Subject: Re: about memory
Posted by Jaco van Gorkom on Mon, 21 May 2001 21:33:00 GMT
View Forum Message <> Reply to Message

Hong Gang wrote:
> The following is the error given by my program.
> % Unable to allocate memory: to make array.
> Cannot allocate memory
>
> I have some large arrays as Tb(632*664, 880). I do not know what is the
> matter with the program. Can anyone give me some idea?

Usually this means that IDL has run out of memory. An array of dimensions 419648 (632*664) by 880 would have 369290240 elements. If it is of type FLOAT, then each element takes up 4 bytes, coming to a grand total of almost 1.4 Gigabyte. Do you have that much memory?

It might be possible to increase the amount of virtual memory (= swap space on disk) available, but swapping memory from disk invariably slows down programs quite a lot. A much better approach would be to rewrite your program so that it needs less memory, e.g. by processing the data in smaller blocks.

hope this helps,
Jaco

Subject: Re: about memory
Posted by thompson on Mon, 21 May 2001 22:01:23 GMT
View Forum Message <> Reply to Message

Hong Gang <honggang@uni-bremen.de> writes:

> Hallo all,

> The following is the error given by my program.
> % Unable to allocate memory: to make array.
> Cannot allocate memory

> I have some large arrays as Tb(632*664, 880). I do not know what is the
> matter with the program. Can anyone give me some idea?

IDL is telling you that the operating system is not allowing it to allocate enough memory to create the array. It's not surprising; a floating point array of that size would take well over a gigabyte of memory. Generally speaking, you'd need several times that much memory to manipulate such an array.

You don't mention what operating system you're using. If Unix, then you can find out how much memory the operating system will allow you to have by

entering in the command "limit" (before calling IDL). You can also type
"limit -h" to find the hard limits. The relevant number is "datasize". You
can increase this, e.g. "limit datasize 200000" would give you 200 megabytes of
memory.

Probably, you should rewrite your software to manipulate your data in pieces,
instead of all at once. One way to do this is to write your data to a file,
and then use the ASSOC() function to address parts of the file.

William Thompson

Subject: Re: about memory
Posted by Amar Nayegandhi on Mon, 21 May 2001 22:38:02 GMT
View Forum Message <> Reply to Message

This is a general question about memory management.
can i free memory allocated to an array if I have finished using it in
the program? suppose
arr_a = fltarr(10000,10000)
if i don't need array arr_a anymore, would
arr_a = 0
free the memory used by arr_a? I would expect it to now utilize only
2(or 4) bytes of memory.

this may be helpful to you, Hong. if you have multiple Tb arrays, and
are not using them anymore in the program, you can free some memory by
initializing Tb to 0(if it works!)

-amar

William Thompson wrote:
>
> Hong Gang <honggang@uni-bremen.de> writes:
>
>> Hallo all,
>
>> The following is the error given by my program.
>> % Unable to allocate memory: to make array.
>> Cannot allocate memory
>
>> I have some large arrays as Tb(632*664, 880). I do not know what is the
>> matter with the program. Can anyone give me some idea?
>
> IDL is telling you that the operating system is not allowing it to allocate
> enough memory to create the array. It's not surprising; a floating point array
> of that size would take well over a gigabyte of memory. Generally speaking,
> you'd need several times that much memory to manipulate such an array.

>
> You don't mention what operating system you're using. If Unix, then you can
> find out how much memory the operating system will allow you to have by
> entering in the command "limit" (before calling IDL). You can also type
> "limit -h" to find the hard limits. The relevant number is "datasize". You
> can increase this, e.g. "limit datasize 200000" would give you 200 megabytes of
> memory.
>
> Probably, you should rewrite your software to manipulate your data in pieces,
> instead of all at once. One way to do this is to write your data to a file,
> and then use the ASSOC() function to address parts of the file.
>
> William Thompson

--
****
Murphy's Law of Research:
Enough research will tend to support your theory.
****
Amar Nayegandhi
Graduate Student
Department of Computer Science,
University of South Florida, Tampa.

Subject: Re: about memory
Posted by Craig Markwardt on Tue, 22 May 2001 00:08:52 GMT
View Forum Message <> Reply to Message

Amar Nayegandhi <anayegan@csee.usf.edu> writes:

> This is a general question about memory management.
> can i free memory allocated to an array if I have finished using it in
> the program? suppose
> arr_a = fltarr(10000,10000)
> if i don't need array arr_a anymore, would
> arr_a = 0
> free the memory used by arr_a? I would expect it to now utilize only
> 2(or 4) bytes of memory.

