Subject: Re: SOS with Idl
Posted by anaisgcristal@gmail.c on Tue, 21 Aug 2007 18:02:11 GMT
View Forum Message <> Reply to Message

Hi Jean, The mess i get it's:

% Out of range subscript encountered: MATRIZ.
% Execution halted at: P 119
/home/iquique/anais/Codigos/Gadget/resultados/Idl/PlotPowerS pec.pro
% $MAIN$

And when I tried to printf in prompt the matriz value, it's a secuence of numbers (in files) not in arrow! I just learnig to program, and I suposse the problem it's some bad definition or something similar...

On Aug 21, 6:01 pm, "Jean H." <jghas...@DELTHIS.ucalgary.ANDTHIS.ca> wrote:
> anaisgcris...@gmail.com wrote:
>> Hi, I just learnig programming with Idl, and I've been trying to
>> modifided a file to print a array 4X27 (matriz), but I can't print
>> this array in a file! I suposse the problems it's some definition or
>> something similar, but I didn't found it. It's posible that someone
>> can check the code and help me to found the bug?
>> Thanks a lot
>
> What do you get exactly?
>
> Jean
>
>> Quote:
>> pro p
>> fname = "/home/iquique/anais/Codigos/Gadget/resultados/w_066/
>> powerspec_004.txt"
>> openr, 1, fname
>
>> Time = 0.0
>> Bins = 0L
>> readf, 1, Time
>> readf, 1, Bins
>> da1= fltarr(10, bins)
>> readf, 1, da1
>> readf, 1, Time
>> readf, 1, Bins
>> da2= fltarr(10, bins)
>> matriz = fltarr(2, 27)
>> readf, 1, da2
>> close,1

```
>
>>  K_A = da1(0,*)
>>  Delta2_A = da1(1,*)
>>  Shot_A = da1(2,*)
>>  ModePow_A = da1(3,*)
>>  ModeCount_A = da1(4,*)
>>  Delta2Uncorrected_A = da1(5,*)
>>  ModePowUncorrected_A = da1(6,*)
>>  Specshape_A = da1(7,*)
>>  SumPower_A = da1(8,*)
>>  ConvFac_A = da1(9,*)
>
>>  K_B = da2(0,*)
>>  Delta2_B = da2(1,*)
>>  Shot_B = da2(2,*)
>>  ModePow_B = da2(3,*)
>>  ModeCount_B = da2(4,*)
>>  Delta2Uncorrected_B = da2(5,*)
>>  ModePowUncorrected_B = da2(6,*)
>>  Specshape_B = da2(7,*)
>>  SumPower_B = da2(8,*)
>>  ConvFac_B = da2(9,*)
>
>>  ; we will do a band averaging of the finely binned points,
>>  ; subject to two conditions:
>>  ; We want enough modes per bin in order to reduce the variance in a
>>  bin,
>>  ; and simultaneously, for large k, we don't want the bins become too
>>  narrow.
>>  ;
>>  ; The first condition is set by "MinModeCount",
>>  ; the second by "TargetBinNummer", which is used to compute a minimum
>>  ; logarithmic bin-size.
>
>>  MinModeCount = 20
>>  TargetBinNummer = 60
>
>>  MinDlogK = (alog10(max(K_A)) - alog10(min(K_A)))/TargetbinNummer
>
>>  istart=0
>>  ind=[istart]
>>  k_list_A = [0]
>>  Delta2_list_A = [0]
>>  count_list_A = [0]
>>  for j = 0, 27-1 do begin
>>  repeat begin
>>  count = total(modecount_a(ind))
>>  deltak = (alog10(max(K_A(ind))) - alog10(min(K_A(ind))))
```

```
>
>>  if (deltak ge mindlogk) and (count ge MinModeCount) then begin
>>  d2 = total(SumPower_A(ind))/total(ModeCount_A(ind))
>>  b =  fix(total(double(ind)*ModeCount_A(ind))/total(ModeCount_A(in d)))
>>  kk = K_A(b)
>>  d2 = ConvFac_A(b)*d2*Specshape_A(b)
>>  k_list_A = [k_list_A, kk]
>>  Delta2_list_A = [Delta2_list_A, d2]
>>  count_list_A = [count_list_A, total(ModeCount_A(ind))]
>>  istart = istart + 1
>>  ind = [istart]
>>  endif else begin
>>  istart = istart + 1
>>  ind = [ind, istart]
>>  endelse
>>  endrep until istart ge Bins
>>  K_list_A = k_list_A(1:*)
>>  Delta2_list_A = delta2_list_A(1:*)
>>  Count_list_A = count_list_A(1:*)
>>  [b]matriz(0, j) = Delta2_list_A
>>  matriz(1,j) = Delta2_list_B
>>  end
>>  openw, 5, 'mypk.dat'
>>  for j= 0, 27 -1 do begin
>>  printf, 5, matriz(*, j)
>>  print, matriz(*, j)
>>  end
>>  close, 5
>>  end
```