Subject: Finding a value in a array efficiently
Posted by news.verizon.net on Thu, 13 Apr 2006 20:31:42 GMT
View Forum Message <> Reply to Message

I was asked an apparently simple question -- what is the most efficient way in IDL to determine if a particular scalar value is in an array.
The loop method is straightforward:

```
found = 0
for i=0,n-1 do  begin
    if array[i] EQ value then begin
        found = 1
        goto, DONE
   endif
endfor
DONE:
```

This ugly code will end the loop as soon as a match is found.
A first attempt at vectorized code might be

```
index = where(array EQ value, ng)
found = ng gt 0
```

But this code both creates an unnecessary index array, and requires searching the entire array even if the match is found in the first element. A more efficent (if less transparent) method might be

```
found = 1 - array_equal( array EQ value, 0)
```

where (array EQ value) will contain all zeros if there is no match.
The ARRAY_EQUAL(x,0) function returns a value of zero as soon as it finds a non-zero value in x. So we are part way there but (array EQ value) still requires testing every value of "array".

So what we need is a new ARRAY_OR(x,y) function which returns 1 as soon as it finds any match between X and Y. Or am I missing another method? --Wayne