04-01-2022, 09:31 PM **Post: #1**

Valentin Albillo
Senior Member

Posts: 822
Joined: Feb 2015
Warning Level: 0%

[VA] "Introducing APRIL !" microchallenge

Hi, all,

Let me introduce ***APRIL*** *(**A**rithmetic **P**roblem **R**ecently **I**dentified **L**ocally)*, which is an arithmetic problem I recently identified locally while using some vintage ***HP*** calcs. My identification might be recent but the problem itself has been lurking unnoticed for *40+* years, *AFAIK*.

To wit, there's a purely *arithmetic operation* that produces a result when executed in one series (*"Spice"*: **HP-33E/C**, **HP-34C** ...) and a ***different*** one when executed in another series (*"Nut" CPUs*: **HP-41**, **HP-10C**, **HP-15C** ...). The term *"arithmetic operation"* does include $\sqrt{x}$ but excludes all transcendental functions, statistics-related ones, y^x, `->H.MS` and `->HR`.

I thought that by the early *'80s* all basic arithmetic firmware would be thoroughly debugged and essentially *cast in stone* but it seems it *wasn't*, which hit me hard when this purely arithmetic program of mine (dealing with values in the range *[0.1 ... 1500]*) would produce *very* different results when run on said *HP* calcs, so it took me some sleuthing to isolate the culprit.

Now that you know *beyond a shadow of a doubt* that there's one, see if you can find such a faulty arithmetic operation. Unless this is another elaborate *April Fools' Day* joke !!.

V.

All My Articles & other Materials here: **Valentin Albillo's HP Collection**

PM WWW FIND EDIT QUOTE REPORT

04-01-2022, 11:43 PM **Post: #2**

J-F Garnier
Senior Member

Posts: 649
Joined: Dec 2013

RE: [VA] "Introducing APRIL !" microchallenge

Expressed in these terms, this challenge seems very vague, and almost impossible to investigate.
But - assuming it's a serious question, despite the white line - we can narrow the search with these considerations:
- the operation must be available on all cited machines,
- it is an arithmetic operation, excluding operations based on approximations (trig, log) and also excluding some operations that we know have problems on certain machines, such as >HMS.

Comparing the features of the two low-end machines, HP33E and 10C, we get:
- basic arithmetic operations: +, - , * and /
- 1/x, sqrt, x^2
- pi, %
- INT, FRAC

- >DEG, >RAD
Furthermore (if we trust the author), the argument is in the range of 0.1..1500 excluding corner cases with very large or very small numbers.

Let's look closer at the suspects:
- the basic operations + - * and / are based on proven algorithms, and are at the bottom of our list of suspects,
- 1/x is just 1 divided by x, so belongs to the above category,
- same as x^2, that is x times x.
- sqrt could be a possible suspect, but again it relies on a solid algorithm,
- pi is just a constant, nothing special to expect here.
- % is basically divide by 100.
- INT and FRAC are simple operations, again nothing special to expect here.
- >DEG and >RAD are conversion operations, basically a multiplication or division, so we are back to the basic operation case.

It seems that all the suspects have good reasons to claim they are innocent.
We need to review again our list, but that will be for tomorrow...

J-F

EMAIL PM WWW FIND QUOTE REPORT

04-02-2022, 05:02 AM **Post: #3**

Paul Dale
Senior Member

Posts: 1,757
Joined: Dec 2013

RE: [VA] "Introducing APRIL !" microchallenge

I think the key here is the *series* of operations. An incorrect rounding in *any one* of the these operations could easily cascade into a completely different result.

We know that the algorithms were improved for the Voyager series over the Spice. That the 41 has an improved algorithm also isn't a big surprise.

Finding a specific example is another matter.

Pauli

PM FIND QUOTE REPORT

04-02-2022, 10:13 AM (This post was last modified: 04-02-2022 10:15 AM by Martin Hepperle.) **Post: #4**

Martin Hepperle
Senior Member

Posts: 331
Joined: May 2014

RE: [VA] "Introducing APRIL !" microchallenge

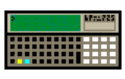
Maybe Y^X should not be excluded?

The limited parameter range of only up to 1500 also suggests something like LN/LOG (excluded), e^X (excluded).

I read "series" as referring to Spice or non-Spice machines, not "series of operations".

PM FIND QUOTE REPORT

04-02-2022, 10:48 AM **Post: #5**



J-F Garnier
Senior Member

Posts: 649
Joined: Dec 2013

RE: [VA] "Introducing APRIL !" microchallenge

> **Martin Hepperle Wrote:** (04-02-2022 10:13 AM)
> Maybe Y^X should not be excluded?

y^x is explicitly excluded, indeed there are easy-to-find differences with y^x between machines, for instance the recently discussed historic 3^201 test:
HP-33E: 7,968419661e95

HP-41C: 7,9684196644e95

But here we are looking for arithmetic operations, not transcendental function approximations.

J-F

EMAIL PM WWW FIND QUOTE REPORT

04-02-2022, 11:15 AM (This post was last modified: 04-02-2022 11:19 AM by C.Ret.) **Post: #6**

C.Ret
Member

Posts: 183
Joined: Dec 2013

RE: [VA] "Introducing APRIL !" microchallenge

> **J-F Garnier Wrote:** (04-01-2022 11:43 PM)
> It seems that all the suspects have good reasons to claim they are innocent.
> We need to review again our list, but that will be for tomorrow...

Since we are looking for a suspect that was silent for a so long time, we can't based you investigation only on suspicion. Any suspect have to be interrogated et tested.

Afaik, I just realize that I have only material from the "*nut*" part and no way to test any "*spice*" calculator 😕

> **Paul Dale Wrote:** (04-02-2022 05:02 AM)
> I think the key here is the *series* of operations.

But, **Valentin** clearly indicate that there is only one specific operation that produces a *different* result !
Can not be due to the way several operations are chained or any cumulative error's handled.

> **Martin Hepperle Wrote:** (04-02-2022 10:13 AM)
> The limited parameter range of only up to 1500 also suggests something like LN/LOG (excluded), e^X (excluded).

We can trust the indications given by **Valentin**, if these operations have to be exclude, that is for sure that they are out of the scope.

But you observation about the given range [0.1 , 1500] is for sure of a primary importance and may certainly indicate something else that what we are suspecting.

At first, this interval makes me believe it is about any trigonometric operation.
But **Valentin** clearly state and insist about a true, simple and single arithmetic operation.

So, I will let apart sin, cos, tan and √× for the moment.

For the arithmetic operations [+] or [-] this range is safe and may produce the same results even between oldest and old calculettes.

For arithmetic operation [×] this range is safe enough to produce similar results between "spice" and "nuts".

For arithmetic operation [÷] this range trigger me.
In one direction, it looks really safe and difference may really surprise me. But what about the opposite direction ?

