
Subject: Re: Logic problem

Posted by [ben.bighair](#) on Thu, 17 Jan 2008 01:33:24 GMT

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

On Jan 16, 4:42 pm, "Ryan." <rchug...@brutus.uwaterloo.ca> wrote:

> Hi All,

>

> I need assistance with the logic of a particular routine to retrieve a
> list of files by date. I have tried a few methods but they haven't been
> successful. I'm posting here to hopefully get some assistance on the
> logic that I'm using. I can't seem to get it correct. Here are the
> details:

>

> I have a directory of folders (As you can see the folder names
> correspond to a time period that overlap):

> 2004-02-19_2004-02-20

> 2004-02-20_2004-02-21

> 2004-02-21_2004-02-25

> 2004-02-25_2004-02-28

> 2004-03-06_2004-03-10

> 2004-03-10_2004-03-13

> ...

>

> Each of these folders contains various files, I am interested in
> obtaining one with a particular extension, say *.SAS (I should note,
> that sometimes this file does not exist within the folder).

>

> I want to have a special routine that given a date (or a start and end
> date) to return the full path of the *.SAS file(s) or if it doesn't
> exist to print out a statement saying it doesn't exist. If the desired
> dates are spread over 2 or more folders I want it to return all the
> paths of the files. If the date desired lands on the overlapping part
> of the date (e.g. 2004-02-20 in sample folders above), I want it to
> return the path of the two files.

>

> Here are some examples of what I would like returned (using the list of
> folders above):

>

> IDL> print, findsasfiles(JULDAY(2,19,2004))

> full_path/2004-02-19_2004-02-20/file.SAS

>

> IDL> print, findsasfiles(JULDAY(2,20,2004))

> full_path/2004-02-19_2004-02-20/file.SAS

> full_path/2004-02-20_2004-02-21/file.SAS

>

> IDL> print, findsasfiles(JULDAY(2,19,2004), JULDAY(2,20,2004))

> full_path/2004-02-19_2004-02-20/file.SAS

> full_path/2004-02-20_2004-02-21/file.SAS

```

>
> IDL> print, findsasfiles( JULDAY(2,20,2004), JULDAY(2,25,2004) )
> full_path/2004-02-19_2004-02-20/file.SAS
> full_path/2004-02-20_2004-02-21/file.SAS
> full_path/2004-02-21_2004-02-25/file.SAS
> full_path/2004-02-25_2004-02-28/file.SAS
>
> Here is the function so far:
>
> FUNCTION findsasfiles, date, enddate, MISSING=missing, $
>     NMISSING=nmissing
>
> ;Get directory names:
> ;directory of SAS files
> sasdir = FILEPATH("", ROOT_DIR=rch_getrootdir(),
> SUBDIRECTORY=['plan'])
>
> ;Getting the list of folder names in 'sasdir' directory
> dirlist = FILE_SEARCH(sasdir+'*', COUNT=nDirs, /FULLY_QUALIFY_PATH, $
> /TEST_DIRECTORY)
>
> ;Remove erroneous directory names:
> ;make sure no extra folder names are found except for the one
> ;corresponding to a time span
> p = STRPOS(dirlist, '200')
> diridx = WHERE(p+1, ndirs)
> IF ndirs GT 0 THEN dirlist = STRMID(dirlist[WHERE(p+1)], 1#p) $
> ELSE dirlist = STRMID(dirlist, 1#p)
>
> ;some needed constants
> dirs = STRARR(nDirs)
> dirdates = DBLARR(nDirs, 2)
>
> ;Extract dates from folder names:
> FOR i = 0L, ndirs-1 DO BEGIN
>     splitdir = STRSPLIT(dirlist[i], PATH_SEP(), COUNT=slashcnt)
>
>     ;retrieve directory name
>     dirs[i] = STRMID(dirlist[i], splitdir[slashcnt-1])
>
>     ;retrieve folder name and dates
>     datesplit = STRSPLIT(dirs[i], '_', /EXTRACT)
>     dirdates[i,0] = JULDAY(STRMID(datesplit[0],5,2),
> STRMID(datesplit[0],8,2), $
> STRMID(datesplit[0],0,4))
>     dirdates[i,1] = JULDAY(STRMID(datesplit[1],5,2),
> STRMID(datesplit[1],8,2), $
> STRMID(datesplit[1],0,4))

