Subject: Re: About keywords and positional parameters
Posted by JD Smith on Fri, 08 Nov 2002 19:45:28 GMT
View Forum Message <> Reply to Message

On Thu, 07 Nov 2002 14:15:18 -0700, David Fanning wrote:

> JD Smith (jdsmith@as.arizona.edu) writes:
>
>> I agree that you can't tell if a parameter is *meant to be* an input
>> parameter, an output parameter, or both without reading the
>> documentation, but you certainly can tell whether it is *capable* of
>> passing output:
>>
>>   if arg_present(parameter) then parameter=myungodlyfunction(foo)
>>
>> ARG_PRESENT() may or many not reveal anything useful, depending on the
>> context. Its primary legitimate use is in avoiding calculations of
>> optional keyword/argument outputs if the caller didn't ask for them.
>
> I have a feeling this piece of helpful advice is more likely to confuse
> Mr. Zhang than enlighten him, but please don't get me started on the
> havoc caused by mis-named functions. Remember, if you want to know if a
> keyword is *used* or not in your program, you have to follow JD's
> helpful advice:
>
>   pro testme, KEY=k
>   if n_elements(k) ne 0 OR arg_present(k) then $
>   print,'You used KEY!' else $
>   print,'You neglected KEY!'
>   end
>

Just to be clear, I've never actually used this method (using
arg_present() essentially as an input test) in real-life code... it's
not really all that helpful. In fact, I wrote it just to get under
your skin, what with your braggadocious claim that you could never
know if a keyword is *used* ;). From my perspective, there are only
three real cases worth keeping track of:

1. An input keyword: Use n_elements() to see if it's present, or
    keyword_set() if it's intended to be a boolean.
2. An output keyword: Use arg_present() if computing the output is
    actually optional in your program, and you want to save the
    trouble of calculating it if the caller didn't ask for it.
3. An input/output keyword: Use a combination of the above two,
    reserving n_elements/keyword_set for input tests, and arg_present
    for output tests.

JD