
Subject: Re: Memory Leakage.....(I think)
Posted by [David Fanning](#) on Tue, 30 Apr 2002 12:28:01 GMT
[View Forum Message](#) <> [Reply to Message](#)

Andrew Fielding (a.fielding@icr.ac.uk) writes:

> I am running IDL 5.5 under Windows 95 and have following problem. I
> have a loop that is repeated a large number of times and the loop
> contains a series of floating point multiplications. The problem is
> that the pc grinds slowly to a halt as the number of iterations
> increases, eventually, stopping completely. Is there a known operating
> system issue with not releasing memory or is this a programming error
> on my part.

It's a programming error. Oh, wait... Windows 95!?
It's *probably* a programming error. :-)

Cheers,

David

--

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

Subject: Re: Memory Leakage.....(I think)
Posted by [Mark Hadfield](#) on Tue, 30 Apr 2002 21:32:58 GMT
[View Forum Message](#) <> [Reply to Message](#)

"David Fanning" <david@dfanning.com> wrote in message
news:MPG.17383bc056468d09898b2@news.frii.com...

> Andrew Fielding (a.fielding@icr.ac.uk) writes:
>
>> I am running IDL 5.5 under Windows 95 and have following
>> problem. ...
>
> It's a programming error. Oh, wait... Windows 95!? It's *probably*
> a programming error. :-)

I vote for the programming error too. Andrew, can you please tell us
more about what you're doing. Post the code!

--

Mark Hadfield "Ka puwaha et tai nei, Hoesa tatou"

Subject: Re: Memory Leakage.....(I think)

Posted by [Peter Scarth](#) on Tue, 30 Apr 2002 23:42:16 GMT

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

Hi Andrew,

I have experienced a similar problem a while back with some code involving a large number of loops processing a large image. IDL would exit after a couple of hours on my win2k system. I never worked out exactly why it would just stop (I never wanted to wait 4 hours for it to see if it was memory or resources related), but removing the following two lines from my code fixed it:

```
caldat,st+(systime(1,/julian)-st)*100./(50.*((n-1.)*r+i)/r), mon,day,yr,hr,mi  
,sec  
progressBar->SetLabel, "Completion Time>  
"+strtrim(string(hr),2)+" ":"+strtrim(string(mi),2)
```

This just put up an estimate of when the code would complete. I *know* the problem couldn't be in David's progress bar. I'm pretty sure the problem was in the systime call - I was calling it way too often. I suspect this is a windows problem, but I don't know...

I also had heaps of problems in win95 a few years back when I was using a Logitech scroll mouse. After about 15 mins of scrolling in IDL I'd run out of system resources. Weird.

Cheers,
Peter.

"Andrew Fielding" <a.fielding@icr.ac.uk> wrote in message
news:3d9ed221.0204300344.3be3a759@posting.google.com...

> I am running IDL 5.5 under Windows 95 and have following problem. I
> have a loop that is repeated a large number of times and the loop
> contains a series of floating point multiplications. The problem is
> that the pc grinds slowly to a halt as the number of iterations
> increases, eventually, stopping completely. Is there a known operating
> system issue with not releasing memory or is this a programming error
> on my part.

Subject: Re: Memory Leakage.....(I think)

Posted by [a.fielding](#) on Wed, 01 May 2002 12:03:51 GMT

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

Thanks for the comments regarding my problem. Someone suggested I post the code to clarify the problem so here it is.

This is a loop within a sub-procedure that is called (256*256) times from the main procedure. The count variable controlling the loop varies from a minimum of 0 up to 510 depending on the call. I am fairly certain the problem is with the arithmetic within the loop as removing it and replacing it with a simple statement results in the main program processing in around 2 minutes! Any hints/suggestions gratefully received.