In one hand:
1500 [ENTER^] 0.1 [÷] may display 15000. on all HP calculators without any worry about the age or the exact technology of the specific model.

In the other hand:
0.1 [ENTER^] 1500 [÷] may not necessary display the same exact result.

Unfortunately, as I already say it, I have no "spice" to test, but the few models ("nut") I just test all display :

0.000066667

May the "spice" have display this result a different way ?
For example an alternative format close to 6.666666-05 ?

PM FIND QUOTE REPORT

04-02-2022, 11:19 AM **Post: #7**

ijabbott
Senior Member

Posts: 1,145
Joined: Jul 2015

RE: [VA] "Introducing APRIL !" microchallenge

> **J-F Garnier Wrote:** (04-01-2022 11:43 PM)
> Comparing the features of the two low-end machines, HP33E and 10C, we get:
> - basic arithmetic operations: +, - , * and /
> - 1/x, sqrt, x²
> - pi, %
> - INT, FRAC
> - >DEG, >RAD

You could include REC->POL in that list as long as you ignore the angular part of the result.

— Ian Abbott

EMAIL PM FIND QUOTE REPORT

04-02-2022, 11:57 AM (This post was last modified: 04-02-2022 12:26 PM by J-F Garnier.) **Post: #8**

J-F Garnier
Senior Member

Posts: 649
Joined: Dec 2013

RE: [VA] "Introducing APRIL !" microchallenge

> **C.Ret Wrote:** (04-02-2022 11:15 AM)
> Afaik, I just realize that I have only material from the "*nut*" part and no way to test any "*spice*" calculator

You can use the MultiCalc emulator to check that there is no obvious differences.
But, as the most famous detectives, we don't need to go to the crime scene at first. We need first to use our *small grey cells* !

> **Quote:**
> Unfortunately, as I already say it, I have no "spice" to test, but the few models ("nut") I just test all display : 0.000066667
> May the "spice" have display this result a different way ?
> For example an alternative format close to 6.666666-05 ?

No, I don't think it's a matter of display, it would be a too bad April Fool. I'm sure it is difference in the real numbers.

J-F

EMAIL PM WWW FIND QUOTE REPORT

04-02-2022, 02:42 PM (This post was last modified: 04-02-2022 06:02 PM by Ángel Martin.) **Post: #9**

Ángel Martin
Senior Member

Posts: 1,341
Joined: Dec 2013

RE: [VA] "Introducing APRIL !" microchallenge

> **Paul Dale Wrote:** (04-02-2022 05:02 AM)
> We know that the algorithms were improved for the Voyager series over the Spice. That the 41 has an improved algorithm also isn't a big surprise.

But the 41 came out earlier than the Spice, so this would indicate a defect in the HP-33's version of the algorithms, which was later corrected for the Voyager, no?

PM FIND QUOTE REPORT

04-02-2022, 02:56 PM **Post: #10**

rprosperi
Super Moderator

Posts: 5,327
Joined: Dec 2013

RE: [VA] "Introducing APRIL !" microchallenge

> **Ángel Martin Wrote:** (04-02-2022 02:42 PM)
>
> > **Paul Dale Wrote:** (04-02-2022 05:02 AM)
> > I think the key here is the *series* of operations. An incorrect rounding in *any one* of the these operations could easily cascade into a completely different result.
> >
> > We know that the algorithms were improved for the Voyager series over the Spice. That the 41 has an improved algorithm also isn't a big surprise.
>
> But the 41 came out earlier than the Spice, so this would indicate a defect in the HP-33's version of the algorithms, which was later corrected for the Voyager, no?

No, the Spice models (all but the 34C) came out more than year earlier than the 41, while the 34C came out at the same time as the 41, as a lower-cost alternative.

--Bob Prosperi

EMAIL PM FIND QUOTE REPORT

04-02-2022, 06:01 PM **Post: #11**

Ángel Martin
Senior Member

Posts: 1,341
Joined: Dec 2013

RE: [VA] "Introducing APRIL !" microchallenge

> **rprosperi Wrote:** (04-02-2022 02:56 PM)
>
> > **Ángel Martin Wrote:** (04-02-2022 02:42 PM)
> > But the 41 came out earlier than the Spice, so this would indicate a defect in the HP-33's version of the algorithms, which was later corrected for the Voyager, no?
>
> No, the Spice models (all but the 34C) came out more than year earlier than the 41, while the 34C came out at the same time as the 41, as a lower-cost alternative.

Thanks for the clarification Bob, I guess what tripped was that SOLVE and INTEG were present on the HP-34 but not on the 41C, so I assumed the 34C came later. But of course the 33C was an earlier model anyway....

PM FIND QUOTE REPORT

04-02-2022, 08:12 PM **Post: #12**

J-F Garnier
Senior Member

Posts: 649
Joined: Dec 2013

RE: [VA] "Introducing APRIL !" microchallenge

Today we are the 2nd of April. No more joke, the Boss said "I want the culprit, now!".

We asked the Experts, they said: "The HP BCD math [in Spice/Voyager/Coconut machines] implements *arithmetic* algorithms described as '*infinite precision truncated to 13 digits*', then rounded *once* to the user 10-digit result. All is fully deterministic, no approximation, no random rounding error, every machine is giving *exactly* the same result for a given operation, and the result is the best representation of the result with 10 digits. No doubt, no any piece of randomness, no place for different answers of any kind. Period".

So we are taking our list again, and assuming the Experts are right, we take as an hypothesis, in *apparent*

contradiction with the OP's claim, that the arithmetic is right and without flaw. What can we still get out of our list of suspects?
Let's skip over +, -, *, /, 1/x, x².
Sqrt is basically (according to the same Experts) similarly implementing a pseudo-division (as the oldest of us may have learn at school looooong ago) and thus out of suspicion.
Let's forget %, INT and FRAC too.
PI is a constant, and we can easily check that its numeric value 3.141592654 is the best 10-digit representation of pi.

What is remaining now? The >DEG and >RAD function that are converting between radians and degrees using a constant ratio.
Can we really suspect them ? Well, "*when you have eliminated the impossible, whatever remains, however improbable, must be the truth*".
Let's see, the converting ratio is (pi/180) or (180/pi) of course, depending on the direction the conversion is done.
How can we check it?
Well, 1 >DEG gives 57.29577951, the best 10-digit representation of (180/pi), and
1 >RAD gives 0.01745329252 , the best 10-digit representation of (pi/180).

All this is perfectly correct but let's try now to do 7 >DEG. We should get 7*180/pi and we indeed get 401.0704566, the best representation of this value with 10 digits. Now notice that this is different from the user-level calculation (RPN style) 7 180 * PI / that gives 401.0704565 with an error of 1 ULP.
This proves that the >DEG and >RAD functions are using a high accuracy ratio value, presumably an internal 13-digit number, to guarantee the result.
Is it possible that some machines have a ratio value slightly wrong in the last places, that shows up only with certain numbers? How can we check it?
If this hypothesis is correct, then the problem would not be an arithmetic issue as such, but a simple programming flaw with no mystery.

