Here's a simple example of a routine that returns different
values if it is called by value or by reference.

```
pro pg,a,b

b=arg_present(a)

end
```

```
IDL> x=1
IDL> pg,x,b
IDL> print,b
      1

IDL> pg,x+0,b
IDL> print,b
      0

IDL> pg,x[0],b
IDL> print,b
      0
```

Ciao,
Paolo

David Fanning wrote:
> yp writes:
>
>> You can download the code from
www.research.plymouth.ac.uk/casix/temp/test_morel_LUT.pro
>>
>> You will also need the LUT (morel_fq.dat) to run the code.
>> www.research.plymouth.ac.uk/casix/temp/MOREL_FQ.DAT
>
> Well, I haven't tracked the actual culprit down yet, but
> the calculations in INT_LUT are doing *something* to the
> variables that are being passed in by reference. If I
> pass everything in by value, then there is no difference:
>
>
> IDL>   int_LUT,412.5D, 1D,   0.0D, 0.0D, 0.0D, 0.03D, result3

```
> IDL>    int_LUT,412.5D+0, 1D+0, nul+0, nul+0, nul+0, 0.03D+0, result4
> IDL>    print,'RESULT 3: ', result3[*]
> IDL>    print,'RESULT 4: ', result4[*]
>
> RESULT 3:    0.090100000    0.00000000    0.00000000
> 0.00000000    0.00000000    0.00000000    0.00000000
>
> RESULT 4:    0.090100000    0.00000000    0.00000000
> 0.00000000    0.00000000    0.00000000    0.00000000
>
> I suspect there is a calculation in there that sets one
> of the values very nearly to what you expect it to be,
> but not quite.
>
> Still weird, but more believable. :-)
>
> Cheers,
>
> David
```