
Subject: freeing memory in programs

Posted by [pjclinch](#) on Thu, 23 Mar 1995 15:03:55 GMT

[View Forum Message](#) <> [Reply to Message](#)

I'm writing a Wave Widgets program that uses some big arrays (256 * 256 * 128 MRI image stacks). Some operations require temporary arrays to hold significant chunks of this, but I don't want these arrays around full time to choke the memory. I can't use DLEVAR as I'm not at the main program level, so how do I free up memory? If I had assigned an array by, say

```
TmpArr = MAKE_ARRAY(256,256,128, /INT)
```

would a subsequent use of something like TmpArr = BYTE(0) or TmpArr = MAKE_ARRAY(1) free up the memory, or would it still be taken up but no longer accessible?

Curiously, Pete.

--

Peter Clinch Dundee Teaching Hospitals NHS Trust
voice: 44 1382 660111 x 3637 snail: Directorate of Medical Physics
fax: 44 1382 640177 Ninewells Hospital
email: p.j.clinch@dundee.ac.uk Dundee DD1 9SY Scotland UK

Subject: Re: Freeing memory

Posted by [rigby](#) on Thu, 12 Sep 1996 07:00:00 GMT

[View Forum Message](#) <> [Reply to Message](#)

In article 31DF@irc.chmcc.org, Phil Williams <williams@irc.chmcc.org> () writes:

```
>  
> I have a function that reads data and does some base processing. As the  
> procedure is running I monitor my system memory using SGI's sysmon.  
> After the function is complete I use some % of memory.  
>  
> IDL> help,/mem  
> heap memory in use: 13710221, calls to MALLOC: 25781, FREE: 25676  
>  
> I then save all the data to a file and quit idl.  
>  
> I then start idl and restore the data and compile the function and there  
> is less memory used.  
>  
> IDL> help,/mem  
> heap memory in use: 13709644, calls to MALLOC: 119, FREE: 14  
>  
> My question is why?
```

I don't know if it explains your observations, but the IDL command history buffer uses up heap memory. Notice what "help,/mem" returns as the history list gets longer.

--

Wayne Rigby
GE Corporate Research and Development
rigby@crd.ge.com

Subject: Re: Freeing memory
Posted by [Christian Soeller](#) on Fri, 13 Sep 1996 07:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

Phil Williams <williams@irc.chmcc.org> writes:

> My question is why? And why is there very little discrepancy in the
> help,/mem for the heap used? The sysmon looks like alot more than a 1%
> difference?

It is because of the way that memory allocation in a process under unix works.
Following example, from an freshly started idl process:

```
IDL> i=fltarr(1000000)
IDL> help,/mem
heap memory in use:  4075014, calls to MALLOC:    138, FREE:    36
```

ps -el for the process:

```
30 S 1115 21705 21701 0 26 20 * 3070:1673 8838e3e0 pts/0 0:01 idl
i.e. Unix tells you that the process uses 3070 blocks of memory (1673 in
physical RAM)
```

But then say in IDL

```
IDL> i=0
IDL> help,/mem
heap memory in use:  74976, calls to MALLOC:    140, FREE:    37
```

Now ps -el tells you

```
30 S 1115 21705 21701 0 26 20 * 3070:1676 8838e3e0 pts/0 0:01 idl
i.e. still the same amount of memory is used by the process, even more is now
in physical RAM.
```

The paradox is resolved by knowing that memory that has been returned by IDL (internally using some equivalent of the free (3C) function from the allocation library) is not(!) being returned to the operating system but merely put in the list of free blocks maintained by the allocation library routines. Here is what the info of the GNU allocation library tells you:

- > Occasionally, free can actually return memory to the operating system
- > and make the process smaller. Usually, all it can do is allow a later
- > later call to malloc to reuse the space. In the meantime, the space
- > remains in your program as part of a free-list used internally by
- > malloc.

So, if you have a temporary computation that uses a *huge* amount of memory you will still see that in sysmon after the associated memory has been internally freed and the heap memory in IDL looks much smaller. Another related issue is memory fragmentation...

Obviously, if you restart the process from scratch and restore only the saved variables you won't incur these costs.

Regards,

Christian

Christian Soeller mailto: csoelle@sghms.ac.uk
St. Georges Hospital Medical School Dept. of Pharmacology
Cranmer Terrace London SW17 0RE
