Subject: Re: Use IDL6.0 to read gcc3.4(Mingw32) written data
Posted by Nianming Zuo on Thu, 09 Aug 2007 07:03:48 GMT
View Forum Message <> Reply to Message

Mark,

Thank you for your replay and your recommendation.

Finally I found the crux of the matter. It was caused by the different
way
to treat " fwrite(fp, "w") " sentence on Linux/Unix and MS Windows.
(Initially, my program was developped on Linux and currently to
transfer it)

On MS Windows, when I use IDL6.0 to read (openr, readu) data generated
by
gcc3.4.2 (Mingw32), it produces large values and prompts
" Program caused arithmetic error: Floating underflow " , so I
suspected
it is caused by gcc (transferred from Linux), then I used
SWAP_ENDIAN .
Perhaps it is a unreasonale trial.

Any way, it has been defeated!!

Thank you!

Tony

On 8 9 , 5 29 , Mark Hadfield <badjelly.wi...@gmail.com> wrote:
> I don't know if the following is the cause of your problem, but in the
> following code snippet...
>
> cmat = fltarr(8,4)
> openr, lun, "cmat.dat",/GET_LUN,/swap_endian
>
> readu, lun, ii
> readu, lun, cmat
> readu, lun, jj
>
> print, "ii", ii
> print, "cmat", cmat
> print, "jj", jj
>
> ...I see no sign that you have created the variables ii and jj before
> reading them. In this case the variables will be created as scalars of
> type float, which will not do what you intend.
>

> On another tack, I suggest you download and install Hedit (it's free)
>
> http://www.yurisw.com/HEdit.htm
>
> With this tool you can look at the contents of your binary file. It
> has a handy display at the top: you move the cursor to any position
> and it shows the value that the 1, 2, 4 or 8 bytes starting at that
> position will have if interpreted as a binary, byte, short, long,
> float or double scalar. It should help you work out why one of your
> files is 2 bytes longer than you expected.
>
> On yet another tack, I don't understand why you would have to open
> your files with SWAP_ENDIAN if they are being written on the same
> platform as they are being read.
