

---

Subject: Conversion Error?

Posted by [Laurent Chardon](#) on Mon, 16 Aug 1999 07:00:00 GMT

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

---

I can't figure this out. Is it a bug or something that I can not understand?

This is with IDL 5.2 running on a PII 333 on Windows NT.

If I do

```
IDL> size=double(12990)
IDL> help,size
SIZE      DOUBLE   =    12990.000
IDL> c=fltarr(size)
IDL> help,c
```

I get

```
C      FLOAT   = Array[12990]
```

which is what I expect.

But... If I do

```
IDL> size=(double(200-70.1))/0.01
IDL> help,size
SIZE      DOUBLE   =    12990.000
IDL> c=fltarr(size)
IDL> help,c
```

I get

```
C      FLOAT   = Array[12989]
```

Does anyone know why??? I am starting to not trust any of my previous calculations done with IDL. Or is this a "feature" of the Pentium II?

Any help would be appreciated...

Laurent Chardon  
Trent University

---

---

Subject: Re: Conversion Error?

Posted by [Med Bennett](#) on Mon, 16 Aug 1999 07:00:00 GMT

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

---

Laurent,

"Size" is also an IDL function. I think that you would do well to avoid using variable names that are the same as IDL functions - this has caused me problems in the past.

Laurent Chardon wrote:

```
> I can't figure this out. Is it a bug or something that I can not understand?
> This is with IDL 5.2 running on a PII 333 on Windows NT.
>
> If I do
>
> IDL> size=double(12990)
> IDL> help,size
> SIZE      DOUBLE   =    12990.000
> IDL> c=fltarr(size)
> IDL> help,c
>
> I get
>
> C      FLOAT   = Array[12990]
>
> which is what I expect.
>
> But... If I do
>
> IDL> size=(double(200-70.1))/0.01
> IDL> help,size
> SIZE      DOUBLE   =    12990.000
> IDL> c=fltarr(size)
> IDL> help,c
>
> I get
>
> C      FLOAT   = Array[12989]
>
> Does anyone know why??? I am starting to not trust any of my previous
> calculations done with IDL. Or is this a "feature" of the Pentium II?
>
> Any help would be appreciated...
>
> Laurent Chardon
> Trent University
```

---

Subject: Re: Conversion Error?

Posted by [davidf](#) on Mon, 16 Aug 1999 07:00:00 GMT

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

---

Laurent Chardon (chardons@NOT\_THISsprint.ca) writes:

> Thanks David and Martin for your help. So if I understand properly, IDL uses  
> fix() by default when converting doubles to integer. That's tricky. Does it  
> make sense to do that?

Well, probably LONG by default. I just happened to choose FIX. But definitely *\*not\** ROUND.

Does it make sense? Yes, I think so. It is consistent. You know what to expect. If *\*you\** want to round, there is a routine to do that. It requires a bit of knowledge about how numbers work on computers. Computer scientists learn that, of course, but the rest of us learn it when we need it (i.e., after struggling for hours trying to figure out why our results are not what we expect.)

Cheers,

David

--

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: Conversion Error?

Posted by [Laurent Chardon](#) on Mon, 16 Aug 1999 07:00:00 GMT

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

Thanks David and Martin for your help. So if I understand properly, IDL uses fix() by default when converting doubles to integer. That's tricky. Does it make sense to do that?

The solution would be to use c=fltarr(round(size)). Shouldn't IDL do that by default?

I'm wondering what's the use of using fix() instead of round() in this instance. Does anybody know?

Laurent Chardon

Trent University

---

Subject: Re: Conversion Error?

Posted by [m218003](#) on Mon, 16 Aug 1999 07:00:00 GMT

```
>> IDL> size=(double(200-70.1))/0.01
>> IDL> help,size
>> SIZE      DOUBLE   =    12990.000
>> IDL> c=fltarr(size)
>> IDL> help,c
>>
>> I get
>>
>> C          FLOAT    = Array[12989]
>>
>> Does anyone know why???
```

That might be in the FAQ by now? Anyway: try  
print,(double(200-70.1))/0.01,format='(f24.14)'  
and you'll see what you really get.

Solution: add a tiny little bit to your double number before truncating it by using it as an index (LONG please, not INTEGER !). To determine how much to add, you should compute the ALOG10 and subtract about 9 or 10 from it. In your case that would give you something like -5 which means you would add  $10^{-5} = 0.0001$  .. hardly noticeable but efficient in getting correct numbers. **\*\*BUT\*\***: as any speed fanatics will tell you: logs and likes are very costly in computer resources, so whenever you know the numerical range of values use a constant rather than the computation above.

Regards,  
Martin.

---

---

Subject: Re: Conversion Error?

Posted by [m218003](#) on Tue, 17 Aug 1999 07:00:00 GMT

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

---

In article <4KVt3.65216\$jl.41071979@newscontent-01.sprint.ca>,  
"Laurent Chardon" <chardons@NOT\_THISsprint.ca> writes:  
> Thanks David and Martin for your help. So if I understand properly, IDL uses  
> fix() by default when converting doubles to integer. That's tricky. Does it  
> make sense to do that?  
> The solution would be to use c=fltarr(round(size)). Shouldn't IDL do that by  
> default?

Definitively NO! Most often you need truncation rather than rounding, e.g. when you want to create an array with a certain number of elements. Of course, you are right that in 99.9999999% of these circumstances, 99.9999999 should be interpreted as 100 and not 99, but that's how bits work together ;-C Unfortunately, we are used to thinking decadal, whereas

practically every machine thinks dual (or for the convenience of the programmer hexadecimal). You can open any book on digital numerics and you will see how such roundoff errors are introduced. Unfortunately, you will not always see lower values, else it would be easy enough to write

```
a = fltarr( long(expression) + 1 )
```

Martin

> I'm wondering what's the use of using `fix()` instead of `round()` in this  
> instance. Does anybody know?

- >
- > Laurent Chardon
- > Trent University

```
--
[[
Martin Schultz Max-Planck-Institut fuer Meteorologie [[
Bundesstr. 55, 20146 Hamburg [[
phone: +49 40 41173-308 [[
fax: +49 40 441787 [[
martin.schultz@dkrz.de [[
[[
```

Subject: Re: Conversion Error?

Posted by [steinhh](#) on Tue, 17 Aug 1999 07:00:00 GMT

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

In article <4KVt3.65216\$jl.41071979@newscontent-01.sprint.ca>  
"Laurent Chardon" <chardons@NOT\_THISsprint.ca> writes:

- > Thanks David and Martin for your help. So if I understand
- > properly, IDL uses fix() by default when converting doubles to
- > integer. That's tricky. Does it make sense to do that?

Yes, it does. I know of no other language that rounds by default when converting float to integer types. (Ok, so there probably are *some* languages that use round).

> The solution would be to use `c=fltrrr(round(size))`. Shouldn't IDL  
> do that by default?  
> I'm wondering what's the use of using `fix()` instead of `round()` in  
> this instance. Does anybody know?

You can bet a number of applications *\*rely\** on this default behaviour. Needs differ from one situation to another. You should

also consider using either `ceil()` or `floor()` instead of `round()`,  
in some cases.

Regards,

Stein Vidar

---