Subject: Re: IDL FOR Loop variable increments
Posted by Wasit.Weather on Sun, 21 Sep 2008 13:44:02 GMT
View Forum Message <> Reply to Message

On Sep 20, 12:51 am, Raghu <raghuram.narasim...@gmail.com> wrote:
> On Sep 19, 12:09 pm, pgri...@gmail.com wrote:
>
>
>
>
>
>> R.G. Stockwell wrote:
>>> "Jean H" <jghas...@DELTHIS.ucalgary.ANDTHIS.ca> wrote in message
>>> news:gauiil$u32$1@news.ucalgary.ca...
>>> ...
>>>> Could you comment on the "risk" of changing the loop counter within the
>>>> loop?
>
>>> my 2 cents.
>
>>> First, it is in changing the counter of a for loop.
>>> A for loop explicitly outlines what all counter variables will be.
>
>>> There are two things:
>
>>> 1) infinite loop, one could easily change the counter to never
>>> reach the end condition. A (valid) for loop will always reach the end
>>> condition.
>
>>> 2) more insidious, you could inadvertantly cast the counter to a float from
>>> an int, and then have one extra (and unintended) statement executed.
>
>> This seems not to be possible in IDL, as loop counters, unlike normal
>> variables, cannot change their type.
>
>> Ciao,
>> Paolo
>
>>> instead of 0,1,2,3,4,5,6 (and not executing i = 7) you could get
>>> 0,1,2,3,4,4.99999999,5.99999,6.99999999, (and effectively executing the
>>> extra i ~ 7 step).
>
>>> Cheers,
>>> bob
>
> Hi all,
>
> Thanks for your replies. Just as David mentioned in his first

> response, a while loop worked out much better. Within a single while
> loop, i was able to accomplish the task, albeit a bit slowly because
> of the non-array operation.
>
> Thanks !- Hide quoted text -
>
> - Show quoted text -

Why not you do not share your final results with us to close this
post.
Elkunn