Subject: Re: idl2matlab translate-o-matic
Posted by Craig Markwardt on Wed, 23 Feb 2000 08:00:00 GMT
View Forum Message <> Reply to Message

"J.D. Smith" <jdsmith@astro.cornell.edu> writes:
>> Since I can't pass a testing function to that routine (IDL doesn't have
>> higher order functions), I will accept a routine, for illustrative purposes,
>> that removes all even values from the array.
>>
>> Now suppose some joker passes an array containing only even values to that
>> routine...
>>
>> - DM
>>
>
> wh=where(array mod 2, cnt)
> if cnt gt 0 then return,array[wh] else return, -1
>
> I use scalars (often -1) as cheap and easy to use empty arrays. Anything with:
>
>   size(x,/N_DIMEN) eq 0
>
> is patently *not* an array.
>
>
> And as far as the lack of "higher order testing functions":
>
> function evens, arr
>   return, arr mod 2 eq 0
> end
>
> function odds, arr
>   return, arr mod 2
> end
>
> function exclude,arr, exc_func
>   wh=where(call_function(exc_func,arr) eq 0,cnt)
>   if cnt gt 0 then return,arr[wh] else return,-1
> end
>
> and to get rid of the odds, e.g.:
>
> IDL> a=exclude(b,"odds")

Okay, but let's say now you wanted to merge two lists like that
together. Wouldn't this be nice:

IDL> c = [exclude(a,'odds'), exclude(b,'evens')]

The way I say it makes it sound like it's just an inconvenience, which
it is. But for gosh sakes, its a *completeness* issue too. We don't
have a general purpose number system without zero! It would be silly.
Why should we have lists without the empty list? Instead we have to
drag around this extra notion of the COUNT or play tricks by returning
scalars.

Craig

P.S. I don't know why I bother ranting. Of course, RSI probably
won't change it, but I don't want to be a defeatist like David :-)

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