Subject: Re: passing structure elements... by value?
Posted by Jaco van Gorkom on Mon, 16 Jul 2001 09:45:29 GMT
View Forum Message <> Reply to Message

Randall Skelton wrote:

```
>  I have a simple routine that increments the value of a structure:
>
>  pro test, mod_struct
>    for i=0, n_tags(mod_struct)-1 do begin
>      mod_struct.(i) = mod_struct.(i) + 1
>    endfor
>  end
>
>  As expected, this will increment all of the values in a passed structure.
>
>  IDL> struct = {A:0, B:0} & big = replicate(struct,3)
>  IDL> print, big
> {       0       0}{       0       0}{       0       0}
>  IDL> test, big
>  IDL> print, big
> {       1       1}{       1       1}{       1       1}
>
>  But, when I try and increment a single element in the structure it fails?
>
>  IDL> struct = {A:0, B:0} & big = replicate(struct,3)
>  IDL> test, big[0]
>  IDL> print, big
> {       0       0}{       0       0}{       0       0}
>  ^^^^^^^^^^^^^^^^^^
>
>  I expected the ^^ element above to be ones?  Is there any way to force IDL
>  to pass this by reference instead of passing by value? It would be nice if
>  you could put brackets around the thing you want to pass by reference...
>  something like '(big[0])'
```

Note that in your example you are passing one *array* element of an array of structures. Indexed array elements are also passed by value - see Liam's list.
One workaround would be to copy the array element(s) into a temporary variable:

```
working_elements = big[range]
test, working_elements
big[range] = working_elements
```

, with range being a vector of indices, just [0] in the example.
Another workaround involves passing the indices-of-interest into the routine.

cheers,
  Jaco

Jaco van Gorkom gorkom@rijnh.nl
FOM-Instituut voor Plasmafysica `Rijnhuizen', The Netherlands