
Subject: Re: pointer question

Posted by [Paul van Delst](#) on Thu, 22 Mar 2001 14:52:28 GMT

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

Ted Graves wrote:

>
> Hi all,
>
> Another lurker question ... let's say you define a pointer using the PTR_NEW
> function and assign to a variable x. As long as you keep track of x and don't
> reassign x and lose the pointer to the heap variable, things are great. You
> can remove the heap variable from memory using the PTR_FREE procedure.
>
> But now let's say i have a function TEST that takes a pointer as an argument,
> and i want to create a pointer on the fly to use in TEST. So i do something
> like
>
> result = TEST(PTR_NEW(value))
>
> where value is whatever i want the heap variable to be. What happens to the
> heap variable assigned in this statement after TEST returns? I'm assuming
> from that because of the way it was created, a heap variable now exists that i
> can't easily get rid of without using HEAP_GC.
>
> Me and my sloppy programming ...

If you recognise this as sloppy programming along with all it's problems (heap variable you can't get rid of easily), then why do this? Seems to me to be a good example of how *not* to use pointers. What's wrong with:

```
x=ptr_new(value)
result = test(x)
```

?

what if value is some huge array? won't your original call suck up a bunch of memory that you can't free without the use of HEAP_GC?

From the HEAP_GC online help:

"Note - Garbage collection is an expensive operation. When possible, applications should be written to avoid losing pointer and object references and avoid the need for garbage collection."

Note the last sentence.

paulv

--

Paul van Delst A little learning is a dangerous thing;
CIMSS @ NOAA/NCEP Drink deep, or taste not the Pierian spring;
Ph: (301)763-8000 x7274 There shallow draughts intoxicate the brain,
Fax:(301)763-8545 And drinking largely sobers us again.
paul.vandelst@noaa.gov Alexander Pope.