Now that we have a plausible suspect, it's time to go back to the crime scene to look for evidences.
However, I need to stop my investigations for now, because I have a sextuple integral to compute for tomorrow morning :-)

J-F

EMAIL PM WWW FIND QUOTE REPORT

04-03-2022, 04:33 PM (This post was last modified: 04-03-2022 05:28 PM by J-F Garnier.) **Post: #13**

J-F Garnier
Senior Member

Posts: 649
Joined: Dec 2013

RE: [VA] "Introducing APRIL !" microchallenge

> **Paul Dale Wrote:** (04-02-2022 05:02 AM)
> We know that the algorithms were improved for the Voyager series over the Spice. That the 41 has an improved algorithm also isn't a big surprise.

We know that the algorithms were improved with the 13-digit precision on the HP-22, 29C and following to make 2^3=8, but I don't remember that the Voyager machines were improved over the Spice.
Do you have more details, or references?

It seems that this is correct, at least for the y^x function, as I mentioned above with the 3^201 test, but I know it since only recently when I tried to run this test on the 33E and the 41C.
BTW, the 34C has the same improvement (for the 3^201 test) as the Voyager and the 41C.
But I really would like to know more (not directly related to the issue reported by Valentin).

J-F

EMAIL PM WWW FIND QUOTE REPORT

04-03-2022, 06:16 PM **Post: #14**

rprosperi
Super Moderator

Posts: 5,327
Joined: Dec 2013

RE: [VA] "Introducing APRIL !" microchallenge

> **J-F Garnier Wrote:** (04-03-2022 04:33 PM)

> We know that the algorithms were improved with the 13-digit precision on the HP-22, 29C and following to make 2^3=8, but I don't remember that the Voyager machines were improved over the Spice.

Do you mean HP-22 here J-F, I think you mean the HP-27? The 22 was introduced in Aug '75 together with the original HP-25.

OTOH, the HP-27 was introduced in May '76, just 2 months after the HP-91, which is the 1st machine with "The New Accuracy" as cited in the Nov '76 HP Journal.

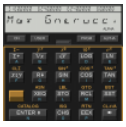
Not surprisingly, the HP-25C introduced a month later retained the original HP-25 ROM so did not inherit this new accuracy, but the -67 and -97 the month after that and then the -29C six months later certainly did.

What a great year 1976 was for HP model introductions!

--Bob Prosperi

EMAIL PM FIND QUOTE REPORT

04-03-2022, 06:57 PM **Post: #15**



Massimo Gnerucci
Senior Member

Posts: 2,467
Joined: Dec 2013

RE: [VA] "Introducing APRIL !" microchallenge

> **rprosperi Wrote:** (04-03-2022 06:16 PM)
> What a great year 1976 was for HP model introductions!

Bicentennial calculators!

Greetings,
Massimo

-+×÷ ↔ *left is right and right is wrong*

PM WWW FIND QUOTE REPORT

04-03-2022, 07:05 PM **Post: #16**

J-F Garnier
Senior Member

Posts: 649
Joined: Dec 2013

RE: [VA] "Introducing APRIL !" microchallenge

> **rprosperi Wrote:** (04-03-2022 06:16 PM)
> > **J-F Garnier Wrote:** (04-03-2022 04:33 PM)
> > We know that the algorithms were improved with the 13-digit precision on the HP-22, 29C and following to make 2^3=8, but I don't remember that the Voyager machines were improved over the Spice.
>
> Do you mean HP-22 here J-F, I think you mean the HP-27? The 22 was introduced in Aug '75 together with the original HP-25.

No, I mean the HP-22, this already has been discussed :-)

Anyway, my question was about the changes in the Voyager family.

J-F

EMAIL PM WWW FIND QUOTE REPORT

04-03-2022, 11:01 PM **Post: #17**

rprosperi
Super Moderator

Posts: 5,327
Joined: Dec 2013

RE: [VA] "Introducing APRIL !" microchallenge

> **J-F Garnier Wrote:** (04-03-2022 07:05 PM)
> No, I mean the HP-22, this already has been discussed :-)

Having been in that conversation, one might think I'd recall the details, but alas, no...

But this is where we discussed the long-to-market timing of the -91 I've read about, which re-reading that thread now, all makes sense.

For the Voyager family, I don't believe there was any extensive work to improve accuracy; I believe even Kahan commented on this noting something like "there wasn't time" to further tune the core math routines but rather the aim was to focus on application layers above the core routines for things like Bonds in the 12C, and refining the Solver and adding Matrices in the 15C, etc.

--Bob Prosperi

EMAIL PM FIND QUOTE REPORT

04-04-2022, 10:58 AM (This post was last modified: 04-04-2022 10:59 AM by J-F Garnier.) **Post: #18**

J-F Garnier
Senior Member

Posts: 649
Joined: Dec 2013

RE: [VA] "Introducing APRIL !" microchallenge

> **rprosperi Wrote:** (04-03-2022 11:01 PM)
> > **J-F Garnier Wrote:** (04-03-2022 07:05 PM)
> > No, I mean the HP-22, this already has been discussed :-)
> Having been in that conversation, one might think I'd recall the details, but alas, no...

Yes, we can't trust our memory, but the MoHPC is here to help. I just searched for "2^3 and accuracy" and that's it.
Also re-reading that thread, I found that the 3^201 test was already discussed too. That did I forget !

Thanks for the feedback on possible Voyager's changes.

Sorry Valentin for the digression, now it's time to go back to the "APRIL" case !

J-F

EMAIL PM WWW FIND QUOTE REPORT

04-04-2022, 08:37 PM (This post was last modified: 04-07-2022 06:15 AM by Valentin Albillo.) **Post: #19**

Valentin Albillo
Senior Member

Posts: 822
Joined: Feb 2015
Warning Level: 0%

RE: [VA] "Introducing APRIL !" microchallenge

.
Hi, all,

Thanks for your interest in my *microchallenge* and most definitely for your posts. I'll give here some comments to said posts so if you posted something, please read on. I also give a *date* for posting my *"original solution"*, so to say. Let's begin ...

> **J-F Garnier Wrote:**
> Expressed in these terms, this challenge seems very vague, and **almost impossible to investigate**.

Not so.

> **J-F Garnier Wrote:**
> But - **assuming it's a serious question** [...] Furthermore (**if we trust the author**), [...]

How rude ! ... What did I *ever* do to you !?

> **C.Ret Wrote:**
> Since we are looking for a suspect that was silent for a so long time, we can't based you investigation only on suspicion. **Any suspect have to be interrogated et tested**.

