

Re: Type of variables and their effect on result...

Source:

<http://www.tech-archive.net/Archive/Excel/microsoft.public.excel.programming/2007-01/msg00045.html>

- *From:* "MichDenis" <michdenis@xxxxxxxxxxx>
 - *Date:* Sun, 31 Dec 2006 09:29:26 -0500
-

Thanks to both of you, Ron et Jerry for all these explanations.

"Ron Rosenfeld" <ronrosenfeld@xxxxxxxxxxx> a écrit dans le message de news:
620ep2he22pluc75if05ukj5ndp3s9det6@xxxxxxxxxxx
On Sat, 30 Dec 2006 12:41:01 -0800, Jerry W. Lewis <post_a_reply@xxxxxxxxxxxxxxxx>
wrote:

"Ron Rosenfeld" wrote:

...

However, when you define A as a variant type, I believe the precision increases, so you wind up with the "correct" answer.

Actually, the opposite is true, 4-function arithmetic that involves only constants and explicitly declared doubles is done in extended precision (10 bytes), where it is only done in double precision (8 bytes) with variants. It is an accident of this particular calculation that the lower precision calculation resulting from using variants was more in keeping with the OP's intent.

It has to do with the inherent inaccuracies in the IEEE standard for double precision floating point numbers.

The number 37.7266 cannot be expressed accurately as a binary number. It's actually the equivalent of something like 37.726599..... so multiplied by 10000 will be 377265.99... and the INT function will return 377265.

Exactly right. The IEEE double precision binary representation for 37.7266 has the decimal value of
37.72659999999999769215719425119459629058837890625

Re: Type of variables and their effect on result...

When you multiply by 10000, you get
377265.9999999999769215719425119459629058837890625
in extended precision, which is approximated by 377266 in double precision.

Thus

Dim A As Variant, B As Variant

A = 37.7266

B = 10000

X = Int(A * B)

returns 377266 because the Int() function received a double precision value,
whereas the other versions returned 3772655 because the Int() function
received an extended precision value.

There are various ways to insure that Int() receives a double precision
value, including

X = Int(CDbl(A * B))

and

X = A * B

X = Int(X)

but the form least likely to give binary surprises is

X = Int(CDbl(CStr(A * B)))

since it works with no more figures than Excel will display.

Jerry

Thanks for that more accurate and precise <g> and coherent description.

—ron

.