```

```

> ENDFOR
>
> ,***** NEED HELP AFTER THIS POINT *****
> ;Find dates that are searched for:
> idx = WHERE(dirdates[,0] GE date, startcnt)
> IF startcnt LT 1 THEN BEGIN
>   PRINT, 'Start Date Not Found. Returning...'
>   RETURN, -1S
> ENDIF
>
> IF endcorrect THEN BEGIN
>   endidx = WHERE(dirdates[,1] GT enddate, endcnt)
>   IF (endcnt GT 0) THEN BEGIN
>     idx = [idx[0], endidx[0]]
>     folders = dirlist[idx[0]:idx[1]]
>   ENDIF
> ENDIF ELSE folders = dirlist[idx[0]]
>
> ;Discover if SAS file exists and return it if it does:
> nFiles = FIX(N_ELEMENTS(folders))
> files = STRARR(nFiles)
> missing = STRARR(nFiles)
> sascounter = 0S
> nMissing = 0S
>
> FOR j=0L, nFiles-1 DO BEGIN
>   sasfind = FILE_SEARCH(sasdir+folders[j], '*.SAS',
> COUNT=sasfindcount, /FULLY_QUALIFY_PATH)
>   IF (sasfindcount GT 0) THEN BEGIN
>     files[sascounter] = sasfind
>     sascounter += 1
>   ENDIF ELSE BEGIN
>     PRINT, 'No SAS file found in folder: '+folders[j]
>     missing[nMissing] = folders[j]
>     nMissing += 1
>   ENDELSE
> ENDFOR
>
> CASE 1 OF
>   (nMissing EQ 0) AND (sascounter EQ 0): BEGIN
>     files = -1S
>     missing = -1S
>   END
>   (nMissing GT 0) AND (sascounter EQ 0): BEGIN
>     files = -1S
>     print, 'No files found'
>   END
>   (nMissing EQ 0) AND (sascounter GT 0): BEGIN

```

```

>     missing = -1$
>     print, 'All files found'
>     END
>     ELSE: BEGIN
>         files = files[0:(sascounter-1)]
>         missing = missing[0:(nMissing-1)]
>         PRINT, 'A bit of this, a bit of that'
>     END
>     ENDCASE
>
> ,*****
>
>     RETURN, files
>
> END
>
> -----
> The routine as it stands doesn't quite work because the indices returned
> (the final 'idx' variable) can be backwards (eg. [175,174]) and thus
> cause an error when trying to execute the line "folders =
> dirlist[idx[0]:idx[1]]".
>
> To me it doesn't seem like an overly difficult problem but I've spent
> the last 2 or 3 days trying to get it right with no success. I need
> some new minds to help me with this.
>
> Thanks,
> Ryan.

```

Hi Ryan,

This doesn't address the whole issue but the following picks the right folder name(s).

Cheers,
Ben

PRO RYAN

```

;split the names up into pieces
;and convert to start/stop dates
names = ['2004-02-19_2004-02-20',$
'2004-02-20_2004-02-21',$
'2004-02-21_2004-02-25',$
'2004-02-25_2004-02-28',$
'2004-03-06_2004-03-10',$
'2004-03-10_2004-03-13']
n = n_elements(names)

```

```

ymd = STRARR(6,n) ;an array for the split up names
ss = LONARR(2,n) ; an array for the julian start/stop dates
for i = 0, n-1 do begin
  ymd[* ,i] = strsplit(names[i], '-_', /extract)
  ss[0,i] = JULDAY(ymd[1,i], ymd[2,i], ymd[0,i])
  ss[1,i] = JULDAY(ymd[4,i], ymd[5,i], ymd[3,i])
endfor

print, "testDate = 2004-02-20"
testDate = JULDAY(2,20,2004)
;divide by two to get the equivalent names indices
A = (WHERE((testDate GE ss[0,*]) AND (testDate LE ss[1,*]), nA))
if nA GT 0 then $
  print, "testDate in ... " , names[a] else $
  print, "testDate not found"

print, "testDate = 2004-03-11"
testDate = JULDAY(3,11,2004)
;divide by two to get the equivalent names indices
A = (WHERE((testDate GE ss[0,*]) AND (testDate LE ss[1,*]), nA))
if nA GT 0 then $
  print, "testDate in ... " , names[a] else $
  print, "testDate not found"

end

