## Subject: Re: What in the world does "Program code area full" mean?
## Posted by steve.kaeppler on Wed, 12 Aug 2009 21:08:21 GMT

View Forum Message <> Reply to Message

On Aug 12, 4:03 pm, "steve.kaeppler" <steve.kaepp...@gmail.com> wrote:

> Hi Everyone-
>
> I am still learning all sorts of new tricks in IDL, but I am running
> into some trouble with IDL.  I am running version 6.4, which may be
> useful.  I am getting an error in my code that says "Program code area
> full."  I will say I am running a fairly beefy code with respect to a
> lot of while loops,  if statements and creation of arrays to be
> printed to text documents.  I am more than willing to put up the code
> if it is helpful, but for now, I just want to understand what exactly
> this means and if there is some way to basically by-pass this error?
>
> If someone could help me, I would be very grateful!
>
> Kindest Regards,
> Steve Kaeppler

Actually, I will attach the code.

The problem happens when I uncomment ;c_av = reform(c_dot_calc,[1,47])
&$ or print, c_dot_calc  As soon as I do that, I get this "Program
code Area is full"  I do not, for the life of me, understand why it is
doing that?  Is something undefined?  Is too much memory being taken
up?

Also any streamlining suggestions are always appreciated!

Thanks all!
Steve

the code:

```
;;; process for real mag pa

base_dir = '/data0/rocket/aces_21139/outfiles/' ;##################
user define
srk_dir = '/home/srk/aces_data/'

; this file is meant to be a fast way to get the correct files needed
to create the chi squared
```

```
.com '/home/srk/aces_data/eepaa/eepaa_read_cdf_flight.pro'
.com '/home/srk/aces_data/eepaa/plot_image_colorbar.pro'

outname = '21139_EEPAA_Electron' ;################### user define
cdf_file = srk_dir+'eepaa/'+outname+'.cdf'

data = eepaa_read_cdf_flight(cdf_file)

;help, data, /st
device, decompose=0

file1 = srk_dir+'/eepaa/spectrogram/outfiles/ref_10_20.dat' ; usr
definied
openw, lun1, file1, /get_lun

file3 = srk_dir+'/eepaa/spectrogram/outfiles/
ACES_Low_data_plot_stats.dat' ; usr definied
openw, lun2, file3, /get_lun



file2 = srk_dir+'/eepaa/spectrogram/chi_sq_results_july2009.csv'
corr_fact = read_ascii(file2,delimiter=',',data_start=2); chi squared
correction factor
help, corr_fact, /st


neg_hun = fltarr([1,47])
neg_hun[0:46] = -100.
acq_time = data.acq_time * 1.e-3 ; convert to seconds
dt_ex = data.dead_time*1.e-9

A = reform(fltarr(50), [1,50])


zero_47 = reform(fltarr(47))
zero_47[0:46] = 0
c_dot_calc = fltarr(47)
i = 0 ; selects channel of interest
j = long(0)
num_rec = data.total_num


c_dot_1 = fltarr(47)
c_dot_2 = fltarr(47)
c_dot_3 = fltarr(47)

chi_1 = fltarr(47)
```

```
chi_2 = fltarr(47)
chi_3 = fltarr(47)
tmp_av = fltarr(47)

; nom pa
;while(t lt 24) do begin &$

pa_high =45. &$
pa_low =35. &$
print, '1' &$
while(j lt num_rec) do begin &$ ; over all records

;if(max(data.time[*,*,j]) gt 250.) then begin &$

;print, where( (data.measured_pa[*,*,j] ge pa_low) and
(data.measured_pa[*,*,j] le pa_high) ) &$
q = where( (data.measured_pa[*,*,j] ge pa_low) and (data.measured_pa
[*,*,j] le pa_high) ) &$

if(q[0] eq -1) then begin &$
;print, [-1, j] &$
; this will give us the locations where we have no data values - no
hit mag hits
time_ = max(data.time[*,*,j]) &$
A = [A,reform([j,time_, 'NaN', reform(neg_hun, [47])], [1,50]) ] &$



endif else begin &$
; this is where I will need to average and all that
;print, [q, j] &$

n = n_elements(q) &$

if( n eq 1) then begin &$
;print, '1' &$
while( k lt 47 ) do begin &$
if(data.dac_sweep[0,k,j] ne 0) then begin &$

c_dot_1[k] = (corr_fact.field01[q[0]]*data.eepaa1[q[0],k,j])/(acq_time
- (corr_fact.field01[q[0]]*data.eepaa1[q[0],k,j]*dt_ex[i] ) )  &$
;c_dot_calc = reform(c_dot_calc,[47]) &$
endif else begin &$
c_dot_1[k] = -100. &$
endelse &$
k++ &$
endwhile &$
```

```
c_dot_calc = c_dot_1 &$
mean_pa = data.measured_pa[q[0],*,j] &$
endif &$
k =0 &$



if( n eq 2) then begin &$
;print, '2' &$
while( k lt 47 ) do begin &$
if(data.dac_sweep[0,k,j] ne 0) then begin &$