The answer to the question is "yes," but also "it depends." The
answer is "yes" because the memory is indeed made available again to
the same IDL process. I use

```
A = 0
```

every day when I am done with a variable. It's totally legit. Hong
may need go no further than this, although I agree that he's dealing

with extremely large arrays which might be better suited with a tiled
or chunked approach.

The answer is also "it depends," because it depends on which platform
you are using. I believe that under Windows the memory is actually
returned to the OS. Thus, other programs are able to use the memory
again. Under Unix this is much less likely, so once the memory is
allocated to one session of IDL, it stays there until the session ends
(and is not available to another process). [I think this is not
always true, but mostly true.]

Every so often somebody asks, "why is IDL eating all my memory?" It's
basically unavoidable. Sorry for this diversion.

Craig

--
-- --------------
Craig B. Markwardt, Ph.D. EMAIL: craigmnet@cow.physics.wisc.edu
Astrophysics, IDL, Finance, Derivatives | Remove "net" for better response
-- --------------

Subject: Re: about memory
Posted by Med Bennett on Tue, 22 May 2001 14:30:55 GMT
View Forum Message <> Reply to Message

Craig Markwardt wrote:

> Amar Nayegandhi <anayegan@csee.usf.edu> writes:
>
>> This is a general question about memory management.
>> can i free memory allocated to an array if I have finished using it in
>> the program? suppose
>> arr_a = fltarr(10000,10000)
>> if i don't need array arr_a anymore, would
>> arr_a = 0
>> free the memory used by arr_a? I would expect it to now utilize only
>> 2(or 4) bytes of memory.
>
> The answer to the question is "yes," but also "it depends." The
> answer is "yes" because the memory is indeed made available again to
> the same IDL process. I use
>
>   A = 0
>
> every day when I am done with a variable. It's totally legit. Hong
> may need go no further than this, although I agree that he's dealing

> with extremely large arrays which might be better suited with a tiled
> or chunked approach.
>
> The answer is also "it depends," because it depends on which platform
> you are using. I believe that under Windows the memory is actually
> returned to the OS. Thus, other programs are able to use the memory
> again. Under Unix this is much less likely, so once the memory is
> allocated to one session of IDL, it stays there until the session ends
> (and is not available to another process). [I think this is not
> *always* true, but mostly true.]
>
> Every so often somebody asks, "why is IDL eating all my memory?" It's
> basically unavoidable. Sorry for this diversion.
>
> Craig
>
> --
> ---------- ----------
> Craig B. Markwardt, Ph.D. EMAIL: craigmnet@cow.physics.wisc.edu
> Astrophysics, IDL, Finance, Derivatives | Remove "net" for better response
> ---------- ----------

Just curious - why not 'delvar,a'? Is this not as effective (or more so) than
'a=0'?

Subject: Re: about memory
Posted by davidf on Tue, 22 May 2001 14:52:38 GMT
View Forum Message <> Reply to Message

Med Bennett (mbennett@indra.com) writes:

> Just curious - why not 'delvar,a'? Is this not as effective (or more so) than
> 'a=0'?

DELVAR can only be used at the IDL main level. Another
alternative is to use something like my UNDEFINE program,
which can be used anywhere and has the effect of undefining
a program variable.

http://www.dfanning.com/programs/undefine.pro

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: about memory
Posted by Craig Markwardt on Tue, 22 May 2001 16:21:10 GMT
View Forum Message <> Reply to Message

davidf@dfanning.com (David Fanning) writes:
> DELVAR can only be used at the IDL main level. Another
> alternative is to use something like my UNDEFINE program,
> which can be used anywhere and has the effect of undefining
> a program variable.
>
> http://www.dfanning.com/programs/undefine.pro

I use UNDEFINE all the time. It's great. When I'm writing an exportable program, where I'm not sure that UNDEFINE will be available, then I use the essence of the algorithm:

a = 0 & dummy = temporary(a)

Then A is undefined and the memory is unallocated.

Craig

--
---------- ----------
Craig B. Markwardt, Ph.D. EMAIL: craigmnet@cow.physics.wisc.edu
Astrophysics, IDL, Finance, Derivatives | Remove "net" for better response
---------- ----------