Very true.

> **C.Ret Wrote:**
> Valentin clearly indicate that **there is *only* one specific operation** that produces a different result !

Replace that *"**only**"* by *"**at least**"* and then your assertion will be correct. I never said *"only one"*, check my *OP* above if in doubt.

> **C.Ret Wrote:**
> But you observation about **the given range [0.1 , 1500] is for sure of a *primary* importance** and may certainly indicate something else that what we are suspecting.

Not really, that's the *approximate* range of values my program dealt with, that *0.1* could be *0.0037*, say, and that *1500* could be *1213.78*, I gave it just to let it be known that neither *extremely small* nor *extremely big* values were involved, just *"normal-ranged*" values.

> **C.Ret Wrote:**
> May the "spice" have **display this result a different way** ? For example an alternative format close to 6.666666-05 ?

Display format has *nothing* to do with this, the difference is between the *values* themselves, not how they're displayed, as **J-F** pointed out.

> **ijabbott Wrote:**
> **You could include REC->POL** in that list as long as you ignore the angular part of the result.

No, you *can't*, I said *"purely arithmetic operation"*, which *polar-rectangular* or vice versa conversions aren't, as they use trigs in both directions.

> **J-F Garnier Wrote:**
> Sorry Valentin for the digression, **now it's time to go back to the "APRIL" case** !

Fantastic ! I'm sure *you'll succeed.*

Enough comments. I fully expect that some of you kind and capable readers will be able to pinpoint such an instance as the one I serendipitously discovered when running my program and finding the results inconsistent with one another.

I'll post the *offending result* as well as my *original sleuthing* and the ***culprit operation*** next ***Thursday, 7*** so you have *3 more days* to try and tackle it.

Be aware that if I see no signs of *sufficient* effort on your part my final post might well be along the lines of *"Ha ha, it was just an elaborate April Fools' Day joke and you fell for it ! Let's forget about it !"* and let the matter rest for another 40 years.

Best regards.
V.
Edit: Corrected misattributed quote.
Edit: Corrected the deadline, it was meant to be Thursday 7, not Friday 8, sorry.

All My Articles & other Materials here: **Valentin Albillo's HP Collection**

PM WWW FIND EDIT QUOTE REPORT

04-04-2022, 09:08 PM (This post was last modified: 04-04-2022 09:13 PM by ijabbott.) **Post: #20**

ijabbott
Senior Member

Posts: 1,145
Joined: Jul 2015

RE: [VA] "Introducing APRIL !" microchallenge

> **Valentin Albillo Wrote:** (04-04-2022 08:37 PM)
>
> > **ijabbott Wrote:**
> > **You could include REC->POL** in that list as long as you ignore the angular part of the result.
>
> No, you *can't*, I said *"purely arithmetic operation"*, which *polar-rectangular* or vice versa conversions aren't, as they use trigs in both directions.

One of the results of POL->REC is purely arithmetical. I thought it was worth mentioning because it might use higher precision intermediate values than the equivalent operation using squaring, summation and square root that would be rounded to "normal" precision at every intermediate step.

— Ian Abbott

EMAIL PM FIND QUOTE REPORT

04-05-2022, 04:30 PM (This post was last modified: 04-06-2022 08:44 AM by J-F Garnier.) **Post: #21**

J-F Garnier
Senior Member

Posts: 649
Joined: Dec 2013

RE: [VA] "Introducing APRIL !" microchallenge

.
The "APRIL" Case, Day #5.

This is continuation of my previous post.

We are now concentrating on our primary suspect, the >DEG, >RAD pair, and are searching for evidences.
So we visited our source, the VASM 41 - the HP-41 source file location.
Quite easily, we found the sections related to >DEG (alias R-D) and >RAD (alias D-R) and found out how they are computed:

The routine RTOD does what we can translate in RPN style by: `90 * [PI/2] /`
Reciprocally, DTOR is doing : `[PI/2] * 90 /`
[PI/2] represents the 13-digit quantity `1.570796326795` and all arithmetic is done with 13 digits (truncated, not rounded).
The rounding to 10 digits is done at the end, and only once.

But we have no equivalent source for the Spice Series.
"*No problem*", our Experts said, "*We don't need the sources, we have access to an emulator able to trace the code (1), we will find it out.*"
And here is what we found for the 33E:
The equivalent of RTOD does `[PI/4] / 45 *` and DTOR does `45 / [PI/4] *`
[PI/4] represents the 13-digit quantity `0.7853981633975`, *exactly* half of the 41C's [PI/2] quantity.

So the problem doesn't come from a flawed constant.

Let's summarize:

```
Model | --- >DEG ---- | ---- >RAD ----
HP41C | 90 * [PI/2] / | [PI/2] * 90 /
HP33E | [PI/4] / 45 * | 45 / [PI/4] *
```

The 41C and 33E expressions are mathematically equivalent, but not necessary equivalent with finite floating-point numbers.

Let's look at the >DEG operation that is easier to analyse.

The 41C version does a multiplication by 90 first, then a division by [PI/2]
This is the optimum code, because we can be sure that any 10-digit number multiply by 90 will fit in 13 digits.
The operation is exact, so the accuracy of the whole >DEG operation is only the result of the next division, that we know (the Experts...) is providing, after rounding, the best (and unique) 10-digit representation of the quantity.

This is not the case with the 33E version.
Dividing by [PI/4] is generally not exact, the result is truncated to the 13-digit value just below the infinite precision value.
For these intermediate results that have a mantissa between 1 and about 2, the next multiplication by 45 will amplify the difference up to 4 ULP (here at the 13th digit).
For most numbers, the difference will be masked when rounding to 10 digits.
But for some numbers, that give a 13-digit result ending with the last 3 digits just around the threshold of ..500 between rounding up and down, the final 10-digit will be 1 ULP below the "best representation".

All we have to do now is to find an example.

Who will be the first (after Valentin) to find a 10-digit number that gives different results with >DEG on the 33E and 41C ?
It should not be too difficult now, we even have an idea of the probability of occurrence.
Instead of the 33E, you can use the emulator referenced below (1)

J-F

(1) The CCE33 emulator, part of the Multicalc project, that I already mentioned.

04-07-2022, 06:07 AM **Post: #22**

Valentin Albillo
Senior Member

Posts: 822
Joined: Feb 2015
Warning Level: 0%

RE: [VA] "Introducing APRIL !" microchallenge

Hi, all *(this was a microchallenge so this post won't be too long)*,

Thank you very much for your interest in my *microchallenge* and your excellent posts. Indeed this was *"a serious question"* and indeed *the author* (that's me) *"can be trusted"*, so this wasn't an *April Fools' Day* joke, despite the almost-unreadable text in white at the end of my *OP* which suggested the possibility (*that* was the joke !).

