Class exercise: Testing a two-dimensional array

COMP 40
September 28, 2009

Group

<table>
<thead>
<tr>
<th>Keeper of the record:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Other group members:</td>
</tr>
</tbody>
</table>

Questions

Most bugs in two-dimensional arrays come from bad invariants or misunderstood invariants. Your invariant must satisfy two key properties:

A. It maps each world-of-ideas container to a unique memory location in the world of code (one-to-one map).
B. Every location in the world of code is in memory you own.

Answer these questions:

1. What simple test could you run to make sure the mapping is one-to-one? (You may ask for a hint.)
2. What tool could you use to make sure every mapped location in the world of code is in memory you own?
3. In the world of ideas, a two-dimensional array is characterized by three pieces of information: a width, a height, and the (common) type of each container. Is it possible to make a mistake by confusing one piece of information with another? If so, what confusion is most likely?
4. Given your answer to question 3, what sorts of two-dimensional arrays should you test in questions 1 and 2?

Please return your work to the course staff.