My User Study Lessons

Joel Galenson

University of California, Berkeley
joel@cs.berkeley.edu

June 11, 2013
Programming is hard
Programming is hard
Programming is hard
My tool (CodeHint) to the rescue!
My tool (CodeHint) to the rescue!

Select pdspec type: [Demonstrate state property]

Demonstrate a property that should hold for x after this statement is executed. You may refer to the values of variables after this statement is executed using the prime syntax, e.g., x

|x| .toString().contains("Eve") |

Give a skeleton describing the form of the desired expression, using ??s for unknown expressions and names and ***s for an unknown number of arguments.

??

- Search top-level constructors
- Search operators
- Call non-standard native methods (fast but dangerous)
- Log and undo side effects (sound but slow)

Continue search

<table>
<thead>
<tr>
<th>Expression</th>
<th>Result</th>
<th>toString</th>
</tr>
</thead>
<tbody>
<tr>
<td>e</td>
<td>java.awt.event.MouseEvent (id=44)</td>
<td>java.awt.event</td>
</tr>
<tr>
<td>tree.getSelectionModel()</td>
<td>javax.swing.tree.DefaultTreeSelectionModel (id=237)</td>
<td>javax.swing.tree</td>
</tr>
<tr>
<td>tree.getSelectionPath()</td>
<td>javax.swing.tree.TreePath (id=228)</td>
<td>[home, Eve]</td>
</tr>
<tr>
<td>tree.getPathForLocation(mouseX, mouseY)</td>
<td>javax.swing.tree.TreePath (id=228)</td>
<td>[home, Eve]</td>
</tr>
<tr>
<td>SwingUtilities.convertMouseEvent(tree, e, null)</td>
<td>javax.awt.event.MouseEvent (id=460)</td>
<td>java.awt.event</td>
</tr>
<tr>
<td>tree.getCellRenderer()</td>
<td>javax.swing.tree.DefaultTreeCellRenderer (id=222)</td>
<td>javax.swing.tree</td>
</tr>
<tr>
<td>tree.getLeadSelectionPath()</td>
<td>javax.swing.tree.TreePath (id=228)</td>
<td>[home, Eve]</td>
</tr>
<tr>
<td>tree.getCloserPathForLocation(mouseX, mouseY)</td>
<td>javax.swing.tree.TreePath (id=228)</td>
<td>[home, Eve]</td>
</tr>
<tr>
<td>tree.getCloserPathForLocation(clickedRow, mouseY)</td>
<td>javax.swing.tree.TreePath (id=228)</td>
<td>[home, Eve]</td>
</tr>
<tr>
<td>tree.getLastSelectedPathComponent()</td>
<td>javax.swing.tree.DefaultMutableTreeNode (id=227)</td>
<td>Eve</td>
</tr>
<tr>
<td>tree.getAnchorSelectionPath()</td>
<td>javax.swing.tree.TreePath (id=228)</td>
<td>[home, Eve]</td>
</tr>
</tbody>
</table>

Check all | Uncheck all | Check selected | Uncheck selected

Filter expressions, results, and javadoc by words:

Filter | Cancel
First study

- Goals:
  - Does my tool improve programmer productivity? Do people like it?
  - How do people use my tool? What can I improve?
- Three scenarios broken into five short problems each.
- Users randomly solved tasks with or without my tool.
- Ran a second phase months later to update results.
First study

Goals:
- Does my tool improve programmer productivity? Do people like it?
- How do people use my tool? What can I improve?

Three scenarios broken into five short problems each.

Users randomly solved tasks with or without my tool.

Ran a second phase months later to update results.
First study results

Fewer bugs ($p < 0.2$)
Lots of interesting usage information.
Second study

- Main idea: larger problems.
- Two complex scenarios broken into three medium-size problems each.
- Initially targeting undergraduates.
Second study results

Productivity (p < 0.01)

\[\text{% completed with CodeHint vs. % completed without CodeHint}\]
Lessons

- Learn statistics.
Lessons

- Learn statistics.
- Collect lots of data.
  - Screencast, instrument tool
- Design your tutorial carefully.
- Short, targeted, interactive, motivating
- Think about your target population.
- Test your user study.
- Try to reduce variance.
- Getting users (with $) is easy?
Lessons

- Learn statistics.
- Collect lots of data.
  - Screencast, instrument tool
- Design your tutorial carefully.
  - Short, targeted, interactive, motivating
Lessons

- Learn statistics.
- Collect lots of data.
  - Screencast, instrument tool
- Design your tutorial carefully.
  - Short, targeted, interactive, motivating
- Think about your target population.
Lessons

- Learn statistics.
- Collect lots of data.
  - Screencast, instrument tool
- Design your tutorial carefully.
  - Short, targeted, interactive, motivating
- Think about your target population.
- Test your user study.
Lessons

- Learn statistics.
- Collect lots of data.
  - Screencast, instrument tool
- Design your tutorial carefully.
  - Short, targeted, interactive, motivating
- Think about your target population.
- Test your user study.
- Try to reduce variance.
Lessons

- Learn statistics.
- Collect lots of data.
  - Screencast, instrument tool
- Design your tutorial carefully.
  - Short, targeted, interactive, motivating
- Think about your target population.
- Test your user study.
- Try to reduce variance.
- Getting users (with $) is easy?