
Subject: Re: accessing heap variable elements
Posted by [Ken Mankoff](#) on Sat, 12 May 2001 17:42:56 GMT
[View Forum Message](#) <> [Reply to Message](#)

On Sat, 12 May 2001, Ken Mankoff wrote:

```
> On Sat, 12 May 2001, Jared Crossley wrote:
>
>> Can any one explain this to me?
>> -----
>> IDL> a=ptr_new( lonarr(2) )
>> IDL> help, a
>> A          POINTER  = <PtrHeapVar2229>
>> IDL> help, *a
>> <PtrHeapVar2229>
>>          LONG      = Array[2]
>> IDL> help, *a[0]
>> <PtrHeapVar2229>
>>          LONG      = Array[2]
>> IDL> help, *a[1]
>> % Attempt to subscript A with <INT      (      1)> is out of range.
>> -----
>> I would expect that *a[0] and *a[1] would be long integers.  How do I
>> access the elements in the long integer array heap variable?
>>
>> Thanks, Jared
>
> Hi Jared,
>
> last line should be (*a)[0]
>
> -k.
>
```

sorry, i realized i should include an explanation, as you requested.

its all about operator precedence. Now, "[" is *not* mentioned in the IDL 5.3 Online Help, but they have this:

Priority Operator

```
-----
First (highest) ( ) (parentheses, to group expressions)
Second  * (pointer dereference)
        ^ (exponentiation)
Third   * (multiplication)
        # and ## (matrix multiplication)
etc.
```

I think its safe to assume that [] is parsed as ().

so:

(*a[0] EQ *(a[0]))

which means "take subscript of 'a', then dereference it."

But 'a' is a pointer to an array, not an array of pointers. So you can't subscript it, and even if you could, i'm not sure what you'd be dereferenceing with your *.

You want the opposite order, or:

(*a)[0]

which means "dereference 'a', then take subscript."

-k.

--

Ken Mankoff

LASP://303.492.3264

<http://lasp.colorado.edu/~mankoff/>
