Subject: Another Late-Night Cautionary Tale: C() format code case sensitivity!
Posted by MarioIncandenza on Mon, 07 Apr 2008 05:12:12 GMT
View Forum Message <> Reply to Message

Hi all,

This is a problem I believe I have now solved. I am posting it for posterity, because of its wicked perniciousness, so that I (or some other poor soul) can search it up later.

I have a routine, well, one of many, which churns through one set of dated files and produces another set of dated files. For my routines, I have become quite religious about handling all time variables internally using double-precision julian dates that come out of JULDAY(). But I handle N types of files each of which has its own unique timestamp convention, so I've built a bestiary of routines to turn each of those into a JULDAY that I can then manipulate easily.

On the input side, that's a lot of STREGEX(), STRMID(), and then JULDAY(), pretty conventional stuff. On the output side, however, there are two very different tools. One is CALDAT, which is just the inverse of JULDAY(). However, why do in two lines (with 2-5 annoying temporary variables to watch after) what you can do in one, with the C() format code. Does anyone else use this?

For the unfamiliar, this works like this:
```
print,julian_date,'' gives you YYYYMMDDHHMM'', $
      format='(c(CYI4.4,CMOI2.2,CDI2.2,CHI2.2,CMI2.2),a)'
```

This is actually in the IDL help somewhere ('format codes, list'), though I always have to dig to find it.

Anyway, after getting some singularly bizarre results out of a routine I had really hoped to see the results of tonight (oh well), I finally found the problem:

> The format for a 24-hour based integer is:
> CHI[+][-][w][.m]
> The format for a 12 hour based integer is:
> ChI[+][-][w][.m]

Did you catch that? It's using the CASE of the 'h' to tell it whether to use 12- or 24- hour time.

So, the upshot of all this, for me, was (you might want to cover your eyes):

```
~/idlpro> find . -name "*pro" -exec sed -i 's/,chi2\.2,/,CHI2.2,/g'
```

'{}' \; -print

I'm happy to report, my routine is working correctly now.

Code safely, everyone,

--Edward H.
