Subject: Re: Array concatenation
Posted by Michael Galloy on Wed, 07 Jun 2006 15:06:45 GMT
View Forum Message <> Reply to Message

maye wrote:
> Hi folks,
> I was searching for an effective way to do this, but can't find anybody
> writing about what (as usual) seems to be an obvious problem/task to
> me. :)
> I don't know how many elements I will collect while scanning a bunch of
> images, so I want to use array concatenation to collect the values like
> this:
> means = [means, currMean]
> But for this to compile/run properly, I need mean to exist before.
> If I do a
> means = 0.
> before the loop, I will always have a zero-element in the beginning
> that I don't want at plotting time. Of course I could remove it with
> means = means[1:*]
> but not only does this waste resources, it also becomes cumbersome if I
> collect 20 different data values, for that I ALL have to remove the
> first value?
> Surely there must be a better way?
> Please help me to program IDL efficiently! :)
> Best regards,
> Michael
>

I have a class MGArrayList that I use for such things. It follows the
same interface as IDL_Container, but allows for any IDL data type (one
data type per object). So for example,

```
IDL> olist = obj_new('mgarraylist', example=0.0, blocksize=10)
IDL> olist->add, 3.0
IDL> olist->add, 5.0
IDL> olist->add, 10.0
IDL> print, olist->get(/all)
      3.00000      5.00000      10.0000
```

There's more info along with links to download and API documentation at:

http://michaelgalloy.com/?p=11

Mike
--
www.michaelgalloy.com