
Subject: Re: Arrays of Structures

Posted by [Mick Brooks](#) on Thu, 08 Feb 2007 18:04:46 GMT

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

On Feb 8, 5:41 pm, "Brian Larsen" <balar...@gmail.com> wrote:

```
> This is what you did and it is true this is an array with 50 elements
> IDL> HELP, structs.a
> <Expression>  INT      = Array[50]
>
> but if you look at structs by itself not structs.a, it is the array,
> NOT structs.a
>
> IDL> HELP, structs
> STRUCTS      STRUCT    = -> <Anonymous> Array[50]
>
> So what is inside the structure?
>
> Here a is inside the struct and it is an int
>
> IDL> HELP, structs, /str
> ** Structure <84bd40c>, 1 tags, length=2, data length=2, refs=1:
>  A          INT          1
```

(I think) I'm following you to here...

```
> Meaning that structs.a[23] doesn't make sense because structs.a is an
> int.
```

... but this is where you lose me. Doesn't the the first line that you showed say that structs.a is an array of 50 ints?

Maybe I just can't read the output of HELP properly, but structs.a certainly sometimes behaves as an array. Try PRINTing it, for instance.

Or let's compare the output of HELP on a real array of 50 ints:

```
IDL> HELP, structs.a
<Expression>  INT      = Array[50]
IDL> HELP, indgen(50)
<Expression>  INT      = Array[50]
```

```
> While structs[23].a does make sense because structs is an array.
```

Yes, I'm happy with why this is right, but still not clear why structs.a[23] is wrong.

```
> This is just one of those lessons that take a while to get a hold of.
>
```

> Make any sense?

Not really, but thanks for trying.

Cheers,

Mick