So, yes, there's a particular operation which returns some result when executed on an **HP-33**E/C and a different result when executed on a *'Nut' CPU* calculator, i.e., **HP-41C**, **HP-10C**, **HP-11C**, **HP-15C**, et-c.). The difference is very slight, just ***1 ulp*** for the instance I serendipitously found, but if used in a program, after a while the values obtained might *diverge* and the final result returned by the program might be *very* noticeably *different*, e.g. the difference between my program's result when run on an **HP-33**E/C *(0.**828**06 65707)* and when run on an **HP-41C** *(0.**821**63 42352)* was quite noticeable, some ~ *64,000,000 ulp* ! Oops !!

After being bitten by it, which left me truly amazed, I carefully traced the program's execution (involving tracing stack's and storage registers' contents, which was very laborious and time-consuming because I had to do it on *both* machines, not just one as is the typical case, having to compare the data after *each and every* number-altering program step) and after ***3,800+*** program steps had been executed I eventually found the *first* instruction (a *radians-to-degrees* conversion, `[->DEG]` or `[R-D]`) where the values were different, namely this instance:

```
HP-33C:   0.91528775  [->DEG] = 52.4421251 1

HP-10C:   0.91528775  [->DEG] = 52.4421251 2
HP-15C:   0.91528775  [->DEG] = 52.4421251 2
HP-41C:   0.91528775  [ R-D ] = 52.4421251 2
```

so for this instruction and this particular 8-digit argument (*0.91528775*) there's a ***1 ulp*** difference between the *Spice* **HP-33**E/C and the *Nut* **HP-41C**, **HP-10C**, **HP-15C**, ...

I then manually executed the equivalent conversion using 10-digit *RPN* user code (which is a less accurate way to

proceed than using the 13-digit internal conversion) and found that the result is the *same* for all of them, namely the one the **HP-33** `[->DEG]` built-in conversion produces:

```
0.91528775 [ENTER] 180 [*] (164.7517950) [Pi][/] = 52.44212511
```

Now, to see which value was the one correctly rounded to 10 digits (not that there was much doubt) and also to look for a possible reason (which I also suspected: a borderline rounding case), I computed it using several much more accurate calcs:

```
HP-71B:   DEG(0.91528775)           = 52.442125115 (12-digit result, uses 15-digit
precision internally)
Free42:   0.91528775 [->DEG]        = 52.442125115(02521 ...)
Win10 :   0.91528775 [x]180[÷][Pi][=] 52.442125115 02521 ...
```

so it's clear that the correct result when rounded to 10 digits is ***52.44212512***, the one produced by the *'Nut'* machines, while the *Spice* **HP-33**E/C's result differs by *1 ulp* for the reasons ***J-F Garnier*** exposed in his post above, where he correctly suspected that `[->DEG]` was the actual culprit, whose very *microcode* he then proceeded to thoroughly analyze by looking at the *VASM 41* for the **HP-41** and using an emulator able to trace the code for the **HP-33E**, then comparing both implementations while extensively detailing his sleuthing process and final conclusions. A *fantastic* detective work, ***kudos*** to **J-F** !

Alas, he stopped short of giving an actual instance *("All we have to do now is to find an example")* but he doesn't fool me, I take that statement as a polite way of letting me produce first the original instance I found several weeks ago, and I'm pretty sure he's already found and will produce a number of them after I've posted mine. If you want to have a go at it, let me assure you that it's not that difficult, I know of at least three different ways to produce more instances and **J-F** hinted at one, as we'll see next.

J-F also pointed out in general what I had observed in my *particular* sample instance, that the result truncated to 13 digits ends in `...502`, which is extremely close to the borderline `...500`, and this causes the **HP-33**E/C different (and *inferior*) implementation to produce the *wrong* 10-digit rounded result sometimes.

So there you are, after *40+ years* I've discovered *a new bug*, which is not the result of an algorithmic error but of a non-optimal implementation of a simple conversion's internal arithmetic. It's also a nice real-life example of *chaos*: an extremely small, *microscopic* difference (*1 ulp*, the smallest possible) gets amplified until it quickly becomes *macroscopic* (*64 million ulp*) and thus *very* noticeable, possibly altering the whole computation, which in my case it *did*.

I may add more comments later but for now let's see yours first, as well as your own instances if you managed to produce some.

Thanks again to ***all of you*** who posted for your interest, contributions and comments, much appreciated, and in particular to **J-F Garnier** for his time, fantastic sleuthing and hard work, I'm honored to get his high-quality contributions to my humble productions.

Best regards.
V.

All My Articles & other Materials here: **Valentin Albillo's HP Collection**

PM WWW FIND EDIT QUOTE REPORT

04-07-2022, 08:26 AM **Post: #23**

EdS2
Senior Member

Posts: 415
Joined: Apr 2014

RE: [VA] "Introducing APRIL !" microchallenge

A very interesting finding! Thanks for the thread - just my kind of thing, although I wasn't able to contribute this time around.

EMAIL PM FIND QUOTE REPORT

04-07-2022, 08:42 AM **Post: #24**

Werner
Senior Member

Posts: 701
Joined: Dec 2013

RE: [VA] "Introducing APRIL !" microchallenge

> **J-F Garnier Wrote:** (04-05-2022 04:30 PM)
> [PI/4] represents the 13-digit quantity `0.7853981633975`, *exactly* half of the 41C's [PI/2] quantity.
>
> *So the problem doesn't come from a flawed constant*.

While that is not the problem, the best 13-digit representation of PI/4 is `0.785398163397`**4**

All they had to do was swap the order of operations, and do 45* first.

Thanks Valentin and J-F, thoroughly enjoyed this one!
Werner
(That actually sounded as if there were other challenges that I *didn't* enjoy. That is not the case of course!)

EMAIL PM FIND QUOTE REPORT

04-07-2022, 10:11 AM (This post was last modified: 04-07-2022 10:57 AM by J-F Garnier.) **Post: #25**

J-F Garnier
Senior Member

Posts: 649
Joined: Dec 2013

RE: [VA] "Introducing APRIL !" microchallenge

> **Werner Wrote:** (04-07-2022 08:42 AM)
> While that is not the problem, the best 13-digit representation of PI/4 is `0.785398163397`**4**

Yes, I noticed that too.
Note however, that the next digits are 0.785398163397448 so both representations are almost equally good.
Matter of fact, the [PI/2] 13-digit value of the 41C is derived from an internal [PI/4] times 2, the same [PI/4] constant as in the 33E.
I guess the same constants are used by the trig reduction algorithm, and it was a convenient choice, ROM-space wise.

> **Quote:**
> All they had to do was swap the order of operations, and do 45* first.

