Subject: Re: Search single column of array - removing nasty loop
Posted by rjp23 on Thu, 01 Dec 2011 10:37:13 GMT
View Forum Message <> Reply to Message

On Nov 30, 8:15 pm, Yngvar Larsen <larsen.yng...@gmail.com> wrote:
> On Nov 29, 6:53 pm, Heinz Stege <public.215....@arcor.de> wrote:
>
>> Hi Rob,
>
>> no loop necessary:
>
>> array=(randomu(seed,2,6,360,42)-.1)>0. ; sample array
>> array=reform(array,n_elements(array)/42,42,/overwrite)
>> ii=where(min(array,dim=2) eq 0.,count)
>> if count ge 1 then array[ii,*]=0.
>> array=reform(array,2,6,360,42,/overwrite)
>
> Hm. The /OVERWRITE keyword to REFORM was new to me. Thanks!
>
> Silly me. I have somehow always imagined that the compiler was smart
> enough to do this (i.e. not copy any data, only alter the internal IDL
> descriptor of the ARRAY variable) automatically when input and output
> to REFORM is the same variable. But a bit of profiling shows this is
> not at all the case. This will be _very_ useful many places in my
> operational code...
>
> A small comment to the code above: "where(min(array,dim=2) eq 0.)"
> obviously only works if array contains only non-negative data. If not,
> "where(total(array eq 0, 2) gt 0)" will do the trick also for floating
> point data containing negative numbers, with more or less the same
> performance.
>
> --
> Yngvar

Thanks, that explains why a few results were coming out slightly
differently as there are a few negative values.

Also, the code fails when the final column only has 1 element in it.

IDL> help, array
ARRAY DOUBLE = Array[4320, 1]
IDL> help, total(array eq 0, 2)
% TOTAL: For input argument <BYTE Array[4320]>, Dimension must be
1.