
Subject: Re: Procedures with variable/dynamic argument list??

Posted by [pgrigis](#) on Fri, 22 Oct 2010 15:50:14 GMT

[View Forum Message](#) <> [Reply to Message](#)

On Oct 22, 11:14 am, Chris Torrence <gorth...@gmail.com> wrote:

> On Oct 22, 8:26 am, Paolo <pgri...@gmail.com> wrote:

>

>

>

>

>

>> It comes out pretty ugly indeed... here my "pritrn" function

>> that I use to work around my bad spelling (!)

>> I guess I didn't know about n_params() when I wrote that.

>

>> Ciao,

>> Paolo

>

>> PRO pritrn,a1,a2,a3,_extra=_extra

>

>> IF exist(a3) THEN print,a1,a2,a3,_extra=_extra ELSE \$

>> IF exist(a2) THEN print,a1,a2,_extra=_extra ELSE \$

>> IF exist(a1) THEN print,a1,_extra=_extra

>

>> END

>

>> ;from solarsoft

>> function exist,var

>

>> return,n_elements(var) ne 0

>

>> end

>

> Hi all,

>

> You might look at SCOPE_VARFETCH. You could dynamically construct the

> argument name, and then pass the string to SCOPE_VARFETCH to retrieve

> the value. Something like:

>

> PRO pritrn,a1,a2,a3,_extra=_extra

> for i=1,n_params() do print, SCOPE_VARFETCH('a'+strtrim(i,

> 2)),_extra=_extra

> end

>

> This only helps in cases where you can break up the arguments and do a

> call with each one separately.

>

> Paolo, if I could suggest a tweak, you could use a CASE statement

```
> instead:
>
> case n_params() of
> 1: print,a1,_extra=_extra
> 2: print,a1,a2,_extra=_extra
> 3: print,a1,a2,a3,_extra=_extra
> else: message,'Incorrect number of arguments'
> endcase
>
```

Yes, the CASE version is nicer and that's how I would have wrote it if I had known about n_params() back then... :)
You always learn something new :)

Thanks,
Paolo

```
> This avoids all of those nasty if/else blocks.
>
> -Chris
> ITTVIS
```
