Subject: Re: Dos and Donts
Posted by Liam E. Gumley on Wed, 27 Sep 2000 07:00:00 GMT
View Forum Message <> Reply to Message

brinks wrote:
> I'm preparing a little talk about efficient programming in IDL. From the
> help files and from my own experience I found many hints about 'what to
> do and not to do' to speed up code in IDL. The most things I listed so
> far are about array operations and data arrangement, avoiding loops etc.
> Taking an intensive look into the help system I found some little tricks
> to improve speed, e.g. using rotation for image rotation by 90 deg
> rather than rot(...,90). Up to now I didn't know that there is a
> difference at all.
>
> I am sure there are many more odds and ends to improve IDL code. Does
> anyone out there can give my a hint where to find more of those tricks?
> Links, recommendation of a book, or just some direct tips would be very
> helpfull.

http://www.rsinc.com/services/output.cfm?tip_id=1799

http://www.sf.med.va.gov/mrs/IDL/idl_docs.htm#MEMORY USAGE REDUCTION IN IDL

And the online document 'Building IDL applications', section 'Programming in IDL'.

And finally, one from me. The TEMPORARY function can often be used to conserve memory. However it's not immediately obvious (at least it wasn't to me until recently) how to use TEMPORARY when extracting an array subset.

For example, TEMPORARY saves no memory in the following example:

```
a = dist(256)
b = temporary(a[0:63, 0:63])
help, a, b
A               FLOAT     = Array[256, 256]
B               FLOAT     = Array[64, 64]
```

The argument a[0:63, 0:63] is passed to TEMPORARY by value, and therefore cannot be modified. However the following method *does* save memory:

```
a = dist(256)
b = (temporary(a))[0:63, 0:63]
help, a, b
A               UNDEFINED = <Undefined>
```

B FLOAT = Array[64, 64]

Cheers,
Liam.
http://cimss.ssec.wisc.edu/~gumley

Subject: Re: Dos and Donts
Posted by promashkin on Wed, 27 Sep 2000 07:00:00 GMT
View Forum Message <> Reply to Message

If I had a talk like that to give, it would be easy for me now since I
have David's new book. I'd simply choose a few pages, according to the
expected level of expertise of the people, and read them to the audience
:-) You don't even need personal IDL experience.
Cheers,
Pavel

Subject: Re: Dos and Donts
Posted by davidf on Wed, 27 Sep 2000 07:00:00 GMT
View Forum Message <> Reply to Message

Hans Joerg Brinksmeyer (h.j.brinksmeyer@physik.uni-marburg.de) writes:

> 'm preparing a little talk about efficient programming in IDL. From the
> help files and from my own experience I found many hints about 'what to
> do and not to do' to speed up code in IDL. The most things I listed so
> far are about array operations and data arrangement, avoiding loops etc.
> Taking an intensive look into the help system I found some little tricks
> to improve speed, e.g. using rotation for image rotation by 90 deg
> rather than rot(...,90). Up to now I didn't know that there is a
> difference at all.
>
> I am sure there are many more odds and ends to improve IDL code. Does
> anyone out there can give my a hint where to find more of those tricks?
> Links, recommendation of a book, or just some direct tips would be very
> helpfull.

Here's a tip: Try the Google search engine. It really is
a marvel. A search on "idl programming tricks" turned up
all kinds of good stuff.

http://www.google.com/

Cheers,

David

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

Subject: Re: Dos and Donts
Posted by on Thu, 28 Sep 2000 07:00:00 GMT
View Forum Message <> Reply to Message

"Liam E. Gumley" wrote:
> And finally, one from me. The TEMPORARY function can often be used to
> conserve memory. However it's not immediately obvious (at least it
> wasn't to me until recently) how to use TEMPORARY when extracting an
> array subset.
>
> For example, TEMPORARY saves no memory in the following example:
>
> a = dist(256)
> b = temporary(a[0:63, 0:63])
> help, a, b
> A FLOAT = Array[256, 256]
> B FLOAT = Array[64, 64]
>
> The argument a[0:63, 0:63] is passed to TEMPORARY by value, and
> therefore cannot be modified. However the following method *does* save
> memory:
>
> a = dist(256)
> b = (temporary(a))[0:63, 0:63]
> help, a, b
> A UNDEFINED = <Undefined>
> B FLOAT = Array[64, 64]

Finally, that's something new, I didn't know before!!!
Thank you very much :-)

Hans Joerg

--
H.J.Brinksmeyer - Physics Department
University of Marburg, Germany
