Subject: Re: 1e38 limit?
Posted by paulartcoelho on Fri, 27 Feb 2009 18:39:08 GMT
View Forum Message <> Reply to Message

i think i found it, but i'm not sure how to explain...

i have a loop (integer) that goes through a list of files, reading
data and computing stuff. i added a 'print,loop' line to the code, and
could see the routine would get stuck in one specific file, staying
there forever. thus i opened this file in IDL prompt and went through
all the steps of the routine one by one. doing the things via prompt,
i was getting an underflow message when reaching a line

pdf = exp(-0.5*chi)

because for the data inside that (only) file, the 1d40 factor i had
applied before was too large and chi ended up of the order of 1e4, and
then pdf would end up equal to 0 + underflow.

but, why running the routine automatically the code would get stuck
there, instead of giving the underflow message and moving on, i don't
know. i can guess the zero-ed array ended up producing a NaN array
somewhere later in the routine that messed the things up. to be honest
i haven't gone further as i stopped to improve the code, so that i
will test/treat the data on-the-fly and avoid the underflow in the
first place :).

anyway, i had to change a couple of 'e' to 'd' in the code indeed, as
you pointed out. as well as creating a double-precision array to read
the data from the asc files (it was float before and some numbers were
being lost without me realising it).

paula

On Feb 27, 6:39 pm, Chris <beaum...@ifa.hawaii.edu> wrote:
> The overflow, by itself, shouldn't result in an infinite process - idl
> should just complain about floating point overflow maybe.
>
> What COULD be happening is that you have a loop in your code. You are
> using a floating point value as the loop variable, and incrementing it
> by a number much smaller than itself. When that happens, there isn't
> enough precision to store the difference between the numbers before
> and after increment, and the loop variable stays the same forever.
> It's actually a floating point UNDERFLOW.
>
> Maybe? any floating point loops?
>

> chris