Subject: Re: keywords and passing values
Posted by Conor on Mon, 16 Jul 2007 18:00:27 GMT
View Forum Message <> Reply to Message

On Jul 16, 12:49 pm, Ingo von Borstel <newsgro...@planetmaker.de>
wrote:

```
>  Hi there,
>
>  I probably have - again - an issue with parameters and calling by
>  reference and value. But I don't see where I miss something as the same
>  works in other places:
>
>  I have a set of routines (actually object routines) that take a
>  parameter "valid". What I do is along these lines (added line numbers
>  for reference):
>
>  1       myobj::setdata, var
>  2       valid = 1
>  3       objnum = self->getdata_index(var,valid=valid,_extra=extra)
>  4       IF NOT valid THEN PRINT, "creating new data!"
>  5       END
>
>  7       FUNCTION myobj::getdata_index, var, valid=valid, $ (some other keywords)
>  8       valid = 1
>  9       (check for several things and return the search result, if successful)
>  10      valid = 0
>  11      RETURN, -1
>  12      END
>
>  If I now call the print routine with a variable value that will fail,
>  print itself never gets reported that its subroutine, getdata_index
>  returned valid=0, though I checked in line 11 that valid=0
>
>  IDL> myobj->print, 'not existent'
>  IDL>
>
>  Funnily, if I trace the programme and stop in line 4, call
>  IDL> valid = 1 & print, getdata_index(var,valid=valid) & print, valid
>  -1      0
>  IDL>
>  I get the results I want (valid=0).
>
>  I'm a bit puzzled right now and don't see where something is missing.
>
>  Best regards & thanks for any advice,
>  Ingo
>  --
>  Ingo von Borstel                <newsgro...@planetmaker.de>
```

> Public Key:http://www.planetmaker.de/ingo.asc
>
> If you need an urgent reply, replace newsgroups by vgap.

Well, this works:

```
pro test,valid=valid
valid = 1
end

test,valid=valid
print,valid
; prints 1
```

I don't see why your example wouldn't work either. I can only assume, therefore, that you have a bug somewhere, possibly in your testing? I really don't know though, sorry...