c_dot_1[k] = (corr_fact.field01[q[0]]*data.eepaa1[q[0],k,j])/(acq_time
- (corr_fact.field01[q[0]]*data.eepaa1[q[0],k,j]*dt_ex[q[0]] ) )  &$
c_dot_2[k] = (corr_fact.field01[q[1]]*data.eepaa1[q[1],k,j])/(acq_time
- (corr_fact.field01[q[1]]*data.eepaa1[q[1],k,j]*dt_ex[q[1]] ) )  &$
;c_dot_calc = reform(c_dot_calc,[47]) &$
endif else begin &$
c_dot_1[k] = -100. &$
c_dot_2[k] = -100. &$
endelse &$
k++ &$
endwhile &$
k =0 &$
while(k lt 47) do begin &$
;c_dot_calc[k] = 0. &$
c_dot_calc[k] = mean([c_dot_1[k],c_dot_2[k]]) &$
mean_pa = mean([data.measured_pa[q[0],*,j],data.measured_pa[q
[1],*,j] ])  &$

if(c_dot_1[k] ne 0.) then begin &$
chi_1[k] =   ( (c_dot_1[k] - c_dot_calc[k])*(c_dot_1[k] - c_dot_calc
[k]) )/c_dot_1[k]&$
endif &$

if(c_dot_2[k] ne 0.) then begin&$
chi_2[k] = ( (c_dot_2[k] - c_dot_calc[k])*(c_dot_2[k] - c_dot_calc
[k]) )/c_dot_2[k] &$
endif &$

k++ &$
endwhile &$
endif &$
k =0 &$


if( n eq 3) then begin &$
```

```
;print, '3' &$
while( k lt 47 ) do begin &$
if(data.dac_sweep[0,k,j] ne 0) then begin &$

c_dot_1[k] = (corr_fact.field01[q[0]]*data.eepaa1[q[0],k,j])/(acq_time
- (corr_fact.field01[q[0]]*data.eepaa1[q[0],k,j]*dt_ex[q[0]] ) )  &$
c_dot_2[k] = (corr_fact.field01[q[1]]*data.eepaa1[q[1],k,j])/(acq_time
- (corr_fact.field01[q[1]]*data.eepaa1[q[1],k,j]*dt_ex[q[1]] ) )  &$
c_dot_3[k] = (corr_fact.field01[q[2]]*data.eepaa1[q[2],k,j])/(acq_time
- (corr_fact.field01[q[2]]*data.eepaa1[q[2],k,j]*dt_ex[q[2]] ) )  &$
;c_dot_calc = reform(c_dot_calc,[47]) &$
endif else begin &$
c_dot_1[k] = -100. &$
c_dot_2[k] = -100. &$
c_dot_3[k] = -100. &$
endelse &$
k++ &$
endwhile &$
k=0 &$
while(k lt 47) do begin &$

c_dot_calc[k] = (c_dot_1[k]+c_dot_2[k]+ c_dot_3[k])/3. &$
mean_pa = mean([data.measured_pa[q[0],*,j],data.measured_pa[q
[1],*,j],data.measured_pa[q[2],*,j] ])  &$

if(c_dot_1[k] ne 0.) then begin &$
chi_1[k] =   ( (c_dot_1[k] - c_dot_calc[k])*(c_dot_1[k] - c_dot_calc
[k]) )/c_dot_1[k]&$
endif &$

if(c_dot_2[k] ne 0.) then begin&$
chi_2[k] = ( (c_dot_2[k] - c_dot_calc[k])*(c_dot_2[k] - c_dot_calc
[k]) )/c_dot_2[k] &$
endif &$

if(c_dot_3[k] ne 0.) then begin&$
chi_2[k] = ( (c_dot_3[k] - c_dot_calc[k])*(c_dot_3[k] - c_dot_calc
[k]) )/c_dot_3[k] &$
endif &$

k++ &$
endwhile &$
endif &$
k =0 &$



time_ = max(data.time[*,*,j]) &$
```

```
A = [A,reform([j,time_, mean_pa, c_dot_calc], [1,50]) ] &$


print, c_dot_calc &$
;n =0 &$
;help, c_av, /st &$
;help, c_dot_calc &$
c1 = reform(c_dot_1, [1,47]) &$
c2 = reform(c_dot_2, [1,47]) &$
c3 = reform(c_dot_3, [1,47]) &$
;c_av = reform(c_dot_calc,[1,47]) &$


endelse &$ ; close out mag pitch angle

printf, lun2, 'Record: ', j &$
printf, lun2, 'Time: ', time_ &$
printf, lun2, 'Number Channels overlap: ', n &$
printf, lun2, 'Channels Overlap: ', q &$
printf, lun2, 'C1, C2, C3, Cave, Chi_1, Chi_2, Chi_3' &$
printf, lun2, [c1, c2, c3]&$
printf, lun2,
 '------------------------------------------------------------ -----' &$

; diagnostic print statments
;printf, lun1, j,time_,  mean_pa, [c_dot_calc] &$;
[data.measured_energy[0,*,j], data.dac_sweep[0,*,j] ] &$;
[data.measured_pa[i,*,j],
;printf, lun1, '----------------------------------------------------' &
$




q = 0 &$
n =0 &$
c_dot_calc = zero_47 &$
c_dot_1 = zero_47 &$
c_dot_2 = zero_47 &$
c_dot_3 = zero_47 &$
chi_1 = zero_47 &$
chi_2 = zero_47 &$
chi_3 = zero_47 &$
mean_pa = 0 &$
j++ &$
endwhile  &$

print, 'end 1' &$
```

```
A = A[1:*,*]

plot_time = A[*,1] &$
plot_energy = (data.cal_energy/1000.) &$
plot_cdot = A[*,3:49] &$


loadct, 39
set_plot,'ps'
device,/color,bits=8, filename = '/home/srk/aces_data/eepaa/
spectrogram/outfiles/aces_low_spectrogram_pa_test.ps'



title = 'ACES LOW(21.139) Spectrogram'+' '+ strcompress(string
(pa_low)) +' < '+'PA'+' < ' + strcompress(string(pa_high))  &$


plot_image_colorbar, plot_cdot, plot_time,
plot_energy,title=title,xrange=[100,400], xtitle='Time', ytitle='Energy
(keV)',/ylog,  position=pos_img, grey_value = -100.0 &$

;print, t &$
;t++ &$


device, /close
set_plot, 'X'

free_lun, lun1
free_lun, lun2
```

