Subject: Re: On Pointers and Culture
Posted by Antonio Santiago on Fri, 04 Mar 2005 08:16:10 GMT
View Forum Message <> Reply to Message

Michael Wallace wrote:
> I've finally reached an impasse in my IDL programming. Arrays just
> aren't doing it for me anymore. As of a few minutes ago I starting
> teaching myself IDL pointers. Does IDL already have implementations of
> the common data structures such as stacks, queues and trees or will I
> need to compose my own? I figured it was easier to ask the group and go
> to bed (it's 1:30AM local time) and see responses in the morning than
> trying to slog through IDL documentation at such a late hour. Although
> the documentation would probably make better sense now...
>
> -Mike

You can found basic data structures in RSI user contributed library
(http://www.rsinc.com/codebank/search.asp?search=category&amp ;product=IDL&catid=16)
. See the hashtable and vector.

Also D.Fanning has a linkedlist implementation
(http://www.dfanning.com/programs/linkedlist__define.pro)

and Craig Markwardt has an implemententation of hash
(http://cow.physics.wisc.edu/~craigm/idl/down/hashtable__defi ne.pro).

With all this you have: hashtables, lists and staks.

If you are new on IDL pointers remember are not the same as C pointers.
Now you have your "conventional" memory (managed by IDL) and the HEAP
memory (managed by yourself).

```
a = INDGEN(100)
p = PTR_NEW(a)
```

This lines creates an 'a' vector and 'p' pointer variable in the
conventional memory, also copies the same 'a' vector into HEAP memory a
makes 'p' points this.
Maybe, it is better:

```
p = PTR_NEW( INDGEN(100) )
```

Now, you only have a 'p' pointer variable in conventional memory and an
array in the HEAP memory.

Bye.
Antonio

(Sorry about mu poor enligh :)