Yes, and HP also changed the formula to use 90 and [PI/2] instead of 45 and [PI/4]. Why? I think this is related to the >RAD operation, that suffers from the same 1 ULP problem between the 33E and 41C.
Dividing by 90 may be less prone to error amplification than 45.
A simple justification (not a proof), is that dividing by 9 is similar to multiplying by 1.1111.. so the amplification is small (no more than 1 ULP) or even non existing. I didn't find any flaw in the 41C's >RAD.

This shows the level of both math and programming skills, but also dedication to quality (I may even say perfection) HP R&D had at the time. I'm pretty sure no customer ever complained about that small flaw in >DEG, nor any marketing asked to improve the degrees/radians conversion.

I will comment later on other aspects.

J-F

EMAIL PM WWW FIND QUOTE REPORT

04-07-2022, 11:09 AM **Post: #26**

Werner
Senior Member

Posts: 701
Joined: Dec 2013

RE: [VA] "Introducing APRIL !" microchallenge

> **J-F Garnier Wrote:** (04-07-2022 10:11 AM)
> Yes, and HP also changed the formula to use 90 and [PI/2] instead of 45 and [PI/4]. Why?
> ...
> Dividing by 90 may be less prone to error amplification than 45.

No, dividing by 90 and 45 has the same relative error, but PI/2 to 13 digits has an error of only 0.1 ulp instead of almost the maximum of 0.5 ulp for PI/4.

Cheers, Werner

EMAIL PM FIND QUOTE REPORT

04-07-2022, 12:12 PM **Post: #27**

J-F Garnier
Senior Member

Posts: 649
Joined: Dec 2013

RE: [VA] "Introducing APRIL !" microchallenge

> **Werner Wrote:** (04-07-2022 11:09 AM)
>
> > **J-F Garnier Wrote:** (04-07-2022 10:11 AM)
> >
> > Dividing by 90 may be less prone to error amplification than 45.
>
> No, dividing by 90 and 45 has the same relative error, but PI/2 to 13 digits has an error of only 0.1 ulp instead of almost the maximum of 0.5 ulp for PI/4.

I'm speaking of amplification of the absolute error in ULPs, not relative errors.
Suppose the previous multiply operation gives a mantissa of 1.00...01 instead of 1 (1 ULP error).
Dividing by 45 gives a mantissa of 2.22...24 (2 ULP error, i.e amplification by 2)
whereas dividing by 90 gives 1.11...12 (1 ULP error, no amplification).

J-F

EMAIL PM WWW FIND QUOTE REPORT

04-07-2022, 10:35 PM **Post: #28**

J-F Garnier
Senior Member

Posts: 649
Joined: Dec 2013

RE: [VA] "Introducing APRIL !" microchallenge

As promised, a few additional (and longer) comments:

> **Valentin Albillo Wrote:** (04-07-2022 06:07 AM)
>
> Thanks again to ***all of you*** [..] and in particular to **J-F Garnier** for his time, fantastic sleuthing and hard work, I'm honored to get his high-quality contributions to my humble productions.

You're welcome, Valentin. It is a kind of tribute for all the challenges you *concocted* during all these years, and that I enjoyed, even if I didn't participate to all (but to most of them, I believe).
Furthermore, the subject of this microchallenge was, by chance, one of my primary interests, meaning the BCD algorithms of the HP calculators.

I'm mainly interested by the 71B and generally the Saturn implementation, but when you look at the Coconut or Woodstock processors and so-called microcodes, you are immediately in a familiar land.
For instance, when I had to identify the >DEG and >RAD algorithms in the 33E, without sources or entry points, what I did in the emulator is to fill the program memory with >DEG (or >RAD) and a final GTO 1, and run it.
Breaking the execution a few times bring me soon in a familiar code with successive addition (or subtraction), register shift, and nibble pointer movement that I recognized as the 13-digit multiplication (or division) code. And the rest was not too difficult.

Some results.
Valentin guessed it right, a few days ago I identified several numbers that give different answers with >DEG .

To achieve my goal, I chose to write a short RPN code on Free42 that mimics the 13-digit computations of the 41C and 33E.
I fed it with random numbers and looked for the differences after rounding to 10 digits.
The code is at the end of this post, for the curious (I didn't clean it, so not necessary great programming, but I added comments to document it)

Here are a few examples with >DEG (spaces added for legibility):

```
1.67 94 93 89
1.61 16 94 84
1.10 94 96 008
```

I mentioned it already but >RAD on the 33E has a similar problem, for instance with:
89.92 93 93 42
74.64 51 01 72
88.84 66 525

(I deliberately chose a few "nice looking" numbers, there are many others)

A comment about the "error".
When we say the 33E has a 1 ULP error for some numbers, it's not really correct.
We should say the 33E answer is 1 ULP below the "best representation".
Both the 41C and 33E have an error of about 0.5 ULP error relative to the exact value, +0.5 ULP for one, -0.5 ULP for the other.
For the engineer this is not a problem, the error is (almost) the same on both machines.
This puts the issue in its real perspective.

A funny case.
I found a funny case with >RAD :
85.31 68 68 31 >RAD (another "nice looking" number...)
--> 1.489060260 on the 41
--> 1.489060259 on the 33E.
At the light of what I explained, you may say: OK the 33E rounded it down because of the few ULP internal error, and the 41C is right.
But NO! In this particular case, the 33E is *right*, and the 41C is *wrong* !
The exact value (here from Free42: 1.4890602594999910...) is so close to the ...500 limit for rounding that the small inaccuracy in the internal [PI/2] constant (in excess) makes the 41C jump to the wrong side of the fence for rounding.

The best for the end.
I will end with a very special, quasi-mystic number I found by chance with >DEG.
The number is
*** 1.23 78 465 ***
and >DEG gives
--> 70.92338014 on the 41
--> 70.92338013 on the 33E.

Isn't it a wonderful number? See:
8 digits only, with *all* the digits 1 to 8,
more important, with the digits exercising the *3 rows*
of the calculator numeric keypad *in turn*,
with the keys *from left to right* except the last two swapped,
as if it was designed to be keyed in our machines.

It seems to tell us "*Hey I've been waiting 40 years*
to demonstrate the change in the algorithm of >DEG !"
But here I'm leaving the world of computer science...

J-F

Code:
```
; Free42 prog for >DEG diff searching

 00 { 105-Byte Prgm }

 01▸LBL "SVA"
FIX 09  ; display mode for rounding to 10 digits
STO 00  ; input : number of iterations
0 STO 02  STO 03  STO 05 ; init some vars
; note: 13-digit [PI/4]=0.7853981633975 is in R04
PI  SEED ; init rnd seed
```

EMAIL PM WWW FIND QUOTE REPORT

04-08-2022, 12:59 AM **Post: #29**

rprosperi
Super Moderator

Posts: 5,327
Joined: Dec 2013

RE: [VA] "Introducing APRIL !" microchallenge

It's interesting to me that in his OP, Valentin commented about how the different machines "...produce *very different results* when run on said HP calcs".

