAWrap

Subject: Java Path Finder - projects/jpf-mango/examples/ltsAWrap

Version: 28

Date: 03/01/2013 06:13:04 PM

Table of Contents

ItsAWrap example, special considerations for loops

3

ItsAWrap example, special considerations for loops

This example focusses on issues that arise in the consideration of loops. In particular, the introduction of "shadow rules" to imply the truth of automatically generated loop-exit conjectures. This example assumes you have already worked through the [Hello World](#) and [CarRecall](#) examples. If not, you might find the steps a little hard to follow. However, links have been provided to parallel steps in the earlier examples for your convenience.

Start by [loading](#) the ItsAWrap project, located inside the "rbk" folder.

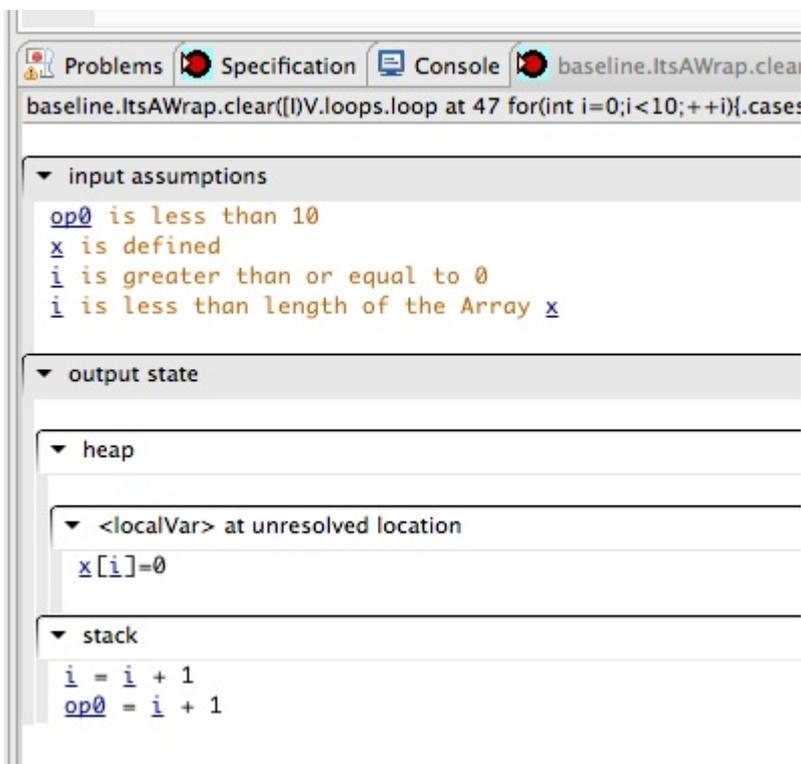
[Restore the default preferences](#), then **check** preferences "Suppress advice dialogs" and "Do not pause for case-splits". Now **uncheck** "Suppress structural hypotheses".

[Open](#) the ItsAWrap class in an editor window.

[Populate and run](#), using the ItsAWrap class as command target.

Observe that three [specifications are generated](#), for the loop body of the clear method, for the clear method, and for the test method. We shall consider each of these in turn.

[Open](#) the loop specification (see figure).



The specification describes the state transition of one pass through the loop. In order to remain in the loop, "i" must be within bounds. The output state sets $x[i]=0$ and increments "i". The rest of the specification concerns "op0". The point is that "i" is pushed onto the operand stack for a compare against 10, just prior to the pass through the loop. Likewise, the loop output state pushes "i+1" onto the operand stack.

Now consider the specification for the clear method.

```

baseline.ItsAWrap.clear([I)V.cases.1
└─ input assumptions
  x is defined
  length of the Array x is greater than or equal to 10
  i^ is greater than or equal to 10
└─ output state
  heap
  heap^
  No return value.

```

shadow rule stuff

The screenshot shows the Mango Explorer IDE with two shadow rule windows open. The left window is titled 'length of x >=10' and the right window is titled 'length of x'. Both windows show the following details:

- Pattern:**
 - Left: `(addHypothesis &<modulePath> 'root')`
 - Right: `(parameterMap (valueH (loc &x ^arrayLength) @inputHeap))`
- Action:**
 - Left: `AddHypothesis`
 - Right: `ShadowParameter`
- Substitution:**
 - Left: `(not (< &"length of x" 10))`
 - Right: (empty)
- Variables:** (empty boxes)

The screenshot shows the IDE's state for a test case. The top bar includes 'Problems', 'Specification', 'Console', and 'baseline.ItsAWrap.clear([I)V.lo'. The main content area shows the following state:

```

baseline.ItsAWrap.test([I)Z.cases.1
└─ input assumptions
  x is defined
  length of the Array x is greater than or equal to 10
  x[5]^ equals x[6]^
└─ output state
  heap
  heap^
  Returns boolean: true

```

main good case

main nonexistent case

length of x >=10 length of x

Problems Specification Console baseline.ItsAWrap.clear(!) cle

baseline.ItsAWrap.test(!)Z.cases.2

▼ input assumptions

- x is defined
- length of the Array x is greater than or equal to 10
- x[5]^ does not equal x[6]^

▼ output state

► heap

Returns boolean: false