## Subject: Re: What in the world does "Program code area full" mean?

Posted by David Fanning on Wed, 12 Aug 2009 21:11:33 GMT

View Forum Message <> Reply to Message

steve.kaeppler writes:

> I am still learning all sorts of new tricks in IDL, but I am running
> into some trouble with IDL.  I am running version 6.4, which may be
> useful.  I am getting an error in my code that says "Program code area
> full."  I will say I am running a fairly beefy code with respect to a
> lot of while loops,  if statements and creation of arrays to be
> printed to text documents.  I am more than willing to put up the code
> if it is helpful, but for now, I just want to understand what exactly
> this means and if there is some way to basically by-pass this error?

You might try this IDL newsgroup discussion:

   http://tinyurl.com/lkwj9o

P.S. I am guessing the problem is a STOP in the code
somewhere.

Cheers,

David
--
David Fanning, Ph.D.
Coyote's Guide to IDL Programming (www.dfanning.com)
Sepore ma de ni thui. ("Perhaps thou speakest truth.")

## Subject: Re: What in the world does "Program code area full" mean?
## Posted by David Fanning on Wed, 12 Aug 2009 21:18:05 GMT

View Forum Message <> Reply to Message

steve.kaeppler writes:

> Actually, I will attach the code.

Oh dear. Who taught you how to write IDL code. :-(

Yes, IDL is really not designed to run one gigantic line of code like this. You would be MUCH better served if you did a global delete of all "&$" marks and put an END at the end of your code, gave it a name (something like "disaster.pro" would be OK), and then "ran" it like this:

```
   IDL> .run disaster
```

This would be called a main-level program. What you are trying to run (I think) is a batch program. And it is too big to be interpreted in one go. It will need to be compiled.

But, there are MUCH better ways to write IDL code than what you have here. :-)

Cheers,

David

--
David Fanning, Ph.D.
Coyote's Guide to IDL Programming (www.dfanning.com)
Sepore ma de ni thui. ("Perhaps thou speakest truth.")

---

## Subject: Re: What in the world does "Program code area full" mean?
## Posted by steve.kaeppler on Wed, 12 Aug 2009 21:33:37 GMT

View Forum Message <> Reply to Message

Understand, David, some of us are picking it up as we go here. :)
That is all part of being a graduate student, right? I am writing it
like a script, because that is basically how the other code I know of
has been ran.

I will to write it like an actual program and see if that helps.

Thanks!

