Subject: Re: allocating memory
Posted by R.G.Stockwell on Thu, 18 Oct 2007 15:37:26 GMT
View Forum Message <> Reply to Message

<jtmcahill@gmail.com> wrote in message
news:1192673434.544923.139390@i38g2000prf.googlegroups.com.. .
> So, I have this large model file that I need to open. Although it is
> 650MB I should be able to open it in IDL on my pc computer which has
> ~4GB of Memory. However, it keeps telling me insufficient memory.
> However, if I try to open it in IDL on a linux machine with ~2GB
> memory I can open it no problem. Is there a way to make my windows
> based pc cooperate and allow me to open this file that should be no
> problem to open?
>
> Thanks,
> Hawaiianite

I've attached a short program memtest.pro below. I grabbed this
off the newsgroup.

It shows you the memory sizes you can allocate.

The problem may be fragmentation of your ram by the many
dlls loaded by windows and other programs. I'd remove everything
you can from the startup (and other automatically loading programs)
and reboot (remove spyware, antivirus, firewalls, mail programs, but
be careful not to forget to turn them back on). That may help.

```
pro memtest
  compile_opt idl2 ; set default integers to 32-bit and enforce [] for
indexing

  MB = long64(2)^20
  currentBlockSize = MB * 2047   ; 2 GB

print,'current block size = ',currentblocksize
  maxIterations = 10            ; Max loop iterations
  memPtrs = ptrarr(maxIterations)
  memBlockSizes = ulonarr(maxIterations)

  for i=0, maxIterations-1 do begin
  ; Error handler
    catch, err

    ; Sepcifically designed for "Failure to allocate memory..." error
    if (err ne 0) then begin
```

```
      currentBlockSize = currentBlockSize - MB     ; ...try 1 MB smaller
allocation
      if (currentBlockSize lt MB) then break  ; Give up, if currentBlockSize
< 1 MB
    endif

  ; This 'wait' enables Ctrl-Break to interrupt if necessary (Windows).
    wait, 0.0001

  ; Allocate memory (if possible)
    memPtrs[i] = ptr_new(bytarr(currentBlockSize, /nozero), /no_copy)
    memBlockSizes[i] = currentBlockSize   ; Store the latest successful
allocation size

  ; Print the current allocated block size and the running total, in Mb
    print, format='(%"Memory block #%2d: %6d Mb (total: %4d Mb)")', $
      i + 1, ishft(currentBlockSize, -20),
ishft(ulong(total(memBlockSizes)), -20)
  endfor

  ptr_free,memPtrs
end
```