Subject: Re: incremental time data file.....
Posted by Manish on Thu, 23 Aug 2001 15:08:33 GMT
View Forum Message <> Reply to Message

Thanx William, that's great!

Manish

--

"William Thompson" <thompson@orpheus.nascom.nasa.gov> wrote in message
news:9m34cp$4sk$1@skates.gsfc.nasa.gov...
> "Manish" <mrmanish@bigfoot.com> writes:
>
>> Pavel, thanks for the help, just one more thing!!
>
>> I've altered it to zero fill the hour and minute values, but how do I
>> introduce a zero to fill the values between 0 and 9 seconds in the same
way?
>> Essentially, how do you zero fill a floating point value??
>
>> I trust this is an easy thing to fix, but I'd appreciate any help, being
>> only a mere novice....!
>
>> Cheers,
>> Manish
>
> Probably the easiest way is treat everything as integers.
>
> ss = fix(my_time-hh*3600L-mm*60L)
> fsec = round(1E5*(my_time-hh*3600L-mm*60L-ss)) ;Fractional second
> out = transpose([[hh],[mm],[ss],[fsec]])
> print, out[*, 82300:82310], format='(i2.2,":",i2.2,":",i2.2,".",i5.5)'
>
> Also, that way, everything comes out exactly the same string length, e.g.
>
> 23:58:57.64063
> 23:58:58.68750
> 23:58:59.73438
> 23:59:00.78125
> 23:59:01.83594
> 23:59:02.88281
> 23:59:03.92969
> 23:59:04.97656
> 23:59:06.03125
> 23:59:07.07813
> 23:59:08.12500

>
> William Thompson
>
>
>> "Pavel A. Romashkin" <pavel.romashkin@noaa.gov> wrote in message
>> news:3B83E468.4E0D9DE4@noaa.gov...
>>> How about
>>>
>>> my_time = (findgen(24.*60.*60./1.04906)*1.04906)
>>> hh = fix(my_time / 3600L)
>>> mm = fix((my_time - hh*3600L)/60L)
>>> ss = my_time-hh*3600L-mm*60L
>>> out = transpose([[hh],[mm],[ss]])
>>> print, out[*, 82300:82310], format='(i2,":", i2,":", F8.5)'
>>>
>>> If you need exact zero-padded field width, play with string conversion
>>> and formatted output.
>>> Cheers,
>>> Pavel
>>>
>>> Manish wrote:
>>>>
>>>> Hi,
>>>> I've only started using IDL recently, and was wondering if anyone can
>> help
>>>> me out.
>>>> I need to produce a data file which steps through increments of time
>>>> (1.04906 s) for an entire day, i.e. to produce a file which looks
like:
>>>>
>>>> 00:00:01.04906
>>>> 00:00:02.0992
>>>> ...
>>>> ...
>>>> 23:59:59....(whatever the last integer would be!)l
>
>