Subject: Re: Runtime error woes
Posted by Mort Canty on Mon, 30 Jul 2007 08:44:00 GMT
View Forum Message <> Reply to Message

Craig Markwardt schrieb:
> mort canty <m.canty@fz-juelich.de> writes:
>
>> I mentioned this one before, but I'm really hung up on it, so I'll try
>> again:
>>
>> Running a rather long program on my desktop PC under Windows XP, IDL
>> 6.3 with !EXCEPT=2 I get the following messages:
>>
>> Program caused arithmetic error: Floating illegal operand
>> Detected at FFN::COST 160 D:\Idl\projects\supervised\ffn__define.pro
>> Program caused arithmetic error: Floating illegal operand
>> Detected at FFN::COST 162 D:\Idl\projects\supervised\ffn__define.pro
>> Program caused arithmetic error: Floating illegal operand
>> Detected at FFN::COST 164 D:\Idl\projects\supervised\ffn__define.pro
>> Program caused arithmetic error: Floating illegal operand
>> Detected at FFNKAL::TRAIN 112
>> D:\Idl\projects\supervised\ffnkal__define.pro
>> Program caused arithmetic error: Floating illegal operand
>> ...
>> ...
>
> It sounds like it's time to put some HELP debugging statements around
> line 160 of ffn__define.pro
>
>> This goes on for a while and then the program crashes. I cannot for
>> the life of me find any errors in the code at the lines indicated
>> that could cause this. In fact, I don't believe there are any, because
>> what's driving me nuts is this: On my laptop, exact same program,
>> exact same OS, exact same IDL, still with !EXCEPT=2, I get no runtime
>> error messages whatsoever and the program completes successfully.
>
> The obvious question, since you didn't mention it: is the data set
> exactly the same?
>
> Craig

Hi Craig. Quite right of you to ask, but yes, the data sets are
identical on both systems. I've been setting breakpoints strategically
in the code so that the program will interrupt when an array has a
non-finite element, a la IF MIN(FINITE(VAR) EQ 0 ... I find that, on the
desktop computer, NaNs are appearing which then propagate through the
calculation and bring it to its knees. On the laptop the breakpoints are
not triggered and everything is fine. Can this maybe be caused by an

incorrect use of pointers, which happens to be innocuous in one memory configuration and disasterous in the other? Or does IDL's heap inmplementation exclude such things?

Mort
