

---

Subject: Re: [Q]: ID analog to FORTRAN "sign" function  
Posted by [Dick Jackson](#) on Fri, 13 Oct 2000 07:00:00 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Phillip David wrote:

> Dick Jackson wrote:

>>

>> Do I dare offer one more? Subscript lookups seem faster than arithmetic  
>> operations, making this one faster, more compact and no less cryptic!

:~)

>>

>> Return, Abs(a) \* ([-1, 1])[b GE 0]

>

> to which I reply:

>

> If this one really works, then why not go even one step further?

>

> Return, ([-Abs(a), Abs(a))][b GE 0]

>

> or

>

> Return, ([a, -a])[a\*b LT 0]

Right, these would work fine for scalar a and b, with negligible time taken in any case. I was looking for the most efficient way when we need this to work on large arrays a and b.

About the 1 vs 1.0 debate, Mark Hadfield wrote:

> I think that's a little \*too\* clever. I just tried multiplying a float  
array

> with 10<sup>7</sup> elements by 1 and then by 1.0. Time taken = 0.52 seconds in both

> cases.

This is getting interesting. For reference, here are a couple of handy timer routines I use:

```
;---
```

```
PRO TStart ; Timer Start  
; Save current time for use by TReport  
COMMON Timer, t0  
t0 = SysTime(1)  
END
```

```
;---
```

```
PRO TReport ; Timer Report  
; Print elapsed time since last TStart
```

```
COMMON Timer, t0
Print, Format='(D10.3," seconds.)',SysTime(1)-t0
END
```

;---

Here's some testing runs from my Win2000 PC:

```
IDL> a=randomu(seed,1000000)-0.5
IDL> b=randomu(seed,1000000)-0.5
IDL> tstart & for i=1,10 do c=Abs(a) * ([-1, 1])[b GE 0] & treport
    3.250 seconds.
IDL> tstart & for i=1,10 do c1=Abs(a) * ([-1.0, 1.0])[b GE 0] & treport
    2.813 seconds.
```

I think the time saving here is not in the multiplying itself, but in the time building an integer array, then converting it to float. In this case it's 10<sup>6</sup> ones/minus-ones, perhaps in your case it was converting only a single 1 to 1.0, then multiplying it.

Fascinating, isn't it? I'd be happy to hear further refinements!

Cheers,

--

-Dick

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

---