Subject: Re: [Q]: ID analog to FORTRAN "sign" function
Posted by Craig Markwardt on Mon, 09 Oct 2000 07:00:00 GMT
View Forum Message <> Reply to Message

Alex Schuster <alex@pet.mpin-koeln.mpg.de> writes:
>> Results:
>> =========
>> The result type is the same as a.
>> The value of the result is
>> | a | if b >= zero
>> and -| a | if b < zero.
>
> No, but you can easily write it:
>
> function sign, a, b
> if (b ge 0) then $
> return, abs(a) $
> else $
> return, -abs(a)
> end

And, after the vector elf is done with it:

```
function sign, a, b
  s = 2*(b GE 0) - 1  ;; +1 or -1
  return, s*abs(a)
end
```

This function works with B as a scalar or a vector. If it's a vector
then it must have the same dimensions as A.

Craig

P.S. Recovering from pneumonia, Ugh.
--
---------- ----------
Craig B. Markwardt, Ph.D. EMAIL: craigmnet@cow.physics.wisc.edu
Astrophysics, IDL, Finance, Derivatives | Remove "net" for better response
---------- ----------

Subject: Re: [Q]: ID analog to FORTRAN "sign" function
Posted by Alex Schuster on Mon, 09 Oct 2000 07:00:00 GMT
View Forum Message <> Reply to Message

Rostyslav Pavlichenko wrote:

> Does the IDl have something close to Fortran SIGN (DSIGN... so on...)

> functions
>
> IN FORTRAN:
> Elemental Intrinsic Function (Generic):
> Returns the absolute value of the first argument times the sign of the
> second argument.
>
> Syntax:
> =======
> result = SIGN (a, b)
> a (Input) Must be of type integer or real.
>
> b Must have the same type and kind parameters as a.
>
> Results:
> =========
> The result type is the same as a.
> The value of the result is
> | a | if b >= zero
> and -| a | if b < zero.

No, but you can easily write it:

```
function sign, a, b
  if ( b ge 0 ) then $
    return, abs( a ) $
  else $
    return, -abs( a )
end
```

It works for both integer and float data types

Alex

--

Alex Schuster Wonko@weird.cologne.de PGP Key available
alex@pet.mpin-koeln.mpg.de