Subject: Re: more bugs in envi !!!
Posted by gauravjn123 on Wed, 16 Jul 2003 07:33:17 GMT
View Forum Message <> Reply to Message

Hi!
Okay...i shall give u the outputs also. Here are the functions:

```
1.;Stored in file trash.pro
  function ex,b1
  print,' This is the function ex,b1'
  print,size(b1)
  help,b1
  end
2.;Stored in file trash2.pro
  function ex2,b1,b2
  print,'This is the function ex2,b1,b2'
  print,size(b1)
  help,b1
  end
```

The outputs of these functions are :

```
ENVI>
% Compiled module: TRASH.

ENVI>
% Compiled module: TRASH2.

ENVI>
 This is the function ex,b1
           2           5           5           4
   25
B1              FLOAT     = Array[5, 5]
 This is the function ex,b1
           2           5           5           4
   25
B1              FLOAT     = Array[5, 5]
 This is the function ex,b1
           2           5           5           2
   25
B1              INT       = Array[5, 5]
 This is the function ex,b1
           2         568         653           2
370904
B1              INT       = Array[568, 653]

ENVI>
This is the function ex2,b1,b2
           2           5           5           4
```

```
  25
B1              FLOAT     = Array[5, 5]
This is the function ex2,b1,b2
         2          5          5          4
  25
B1              FLOAT     = Array[5, 5]
This is the function ex2,b1,b2
         2          5          5          2
  25
B1              INT       = Array[5, 5]
This is the function ex2,b1,b2
         2        568        326          2
185168
B1              INT       = Array[568, 326]
This is the function ex2,b1,b2
         2        568        327          2
185736
B1              INT       = Array[568, 327]

ENVI>
```

Hope this helps you in finding out the answer to my question.

Gaurav Jain

Marc Schellens <m_schellens@hotmail.com> wrote in message news:<3F12DAB7.607@hotmail.com>...
> Would be helpful if you post WHAT they output also.
> And then of course the context from which you call them...
> (Two functions with the same name cannot live in one (IDL-) program)
>
> Gaurav wrote:
>> Hi all!
>> Thanx for the help...well u both were right...i was under the wrong
>> impression that i had defined it as a long integer...but i hadn't. But
>> i still cant find out why these 2 functions are producing different
>> outputs :
>> Please go through the 2 functions:
>>
>>>> 1.function example,b1
>>>> help,b1
>>>> end
>>>> 2.function example,b1,b2
>>>> help,b1
>>>> end
>>>>

>>>> Now could anyone tell me why these 2 codes are producing different
>>>> final answers??? I just cant find any reason for it.
>>>
>>
>> Gaurav Jain
>> ENST-Bretagne
>>
>>
>> Marc Schellens <m_schellens@hotmail.com> wrote in message
news:<3F115ED4.6040200@hotmail.com>...
>>
>>>> Hi folks!
>>>> Sorry to trouble u again....but envi seems to be just going above my
>>>> head...there is another problem that i am facing:
>>>> Please go through the 2 functions:
>>>>
>>>> 1.function example,b1
>>>> help,b1
>>>> end
>>>> 2.function example,b1,b2
>>>> help,b1
>>>> end
>>>>
>>>> Now could anyone tell me why these 2 codes are producing different
>>>> final answers??? I just cant find any reason for it.
>>>>
>>>> One more problem :
>>>>
>>>> 'b1' is a positive array and 'avgb1' is a positive number but when i
>>>> use the formula : result=b1*100/avgb1
>>>>   print, result
>>>> , the answer that i get is a negative array.
>>>>
>>>>   Whereas if i use the formula : result=b1/avgb1
>>>>   print, result * 100
>>>> , then i get a positive array
>>>>
>>>
>>> Even without knowing ENVI, the second problem isn't a bug in ENVI.
>>>
>>> A integer occupies (in IDL) 16-bit, and therfore covers a range form
>>> -32768 to
>>> 32767
>>>
>>> anything larger (or smaller) cannot be represented and therefore you
>>> observe an 'overflow', resulting in a negative number.
>>> IDL evalutates your expressinon from left to right, the
>>> overflow occurs after the multiplication with 100.

>>> In the second case the division is done first and thus the value seems
>>> to be small enough not to overflow in the later multiplication.
>>>
>>> The solution is to convert your b1 array to LONG.
>>> But did you notice that you do an integer division?
>>> 1/2 equals 0.
>>> Maybe you want to use FLOAT or DOUBLE.
>>>
>>> hdh,
>>> marc
>>
