Subject: Going outside the interger limits
Posted by lloyd on Tue, 21 Sep 2004 11:22:01 GMT

Hi all,

Have been muddling about with this problem for the past weeek, and I'm hoping someone can help.

I'm creating an atmospheric transmission model for the sub-mm/far infrared region. If I try to calculate the spectrum between 1 and 500 wavenumbers (numbers are not important) and I set a high resolution, the end of the spectrum does not seem to have had spectral line data applied to it.

Having looked at where this is happening (at about point number 32700), it lies at the point where an integer runs outside it's limits. Inside the code, if a spectral line value is negative then it gets ignored, hence why I'm assuming (well guessing really) that this is the problem.

I have been through my code and converted as much as I can see into either a long or a float depending on what it needs to be.

Still getting problems!

I was wondering whether there is anyway to tell in IDL if an integer is trying to go outside it's limits? Such as !except = 2, or is there a compiler switch which would break the execution if this problem was encountered (the latter would be very handy).

Thanks for any help,

Lloyd Watkin

Subject: Re: Going outside the interger limits
Posted by David Fanning on Wed, 22 Sep 2004 15:03:03 GMT

Lloyd Watkin writes:

> In the end, it turned out that I was taking fix(a float) where as what
> I should of done is taken long(a float) as I knew I'd be going outside
> the limit.
>
> Typical!

Oh, it always turns out like this. Even with the experts. :-)

Cheers,

David

--
David W. Fanning, Ph.D.
Fanning Software Consulting, Inc.
Coyote's Guide to IDL Programming: http:/www.dfanning.com/
Phone: 970-221-0438, IDL Book Orders: 1-888-461-0155

Subject: Re: Going outside the interger limits
Posted by R.G. Stockwell on Wed, 22 Sep 2004 16:55:31 GMT
View Forum Message <> Reply to Message

"Lloyd Watkin" <lloyd@evilprofessor.co.uk> wrote in message
news:c3f97ff.0409220637.380f5018@posting.google.com...
> In the end, it turned out that I was taking fix(a float) where as what
> I should of done is taken long(a float) as I knew I'd be going outside
> the limit.
>
> Typical!
>
> Thanks for all the help as usual,
>
> Lloyd Watkin

lol, i do this all the time (well i did until i outlawed the use of fix)
That and using "for i = 0, maxint-1 do begin"
instead of "for i = 0L, maxint-1 do begin"

They should change the name of fix() to areyousureyouwanttocasttoashort()

Cheers,
bob

Subject: Re: Going outside the interger limits
Posted by JD Smith on Wed, 22 Sep 2004 18:46:45 GMT
View Forum Message <> Reply to Message

On Wed, 22 Sep 2004 10:55:31 -0600, R.G. Stockwell wrote:

> "Lloyd Watkin" <lloyd@evilprofessor.co.uk> wrote in message
> news:c3f97ff.0409220637.380f5018@posting.google.com...

>> In the end, it turned out that I was taking fix(a float) where as what I
>> should of done is taken long(a float) as I knew I'd be going outside the
>> limit.
>>
>> Typical!
>>
>> Thanks for all the help as usual,
>>
>> Lloyd Watkin
>
> lol, i do this all the time (well i did until i outlawed the use of fix)
> That and using "for i = 0, maxint-1 do begin" instead of "for i = 0L,
> maxint-1 do begin"
>
> They should change the name of fix() to areyousureyouwanttocasttoashort()

You could always add:

 COMPILE_OPT DEFINT32

to the top of your routine definitions. It would be nice if you could
do it once for an entire class or package of routines, but I think
each routine requires it. You could always @include it, for easy
change later. For me, LOGICAL_PREDICATE would also be a good
addition. I just can't bring myself to stick it at the top of every
routine. I suppose I could change my IDLWAVE templates to include
the correct incantations...

JD