Subject: Re: Strange array subscripting error
Posted by pgrigis on Tue, 07 Jul 2009 23:37:11 GMT
View Forum Message <> Reply to Message

robintw wrote:
> Hi,
>
> I'm getting the error "% Out of range subscript encountered: VALUES.",
> but I can't work out why. I have three arrays (azimuths, zeniths and
> values) each of which is set to a size of 360 * 90 (which is 32400). I
> then have a loop which populates these arrays with values, but when
> the loop gets to 32398 it stops and gives the error above.
>
> I really can't understand what's going on here. I've made sure that
> the variable I'm using to keep the array_index in is a long, in case
> above 32398 it was going over a limit in a standard integer, but that
> didn't help. The only way I've found to get round it is to manually
> add three to my array declaration (ie. change it to fltarr((360*90) +
> 3)). That is obviously a very ugly hack, and ends up with me having
> some blank unused array values at the end.
>
> Does anybody have any ideas why this is happening and what I can do
> about it? I've attached the code below:
>
> PRO BRUNGER_HOOPER_MODEL, a0, a1, a2, a3, azimuths=azimuths,
> zeniths=zeniths, values=values, s_theta, s_phi
> sun_theta = s_theta*!DTOR
> sun_phi = s_phi*!DTOR
>
> ; Initialise arrays
> array_size = (360*90) + 3 ; BUG ALERT! When set to 360*90 (32400) it
> seems to overrun at 32398, this is an ugly fix
>
> azimuths = intarr(array_size)
> zeniths = intarr(array_size)
> values = fltarr(array_size)
>
> FOR phi=0, 360-1 DO BEGIN
> FOR theta=0, 90-1 DO BEGIN
> ; Convert the current phi and theta to radians
> view_phi = phi*!DTOR
> view_theta = theta*!DTOR
>
> value = CALCULATE_SKY_VALUE(a0, a1, a2, a3, view_theta,
> view_phi, sun_theta, sun_phi)
>
> array_index = long((90*phi) + theta)

This statement looks very dangerous. While it won't
actually overflow in your example, it comes very close
to it. instead of applying long() to the result, phi and theta
should be longs from the start (i.e. for phi=0L,360...).
If you want to see why, try the difefrence between:
IDL> print,long(256^2)
IDL> print,256L^2

The former is probably going to mess you up, while the latter
is fine.

This is good advice, even if that was not the cause of your problem :)

Ciao,
Paolo

```
>
>       ; Put the value into the array
>       values[array_index] = value
>       azimuths[array_index] = phi
>       zeniths[array_index] = theta
>
>     ENDFOR
>   ENDFOR
>
>   ; Normalise the values
>   values = values / MAX(values)
> END
```