In these cases, the results differ by about 1E-9. Can that really be considered a *very* different result, or am I missing something?

I think it would be very helpful for me (plus all the other less vocal folks) if someone could clearly explain the actual meaning of "1 ULP". I've taken lots of numerical analysis classes (in the late 70's) and written lots of long/complex Fortran programs so the issues of accuracy and precision are not new topics, but I've never been taught the term ULP, or how it is used here in posts like these. It is thrown around here often, to highlight and/or justify some seemingly obvious conclusions, etc. so having this explained would benefit the readers that aren't clued-in. Maybe I'm the only one... (that's willing to ask).

I'm not looking for someone to go to Wikipedia and copy/paste some succinct definition, I mean really explain it. For example, if it refers (as it seems) to some nth digit of accuracy, how can one refer to 0.5 ULP? (Note: any takers are requested to please start a new thread, no need to pollute Valentin's thread more than I have here).

--Bob Prosperi

EMAIL PM FIND QUOTE REPORT

04-08-2022, 09:46 AM (This post was last modified: 04-08-2022 01:32 PM by J-F Garnier.) **Post: #30**

J-F Garnier
Senior Member

Posts: 649
Joined: Dec 2013

RE: [VA] "Introducing APRIL !" microchallenge

ULP = Unit in the Last Place

Sorry, I may have clarified, since I used it a lot, so it's still on-topic.
It's the weight of the last place in a representation of a number in a given system.
Or, in an other way, the minimum change you can do on a number (in a given representation).
It is particularly useful with the BCD (Binary Coded Decimal) representations, although it may be used with binary representations too.
For instance the ULP of the PI representation is 1e-9 on the 41C, 1e-11 on the 71B, and 1e-33 on Free42 (Decimal)
When I used it as in 0.5 ULP, I just used it as a convenient unit. Nothing more complicate.

> **rprosperi Wrote:** (04-08-2022 12:59 AM)
> It's interesting to me that in his OP, Valentin commented about how the different machines "...produce *very different results* when run on said HP calcs".
> In these cases, the results differ by about 1E-9. Can that really be considered a *very* different result, or am I missing something?

I don't want to be Valentin's advocate (he doesn't need), but he wrote:
"*... [an] operation that produces a result when executed in one series (...) and a different one when executed in another series (..)*".
then later:
"*... [an] arithmetic program of mine (...) would produce very different results when run on said HP calcs*".
He mentioned in his comment how chaos can exponentially amplify a little perturbation, a well-known, purely deterministic mechanism (when applied to computing).
Seems clear to me (but I may not be the best to judge on this matter :-).

And finally, you may consider that:
- it's a micro-challenge,
- it was posted on April 1st,
- and all this is for fun first !

J-F

EMAIL PM WWW FIND QUOTE REPORT

04-08-2022, 02:04 PM **Post: #31**

Martin Hepperle
Senior Member

Posts: 331
Joined: May 2014

RE: [VA] "Introducing APRIL !" microchallenge

> **J-F Garnier Wrote:** (04-07-2022 10:35 PM)
>
> ...
>
> **The best for the end.**
> I will end with a very special, quasi-mystic number I found by chance with >DEG.
> The number is
> `*** 1.23 78 465 ***`
> and >DEG gives
> `--> 70.92338014` on the 41
> `--> 70.92338013` on the 33E.
> ...

Aiming a perfectly straight line beam, I would miss Mars (approximately 56 million km away) by about 10 meters (rounding by engineer, trusting his HP 49G) due to this error ;-)

PM FIND QUOTE REPORT

04-11-2022, 01:43 AM (This post was last modified: 04-11-2022 03:26 AM by Valentin Albillo.) **Post: #32**

Valentin Albillo
Senior Member

Posts: 822
Joined: Feb 2015
Warning Level: 0%

RE: [VA] "Introducing APRIL !" microchallenge

.
Hi, **Bob**,

> **rprosperi Wrote:** (04-08-2022 12:59 AM)
>
> It's interesting to me that in his OP, Valentin commented about how the different machines "...produce ***very different results*** when run on said HP calcs".
>
> In these cases, **the results differ by about 1E-9. Can that really be considered a *very* different result**, or am I missing something?

I couldn't comment earlier because doing it would spoil a key ingredient of my solution to the challenge featured in ***[VA] SRC #011 - April 1st, 2022 Bizarro Special***, but now that I've published my solution there's nothing to spoil anymore, so I'll tell you how I discovered that the **->DEG** instruction gave different results for some arguments when executed in the **HP33E** and in the **HP-41C** (where it's called **R-D**), so you'll perfectly understand my *"very different results"* statement. Let's see ...

My ***SRC#11*** thread asked for a program written for the **HP-10C** which would return the numeric value of a *sextuple integral* with three correct digits. Of course I knew that the one and only way to perform the feat would be using a ***Monte Carlo*** cubature method (nothing else would fit at all) and *that* required evaluating the integrand for a given number of samples, each requiring the generation and use of six pseudo-random numbers.

Due to the extremely little *RAM* available *(79 bytes)* and the lack of both a built-in *RNG (Random Number Generator)* and *subroutines*, an *RPN* user-code *RNG* had to be implemented and duplicated six times while having decent quality, so I decided to use the *RNG* I discovered and advocated 40+ years ago in *PPC CJ*, which features the **->DEG** *radians-to-degrees* instruction, i.e. using the built-in value for *180/Pi* as a suitable multiplier.

So far, so good. I readily wrote a *49 step* **HP-10C** program and as I had no *HP-10C* available, I executed it on the **HP-33E**, the most similar model, strictly using *HP-10C* capabilities (i.e. no subroutines, which the *HP-33E* has but the *HP-10C* lacks) and *RAM* limitations (i.e. 79 bytes for program *and* storage registers). As a test, I used ***1*** as the seed, and ***100*** as the number of samples, like this:

```
1 [STO 3] 100 [R/S] -> 0.8281  (0.82806 65707)
```

I was pretty confident that running this same program and test on an actual **HP-10C** would produce the exact same result, but just in case I ran it on the **HP-41C** (with `GTO` ***step*** changed to `GTO` ***label*** but otherwise identical), like this:

```
1 [STO 03] 100 [R/S] -> 0.8216  (0.82163 42352)
```

and lo and behold, to my utter amazement the results were ***noticeably different***, *~64 million ulp* different no less !! Of course, I was intent in discovering where did the computations first differ, so I executed both programs using *10, 20, ..., 90, 100* samples, and the only different result was the one for *100* samples, while up to *90* samples they all were identical.