Steve

On Aug 12, 4:18 pm, David Fanning <n...@dfanning.com> wrote:
> steve.kaeppler writes:
>> Actually, I will attach the code.
>
> Oh dear. Who taught you how to write IDL code. :-(
>
> Yes, IDL is really not designed to run one gigantic
> line of code like this. You would be MUCH better
> served if you did a global delete of all "&$" marks
> and put an END at the end of your code, gave it
> a name (something like "disaster.pro" would be OK),
> and then "ran" it like this:
>
>    IDL> .run disaster
>
> This would be called a main-level program. What you
> are trying to run (I think) is a batch program. And it
> is too big to be interpreted in one go. It will need
> to be compiled.
>
> But, there are MUCH better ways to write IDL code
> than what you have here. :-)
>
> Cheers,
>
> David
>

> --
> David Fanning, Ph.D.
> Coyote's Guide to IDL Programming (www.dfanning.com)
> Sepore ma de ni thui. ("Perhaps thou speakest truth.")

Subject: Re: What in the world does "Program code area full" mean?
Posted by David Fanning on Wed, 12 Aug 2009 21:54:55 GMT
View Forum Message <> Reply to Message

steve.kaeppler writes:

> Understand, David, some of us are picking it up as we go here. :)
> That is all part of being a graduate student, right?  I am writing it
> like a script, because that is basically how the other code I know of
> has been ran.
>
> I will to write it like an actual program and see if that helps.

It will *definitely* help!

You probably had the right idea on a script. Just the
wrong kind of script there. The kind you were writing
is 1970's era. We are trying to get you up to 1985 today.
And, with luck, we will have you into the 21st century
within the week!

Cheers,

David

P.S. There are always discounts on IDL books
for the asking when you are a student. :-)

--
David Fanning, Ph.D.
Coyote's Guide to IDL Programming (www.dfanning.com)
Sepore ma de ni thui. ("Perhaps thou speakest truth.")

Subject: Re: What in the world does "Program code area full" mean?
Posted by wlandsman on Wed, 12 Aug 2009 22:11:36 GMT
View Forum Message <> Reply to Message

On Aug 12, 5:18 pm, David Fanning <n...@dfanning.com> wrote:
>
> Yes, IDL is really not designed to run one gigantic

> line of code like this. You would be MUCH better
> served if you did a global delete of all "&$" marks
> and put an END at the end of your code, gave it
> a name (something like "disaster.pro" would be OK),
>

Early one in one's IDL programming life, one learns not to write a single long line of code like this. But I only recently appreciated how using EXECUTE() is similar to writing a single line of code -- you better be careful if it gets too long! The problem came up in code I have that uses EXECUTE() to dynamically create a structure. The little code program below will probably fail with a "Code Area Full" at some point -- depending on your machine -- when the execute string becomes too long.

```
pro test
FOR i=100, 600 DO BEGIN
  ;;Get unique names for elements, make sure they are 8 long
  exstring = 'a={' + $
     STRJOIN( STRING(INDGEN(i),FORMAT='("TAG",I05,":0.0d0")'),',') +
'}'
  res = EXECUTE(exstring)
  IF res EQ 0 THEN MESSAGE,"Execute failed on "+STRING(i)
ENDFOR
return
end
```

The problem is not that the structure itself is too large - it can easily be created if one puts the structure definition in a procedure file and compiles it. (This is one of the ways to get around the problem. The other way is to break up the structure definition into chunks and then combine the substructures with CREATE_STRUCT() -- this is the method used by mrd_struct.pro =
http://idlastro.gsfc.nasa.gov/ftp/pro/structure/mrd_struct.p ro )

One thing that confused me was that, in ancient days, EXECUTE() had strict limits on the number of characters. Then sometime in IDL 5.x these limits were removed, giving the impression that one could use arbitarily long strings within EXECUTE(). But in fact it is subject to the same code memory limits as when writing a program as a single line of code. --Wayne

## Subject: Re: What in the world does "Program code area full" mean?
## Posted by Craig Markwardt on Thu, 13 Aug 2009 00:23:24 GMT

View Forum Message <> Reply to Message

On Aug 12, 5:18 pm, David Fanning <n...@dfanning.com> wrote:
> steve.kaeppler writes:

>> Actually, I will attach the code.
>
> Oh dear. Who taught you how to write IDL code. :-(

I have a colleague who writes reams of IDL code like that. He prefers
to cut-paste into the IDL window than to .run it. That's just how he
rolls.

Craig

Subject: Re: What in the world does "Program code area full" mean?
Posted by David Fanning on Thu, 13 Aug 2009 02:39:18 GMT
View Forum Message <> Reply to Message

Craig Markwardt writes:

> I have a colleague who writes reams of IDL code like that. He prefers
> to cut-paste into the IDL window than to .run it. That's just how he
> rolls.

Well, yeah, but you work for a government agency... ;-)

Cheers,

David

--
David Fanning, Ph.D.
Fanning Software Consulting, Inc.
Coyote's Guide to IDL Programming: http://www.dfanning.com/
Sepore ma de ni thui. ("Perhaps thou speakest truth.")