Subject: Re: Functions and arrays
Posted by gunter on Wed, 04 Dec 1996 08:00:00 GMT
View Forum Message <> Reply to Message

William Thompson (thompson@orpheus.nascom.nasa.gov) wrote:
[see previous post for details]

I find this behaviour to be irresponsible on the part of IDL. To really stress the point, do the following. (There is no new info just a better illustration of the problem for those new to IDL)

Insert a STOP command in the function test1, after assigning the values to the test2 array:

```
:  function test1, i
:  test2 = [1,3,7]
 STOP
:  a = test2(i)
:  return, a
:  end
```

Then compile test2 first, then test1 and run test1 via
  print, test1(0)

When you get back the command prompt, type HELP and you'll see that there is an array named test2 and lower you will see that there is a function named test2.

```
 (idl)% print, test1(0)
 % Stop encountered:  TEST1              3 /TEST1.pro
 (idl)% help
 % At  TEST1              3 /TEST1.pro
 %     $MAIN$
 Code area used: 100% (200/200),  Symbol area used: 21% (40/184)
 # local variables: 3,  # parameters: 1
 A             UNDEFINED = <Undefined>
 TEST2           INT       = Array(3)
 I             INT       =       0
 Compiled Procedures:
    $MAIN$  LOADCT
 Compiled Functions:
    TEST1    TEST2
```

Now, type
  print, test2
and you'll see the values in the array. If you then type
  print, test2(0)
You will get the value of the function test2 (-1 in this case).

Again, there's nothing new that previous posts haven't told us, but I find the behaviour rather silly.
--
david gunter
http://www.mcs.anl.gov/people/gunter/

"When you are a Bear of Very Little Brain, and you Think of Things, you find sometimes that a Thing which seemed very Thingish inside you is quite different when it gets out into the open and has other people looking at it."
- A.A. Milne, "The House At Pooh Corner"