Subject: Re: help: Type & size checking
Posted by davidf on Thu, 25 Mar 1999 08:00:00 GMT
View Forum Message <> Reply to Message

Phil Aldis (philaldis@geocities.com) writes:

> P.S. I'm not on any form of commission or anything, this is a genuine
> recommendation of a great book.

Well, shucks. But I'll send you a check anyway (or at least
treat you to dinner) if he orders a copy. :-)

Cheers,

David

P.S. I'm thrilled to have people purchase the book through
Amazon.com (or from any source), but the folks who read the
newsgroup know you can get it faster and cheaper by ordering
it directly and mentioning that you read this newsgroup. :-)

--
David Fanning, Ph.D.
Fanning Software Consulting
Phone: 970-221-0438 E-Mail: davidf@dfanning.com
Coyote's Guide to IDL Programming: http://www.dfanning.com/
Toll-Free IDL Book Orders: 1-888-461-0155

Subject: Re: help: Type & size checking
Posted by Phil Aldis on Thu, 25 Mar 1999 08:00:00 GMT
View Forum Message <> Reply to Message

From the questions you have been asking (and I think you may even have said it
yourself), you are just beginning to learn how to program in IDL.

I don't about you but when I started I found that the IDL help files were all
very well if you wanted to find out the particular syntax for a function, but
they didn't really tell you how to program good code in IDL. They are great
reference tools (I don't know how I would be able to get by without them now)
but when you start you want a tutorial from a very experienced IDL programmer
which guides you through wreiting tight IDL code.

Well as luck would have it, I soon went on an RSI course with David Fanning
and his book as given out as part of the course. It is an excellent book
which willl take you through all sorts of aspects of IDL. It is clear and
simple to understand and gave me a great founding in this language.

You can find out more details by visiting David's website www.dfanning.com.
Or for a more impartial view you can visit Amazon books and order through
there, I believe there are some reviews from people who've bought it.

I woudl thouroughly recommend that you get it as it would help teach you some
great IDL technique.

Cheers,
Phil

P.S. I'm not on any form of commission or anything, this is a genuine
recommendation of a great book.

Subject: Re: help: Type & size checking
Posted by Phil Aldis on Thu, 25 Mar 1999 08:00:00 GMT
View Forum Message <> Reply to Message

In article <7dd618$j5i@hermes.fundp.ac.be>,
"Tri VU KHAC" <tvk@info.fundp.ac.be> wrote:
> Hi folks,
> Given a structured variable, how can I verify its type and its size as
> in C/C++ (sizeof, typeof) ?
> Thanks in advance.
> Best regards.
> Tri.
>
>

I presume a structured variable means just a structure, in which case it's
quite simple. I'm not entirely sure what sizeof returns, but if you're trying
to find out the number of tags in the structure then the function is N_Tags

```
IDL> mystruct = {string:'', integer:0,float:0.}
IDL> print, N_Tags(mystruct)
           3
```

If however you want to find tou the actual byte size of it then specify the
keyword LENGTH like this:

```
IDL> print, N_Tags(mystruct, /LENGTH)
          16
```

Remember /LENGTH means LENGTH=1, which sets the keyword.

However bear in mind this excerpt from the IDL help files:

Note The length of a structure is machine dependent. The length of a given structure will vary depending upon the host machine. IDL pads and aligns structures in a manner consistent with the host machine's C compiler.

The way to find out the type of a variable is to use the Size function with the type keyword set. The different variables are numbered like this (check out the help file on Size)

0 Undefined
1 Byte
2 Integer
3 Longword integer
4 Floating point
5 Double-precision floating
6 Complex floating
7 String
8 Structure
9 Double-precision complex
10 Pointer
11 Object reference
12 Unsigned Integer
13 Unsigned Longword Integer
14 64-bit Integer
15 Unsigned 64-bit Integer

However, bear in mind that 12-15 are specific to IDL5.2.

```
IDL> print, Size(mystruct, /TYPE)
           8
IDL> print, Size(mystruct.integer, /TYPE)
           2
```

.....so you can find out that mystruct is a structure and and that the integer tag is an integer.

Anyway I hope this all helps.

Cheers,
Phil