Proof Rules for Arrays

Arrays

We have seen that the regular Assignment Axiom is not valid for assignments involving arrays. This does not preclude proving with arrays, but we can't use the regular Assignment Axiom. In this presentation, we examine how to perform valid deductions when array assignments are permitted.

In fact, the regular Assignment Axiom is still valid when arrays are present, as long as assignment to *subscripted* variables is prohibited in the programs to be proven (i.e., "read only" arrays). It is when assignments to subscripted variables are permitted that we need to change. To accomplish sound reasoning about (read/write) arrays, it will be shown sufficient to require assertions that treat arrays exclusively as a whole. Assertions only characterize the values of array variables, not element variables. This is a restriction on the logical assertions, not the program text.

Syntax (Wren and Pelican) for arrays is described informally. Identifiers are declared to denote arrays, and we refer to them as *array-identifiers*. We will consider only one-dimensional arrays here, and we will assume that the items in an array are of type **integer**. We will not be concerned about the subscript ranges at this point. As a practical matter, our assertions commonly restrict the range of subscript values. The syntax denoting an array entry will be the familiar form, array-id[integer expr], and subscripted variables may appear at any point where ordinary variables are allowed. For this discussion we focus on the proof elements.

For an array *value* (including, but not restricted to a variable), say **a**, we use the notation **a**[i: e], where subscript i and expression e are integers, to denote the array value that is identical to **a** except at index i, and at that position contains the value e; since **a**[i: e] is another array value, similar notation may again be applied to it, etc. This notation is reserved for use in assertions, not in program code. The following rules are provided for array value deductions:

Array assignment axiom

*The conventional assignment axiom is forbidden for use with assignment to array element**s*** (i.e., subscripted variables), but continues to be used for simple variables. For assignment to subscripted variables we provide an additional axiom scheme:

{P[a a[i: e]} a[i]:= e {P} — **ARRAY-ASSIGN** (for any post-condition P).

Note that each occurrence of the array-identifier is to be replaced. Hence each element variable a[k] in the post-condition P would be changed to a[i: e][k] in the pre-condition.

We also add the following auxiliary rules.

Array value rules — ARRAY-VAL

$$\frac{j \neq i}{\mathbf{a}[i:e][j] = \mathbf{a}[j]}$$ (or we could write j ≠ i **a**[i: e][j] = **a**[j])

$$\frac{j = i}{\mathbf{a}[i:e][j] = e}$$ (or we could write j = i **a**[i: e][j] = e

Note that i and j are permitted to be (integer) expressions in these proof rules.