
Subject: about replicate_inplace

Posted by [xje4e](#) on Tue, 22 Jul 2003 20:37:15 GMT

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

Hi, there,

I am really puzzled by the IDL usage of memory. Here is an example:

```
T = systime(1)
b = bytarr(nreg, nreg)
print, systime(1) - T, ' seconds.'
```

```
T = systime(1)
replicate_inplace, b, 0
print, systime(1) - T, ' seconds.'
```

In the example, the first part is to create an array with values 0.
The second part is to assign each element in the array the value 0.
The running time for the first part is 0.12sec, while the running time
for the second part is 0.37sec.

My question is: since the first part needs to allocate the memory and
assign values to elements, it should take longer time than the second
part. In IDL help, it says: "REPLICATE_INPLACE can be faster and use
less memory than the IDL function REPLICATE or the IDL array notation
for large arrays that already exist. " However, why the first part
takes less time?

If this is really the case in IDL, then it's better to allocate a new
array than to use the current array. It's really strange for me.

Any suggestion?

Xiaoying