I then refined the search by executing both programs for *91, 92* ... samples and though the *91* runs were identical, using *92* samples gave these results:

```
HP-33E: 92 samples -> 0.8561  (0.85609 10158) RCL  3 = 0.33190 28200 (seed at the end)
HP-41C: 92 samples -> 0.8561  (0.85609 10161) RCL 03 = 0.33190 33900 (ditto)
```

and it can be seen that the *results* differ by ***3*** *ulp* while the final *seeds* differ by ***5,700*** *ulp*.

Now, running it using again *92* samples but stopping the program after the *91th* sample had been processed and single-stepping (while stack-tracing) through the *92th* sample's processing, I finally discovered that the *X*-register had a different value after executing steps *28* and *29*, namely:

```
Step           X in HP-33E      X in HP-41C
-----------------------------------------------
28  RCL 3     0.91528 77500    0.91528 77500   (identical seed)
29  ->DEG    52.44212 511     52.44212 512     (1 ulp difference)
```

This is the very *first* time both programs differ, and the results in the *X*-register after executing **->DEG (R-D** in the **HP-41C**) with the exact same argument *0.91528 77500* differed by *1 ulp*. By this time **->DEG** (**R-D**) had been executed *550* times flawlessly but the *551th* time it gave a ***1*** *ulp* difference, which is the minimum possible but still a difference.

If it had remained a *1 ulp* difference until the execution's end it would be little problem. But as seen above, by the time the *92th* sample had been processed, the difference had grown to ***3*** *ulp*. and by the time the *100th* sample had been processed and the program halted, the difference had exploded to ***64*** *million ulp*.

This explosion is caused by the fact that the result of **->DEG** applied to the seed in **R3** is reduced to its fractional part and then stored *back ("feeded back")* in **R3** as the new seed and so the error compounds over time and essentially explodes in a *chaotic* way.

That's one of the effects of *chaos*: *any* difference in the initial conditions, however small, causes *completely different results* over time, so the final result essentially can't be predicted after a while.

Just to give an example taken from real life, if I remember correctly the position of ***Pluto*** in its orbit can be predicted with great accuracy over the next 10-20 million years or so, but after that, although the orbit path remains predictable, the *position* of *Pluto* in its orbit can't be correctly predicted anymore, it could be near its perihelion, near its aphelion, or somewhere in between.

> *P.S.: Besides the HP-33E and the HP-41C, I also ran the program and test case on the **HP-15C**, which agreed with the HP-41C's result, but to be extra sure I asked **Didier Lachieze** to run it on his physical **HP-10C**, which he most kindly did, and the result again agreed with the HP-15C's and HP-41C's, even when using up to **1,312** samples.*

Best regards.
V.
Edit: added link to Lyapunov Time (Wikipedia)

All My Articles & other Materials here: **Valentin Albillo's HP Collection**

04-11-2022, 03:28 PM **Post: #33**

Albert Chan
Senior Member

Posts: 1,905
Joined: Jul 2018

RE: [VA] "Introducing APRIL !" microchallenge

> **J-F Garnier Wrote:**
> Fantastic, amazing !
>
> **153** >DEG :
> HP41C: 8766.254266
> HP33E: 8766.254265
>
> **0.123456987** >DEG :
> HP41C: 7.073564307
> HP33E: 7.073564306
>
> Do post them !
> (and how you found them)

Thanks to J-F Garnier, confirmation from actual machine.

I found them in lua, working in **integers**.
This avoided dec->bin conversion errors.
2nd found is actually tested from 123456987 radians

Constant PI/2, PI/4 still use fractions, but dec->bin conversion error is tiny, under 0.14 ulp

float(1.570796326795) = 1.57079632679500003078 ...
float(0.7853981633975) = 0.78539816339750001539 ...

There may also be errors of truncation to 13 digits.
(also, possible rounding to 10 digits error, since I don't know what rules for half-way case)

Actual confirmation from physical decimal machines may be needed.

EMAIL PM FIND QUOTE REPORT

04-11-2022, 09:04 PM (This post was last modified: 04-11-2022 09:53 PM by Albert Chan.) **Post: #34**

Albert Chan
Senior Member

Posts: 1,905
Joined: Jul 2018

RE: [VA] "Introducing APRIL !" microchallenge

> **Werner Wrote:** (04-07-2022 11:09 AM)
>
> > **J-F Garnier Wrote:** (04-07-2022 10:11 AM)
> > Yes, and HP also changed the formula to use 90 and [PI/2] instead of 45 and [PI/4]. Why?
> > ...
> > Dividing by 90 may be less prone to error amplification than 45.
>
> No, dividing by 90 and 45 has the same relative error ...

At first, I agree with Werner.
After a while, I switched side, and agree with J-F Garnier.

ULP error analysis is hard, because ULP is on a variable scale, depends on actual number.

It is even harder when number is close to boundary base^n, with different size ULP.
see The Shortest Decimal String That Round-Trips May Not Be The Nearest

We can assume truncation error, in terms of ULP, for both /90 or /45 about the same.
However, relative error depends on size of mantissa (range from 0.100 to 0.999...)

Say, we only consider mantissa ranged of 0.9 to 1
For x/90, it is 10% (mantissa(x) = 0.81 to 0.90)
For x/45, it is 5% (mantissa(x) = 0.405 to 0.45)

For the same ULP error, the bigger the mantissa, the smaller its relative error.
In terms of relative errors, we would expect x/90 twice as accurate, compared to x/45
(might not be twice as accurate, due to Benford's Law, leading digit is likely to be small)

Below count bad rad(x), with 10 million random n-digits numbers

Intermediate calculations chopped to 13 digits.
Final result rounded to 10 digits, then compared with true rad(x)
pi2 = 1.570796326795
pi4 = .7853981633975 (wrongly rounded pi/4 helps a little here)

Code:

```
        x*pi2/90    x*pi4/45    x/90*pi2    x/45*pi4
n=9     2500        2500        3240        6277
n=10    2175        2176        4053        5134
```

The winner is to do division last.

EMAIL PM FIND QUOTE REPORT

04-12-2022, 08:45 AM **Post: #35**

Werner
Senior Member

Posts: 701
Joined: Dec 2013

RE: [VA] "Introducing APRIL !" microchallenge

Albert Chan Wrote: (04-11-2022 09:04 PM)

> **Werner Wrote:** (04-07-2022 11:09 AM)
> No, dividing by 90 and 45 has the same relative error ...

At first, I agree with Werner.
After a while, I switched side, and agree with J-F Garnier.

For the record, I have long switched sides as well, and don't agree with Werner any longer ;-)
Werner

EMAIL PM FIND QUOTE REPORT

« Next Oldest | Next Newest »

Enter Keywords Search Thread

NEW REPLY

View a Printable Version
Send this Thread to a Friend
Subscribe to this thread

User(s) browsing this thread: Valentin Albillo*