Andrew

```
; Weighted Interpolation voxel density calculation
```

```
fval=0.0
sum=0.0

for i=0L,count-1 do begin
  index=ishft(X1,-14) + row[ishft(Y1,-14)] + slice[ishft(Z1,-14)]
  val=ctcube[index]
  if (val gt config.air) then begin
    d0= X1/SCALE - fix(X1/SCALE)
    d1= Y1/SCALE - fix(Y1/SCALE)
    d2= Z1/SCALE - fix(Z1/SCALE)
    fval=(1.0-d0)*(1.0-d1)*(1.0-d2)*float(val)
    fval=temporary(fval) + d0*(1.0-d1)*(1.0-d2)*float(ctcube[index+1])
    fval=temporary(fval) +
(1.0-d0)*d1*(1.0-d2)*float(ctcube[index+xdim])
    fval=temporary(fval) + d0*d1*(1.0-d2)*float(ctcube[index+1+xdim])
    index=temporary(index) + xdim*ydim
    fval=temporary(fval) + (1.0-d0)*(1.0-d1)*d2*float(ctcube[index])
    fval=temporary(fval) + d0*(1.0-d1)*d2*float(ctcube[index+1])
    fval=temporary(fval) + (1.0-d0)*d1*d2*float(ctcube[index+xdim])
    fval=temporary(fval) + d0*d1*d2*float(ctcube[index+1+xdim])

    val= fix(fval)
    sum=temporary(sum) + convert_ct2mu[val-(Hmin-1024)]
  endif

  X1=X1+dX
  Y1=Y1+dY
  Z1=Z1+dZ
endfor
```

Subject: Re: Memory Leakage.....(I think)
Posted by [Rick Towler](#) on Wed, 01 May 2002 16:11:22 GMT

Have you tried removing the calls to the temporary function?

-Rick

"Andrew Fielding" <a.fielding@icr.ac.uk> wrote in message
news:3d9ed221.0205010403.406fe470@posting.google.com...

> Thanks for the comments regarding my problem. Someone suggested I post
> the code to clarify the problem so here it is.

> This is a loop within a sub-procedure that is called (256*256) times
> from the main procedure. The count variable controlling the loop
> varies from a minimum of 0 up to 510 depending on the call. I am
> fairly certain the problem is with the arithmetic within the loop as
> removing it and replacing it with a simple statement results in the
> main program processing in around 2 minutes! Any hints/suggestions
> gratefully received.

>

> Andrew

>

> ; Weighted Interpolation voxel density calculation

>

> fval=0.0

> sum=0.0

>

> for i=0L,count-1 do begin

> index=ishft(X1,-14) + row[ishft(Y1,-14)] + slice[ishft(Z1,-14)]

> val=ctcube[index]

> if (val gt config.air) then begin

> d0= X1/SCALE - fix(X1/SCALE)

> d1= Y1/SCALE - fix(Y1/SCALE)

> d2= Z1/SCALE - fix(Z1/SCALE)

> fval=(1.0-d0)*(1.0-d1)*(1.0-d2)*float(val)

> fval=temporary(fval) + d0*(1.0-d1)*(1.0-d2)*float(ctcube[index+1])

> fval=temporary(fval) +

> (1.0-d0)*d1*(1.0-d2)*float(ctcube[index+xdim])

> fval=temporary(fval) + d0*d1*(1.0-d2)*float(ctcube[index+1+xdim])

> index=temporary(index) + xdim*ydim

> fval=temporary(fval) + (1.0-d0)*(1.0-d1)*d2*float(ctcube[index])

> fval=temporary(fval) + d0*(1.0-d1)*d2*float(ctcube[index+1])

> fval=temporary(fval) + (1.0-d0)*d1*d2*float(ctcube[index+xdim])

> fval=temporary(fval) + d0*d1*d2*float(ctcube[index+1+xdim])

>

> val= fix(fval)

> sum=temporary(sum) + convert_ct2mu[val-(Hmin-1024)]

> endif

>

> X1=X1+dX

```
> Y1=Y1+dY  
> Z1=Z1+dZ  
> endfor
```