```

Subject: Re: Logic problem

Posted by [Craig Markwardt](#) on Thu, 17 Jan 2008 16:51:12 GMT

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

"Ryan." <rchughes@brutus.uwaterloo.ca> writes:

```

> Hi All,
>
> I need assistance with the logic of a particular routine to retrieve a
> list of files by date. I have tried a few methods but they haven't
> been successful. I'm posting here to hopefully get some assistance on
> the logic that I'm using. I can't seem to get it correct. Here are
> the details:
>
> I have a directory of folders (As you can see the folder names
> correspond to a time period that overlap):
> 2004-02-19_2004-02-20
> 2004-02-20_2004-02-21
> 2004-02-21_2004-02-25
> 2004-02-25_2004-02-28
> 2004-03-06_2004-03-10

```

> 2004-03-10_2004-03-13
> ...
>
> Each of these folders contains various files, I am interested in
> obtaining one with a particular extension, say *.SAS (I should note,
> that sometimes this file does not exist within the folder).
>
> I want to have a special routine that given a date (or a start and end
> date) to return the full path of the *.SAS file(s) or if it doesn't
> exist to print out a statement saying it doesn't exist. If the

Assuming that you can convert your file names to numerical values,
which you seem to have done quite well, then you could do a simple
VALUE_LOCATE() to find match-ups.

For example,

```
;; A list of days from start to finish  
RANGE = DATE + DINDGEN(ENDDATE-DATE+1)  
;; Which files match the file start date?  
;; (NOTE: assumes DIRDATES is ascending order)  
II = VALUE_LOCATE(DIRDATES(0,*), RANGE)  
;; Keep only those that agree with the file end date.  
WH = WHERE(RANGE LT DIRDATES(1,II), CT)  
IF CT EQ 0 THEN MESSAGE, 'No match!!!!'  
II = II(WH)  
  
;; These are the acceptable directories  
II = II(UNIQ(II, SORT(II))) ;; remove duplicates  
FOUND_DIRS = DIRS(II)
```

Good luck,
Craig

--

Craig B. Markwardt, Ph.D. EMAIL: craigmnet@REMOVEcow.physics.wisc.edu
Astrophysics, IDL, Finance, Derivatives | Remove "net" for better response

Subject: Re: Logic problem
Posted by [Ryan](#). on Thu, 17 Jan 2008 17:46:30 GMT
[View Forum Message](#) <> [Reply to Message](#)

Craig Markwardt wrote:

> For example,
> ;; A list of days from start to finish
> RANGE = DATE + DINDGEN(ENDDATE-DATE+1)

```
> ;; Which files match the file start date?
> ;; (NOTE: assumes DIRDATES is ascending order)
> II = VALUE_LOCATE(DIRDATES(0,*), RANGE)
> ;; Keep only those that agree with the file end date.
> WH = WHERE(RANGE LT DIRDATES(1,II), CT)
> IF CT EQ 0 THEN MESSAGE, 'No match!!!!'
> II = II(WH)
>
> ;; These are the acceptable directories
> II = II(UNIQ(II, SORT(II))) ;; remove duplicates
> FOUND_DIRS = DIRS(II)
```

Thanks Ben and Craig for your help. I wasn't aware of the VALUE_LOCATE function in IDL. I've tried these lines to get the results I want but I get "No Match" for every date I enter. I've traced it back to the "II" variable being an array of ones, all the time. Is this signaling that my DIRDATES array is not in ascending order? I currently haven't made any attempts to make it that way.

Ryan.

Subject: Re: Logic problem

Posted by [Ryan](#) on Thu, 17 Jan 2008 19:20:24 GMT

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

```
> Thanks Ben and Craig for your help. I wasn't aware of the VALUE_LOCATE
> function in IDL. I've tried these lines to get the results I want but I
> get "No Match" for every date I enter. I've traced it back to the "II"
> variable being an array of ones, all the time. Is this signaling that
> my DIRDATES array is not in ascending order? I currently haven't made
> any attempts to make it that way.
```

OK, I've figured it out. When I copied and pasted Craig's code into mine, the indices were switched. That is why it kept spitting out ones.

The array it was using only had 2 elements so it was telling me that all the values I use were located after the second element.

Thank you all for your help,
Ryan.
