
Subject: Re: Recognizing double precision?

Posted by edward.s.meinel@aero on Wed, 10 Oct 2007 18:21:52 GMT

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

On Oct 10, 11:53 am, "R.G. Stockwell" <noem...@please.com> wrote:

> <mei...@aero.org> wrote in message

>

> news:1192025265.336346.261400@o3g2000hsb.googlegroups.com...

>

>

>

>> On Oct 9, 11:50 am, "R.G. Stockwell" <noem...@please.com> wrote:

>>> "wlandsman" <wlands...@gmail.com> wrote in message

>

>>> news:1191597160.614557.153160@50g2000hsm.googlegroups.com...

>

>>>> About once a year I receive a complaint about my code because someone

>>>> inputs a Julian date like this

>

>>>> IDL> jd = 2441636.1

>

>>> btw, this may be obvious to all, but you can force the input

>>> with a read command, read it as string, and cast it to double.

>

>>> IDL> .GO

>

>>> : 2441636.1

>

>>> S STRING = ' 2441636.1'

>

>>> 2441636.1

>

>>> single:

>

>>> 2441636.00000

>

>>> double

>

>>> 2441636.10000

>

>>> Cheers,

>>> bob

>

>> Sure, you could do that, but that is even worse than

>

>> IDL>jd = 2441636.1d

>

> Not worse, just different. The problem is that we are talking about

> different things. One is a user interface, in which the developer should
> be responsible for inputting the correct variable type. (thus, read string,
> and then do error checking, range checking, valid input checking and casting
> to the appropriate type). Because one thing is certain, if a user _can_
> do something that will crash the code, then the user _will_ do something
> to crash the code (I've even run into malicious users who try to crash
> the code so they can stop working.)
>
> The second, above, is input at the IDL command line. That is, in my
> opinion, identical to a line of code in a program. If the developer types in
> `jd = 2441636.1`
> in their routine when they should have made it a double, then that is a
> programming error.
> It is entirely the programmers responsibility to have the correct type for
> their variables.
>
> I don't think the compiler should take the defined floating point variable
> and force it to double unless the programmer tells it to (either explicitly
> or implicitly). Just my opinion anyway.
>
> Cheers,
> bob

Oh, I absolutely agree with the GUI issue. I do the same thing you do.

The thing is that Wayne was complaining about command line inconsistency, not a problem with his routine. IDL correctly assigns integer types, but not floating point types. Once you type something on the command line and it gets assigned a variable type, if it is the wrong one, you're hosed. Example:

```
IDL>jd = 2441636.1
IDL>result = WL_ROUTINE(jd)
```

IDL just changed the value of `jd` without notifying the user. Do you consider that good programming practice? The programming error is on IDL's part, not Wayne.

Ed
