Subject: Re: Obtaining exponent from a scientific format number
Posted by Dick Jackson on Wed, 17 Apr 2002 16:56:20 GMT
View Forum Message <> Reply to Message

Hi Juan,

"Juan I. Cicuendez" <jicicuendez@gmv.es> wrote...

> I have a scientific formatted number (e.g. 6.8977653e-18) and I have
> to split the exponent and number into two parts like this:
> long:68977653
> exp:-25
> where the first factor has to be a long number and the second the
> exponent.
> The exponential factors can algo change.
>
> The solution I came up is to turn the number into strings and then
> byte(mystring), obtaining the position of '.' and 'e' and then back to
> numbers. This seems to be quite slow and since I have a large number
> of data I don't think is very efficient.

Here's a way that you might like, but I know some others won't, as it uses the mysterious and cryptic "regular expressions" feature. I make a few assumptions:

- you have an array of strings ready to process, for example:
  str = ['6.8977653e-18', '-46.7654e-19']

- all of them have a '.' and an 'e'

If so, these 3 lines of IDL code should work for you:

```
;    Match each string with a pattern like "*.*e*"
;                      "  *   .   *  e   *"
strPieces = StRegEx(str, '(.*)(\.)(.*)(e)(.*)', /Extract, /SubExpr)
;    strPieces now contains StrArr(6, n), columns 1-5 hold the pieces

;    Put columns together to make desired 'long' values
longs = Reform(Long(strPieces[1,*] + strPieces[3,*]))

;    Adjust values in last column to get correct exponents
exps = Reform(Long(strPieces[5,*]) + StrLen(strPieces[3,*]))

IDL> print, longs
   68977653     -467654
IDL> print, exps
       -11        -15
```

There may be methods that execute faster, on my system this takes 0.0003 s per string.

Cheers,
--
-Dick

Dick Jackson / dick@d-jackson.com
D-Jackson Software Consulting / http://www.d-jackson.com
Calgary, Alberta, Canada / +1-403-242-7398 / Fax: 241-7392