[VA] SRC#002- Almost integers and other beasties

Threaded Mode | Linear Mode

12-13-2018, 11:30 PM **Post: #1**

Valentin Albillo
Senior Member

Posts: 636
Joined: Feb 2015
Warning Level: 0%

[VA] SRC#002- Almost integers and other beasties

.
Hi all, welcome to my ***SRC#002 - Almost integers and other beasties***.

Here I'll show an assortment of the results I can get using my **HP-71B IDENTIFY** program version **2.0**. The original version 1.0 was extensively discussed and demonstrated with examples galore in my article ***Boldly Going ... Identifying Constants***, published more than 10 years ago. You can download it as a PDF document using this link:

Boldly Going ... Identifying Constants

Shortly after publishing it I expanded its already substantial capabilities with important additional features such as the ability to find *Minimal Polynomials* and other implicit expressions, which greatly increased the recognition of arbitrary constants, and further this version **2.0** can be used *creatively* to find **interesting, uncanny expressions** never before seen, like the following ones I found and which you might enjoy seeing and checking using your trusty *HP* calculator:

- if $x = \mathrm{Ln}(1+\pi)$ then $62x^2+123x$ = 300.000002
- if $x = \mathrm{Sin}\ e$ then $11x-9x^2$ = 2.9999227778868800
- if $x = \mathrm{Sin}\ 4$ then $5x(x-4)$ = 17.99979999
- if $x = \mathrm{Sin}\ 1 + \mathrm{Cos}\ 2 + \mathrm{Tan}\ 3$ then 6^3x-x^2 = 60.9999995
- if $x = \frac{3}{7}\sqrt[3]{\frac{261}{\pi}}$ then x^5-x = 21.0000000100
- if x = 1% of $\mathrm{Acos}\frac{-317}{664}$ then $\sqrt{10}-x$ = 3.14159265358
- if $x^X = \pi$ then $(x+6)(9x-x^3)$ = 80.999999999
- if $x = \mathrm{Gamma}\ \pi$ then $38x^2-3x^3$ = 163.00000

- if x is the positive root of $.7x^2-6x-236 = 0$, then $\mathrm{Ln}\ x$ = 3.14159265

Go ahead, check them, and I'd love to see any and all comments you would have on the matter, as well as your own uncanny expressions of a similar nature (**Gerson**, I'm looking at you :)), please post your very best, original ones discovered by you (no 3rd-party ones harvested on the Internet, please) as replies in this thread.

Regards.
V.
.

Find All My HP-related Materials here: **Valentin Albillo's HP Collection**

PM WWW FIND EDIT QUOTE REPORT

12-14-2018, 05:41 AM **Post: #2**

ttw
Member

Posts: 215
Joined: Jun 2014

RE: [VA] SRC#002- Almost integers and other beasties

The PDF had fonts the I couldn't read, but that doesn't stop me from commenting. (I is the internet.)

If interpreted your comments correctly, you compare by comparing fractions in lowest terms. This suggests (if you are not doing this already, converting decimals to continued fractions and do comparisons by generating a single partial quotient at each step. This could lead to a nice speedup as grossly different numbers could be eliminated quickly.

EMAIL PM FIND QUOTE REPORT

12-14-2018, 04:14 PM **Post: #3**

Gerson W. Barbosa
Senior Member

Posts: 1,361
Joined: Dec 2013

RE: [VA] SRC#002- Almost integers and other beasties

> **Valentin Albillo Wrote:** (12-13-2018 11:30 PM)
> Go ahead, check them, and I'd love to see any and all comments you would have on the matter, as well as your own uncanny expressions of a similar nature (**Gerson**, I'm looking at you), please post your very best, original ones discovered by you (no 3rd-party ones harvested on the Internet, please) as replies in this thread.

Hello, Valentin,

More or less in the same vein,

- if **x** = $\pi\sqrt{2}$ then $\left(12^2 - 5 \times 10^{-5}\right)^{\mathbf{x}+\mathbf{x}^{-1}}$ = 640.00000003

Here are a few more original near-integers and near-identities:

$$2\left(\pi + e - \psi\right) = 4.9999776$$

$$2\left(e - \tan^{-1}(e)\right) = 2.9999978$$

$$\ln\left(\frac{16\ln 878}{\ln(16\ln 878)}\right) = 3.14159265377$$

$$\frac{e^{\frac{23}{4}}}{100 + \frac{1}{100 + \frac{1}{\sqrt{100\sqrt{5}}}}} = 3.141592\,65354$$

$$3.141593 - \frac{\sqrt{3}}{5 \times 10^6} = 3.1415926535898$$

$$\frac{\ln\left(\sqrt{8} \cdot 10^8\right)}{\ln \pi} = 16.999994$$

$$\frac{\ln\left(2 \cdot \varphi^{39}\right)}{\ln \pi} = 17.00000026$$

Best regards,

Gerson

EMAIL PM FIND QUOTE REPORT

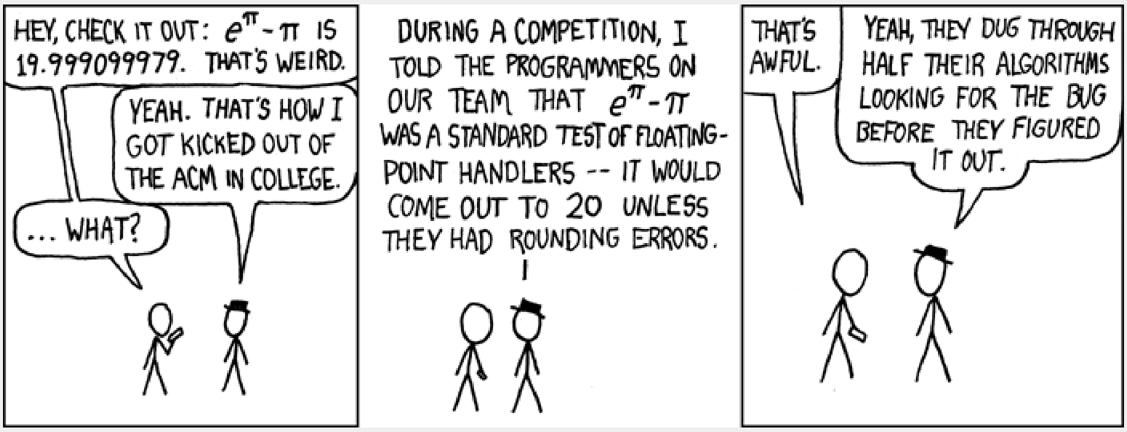
12-14-2018, 09:29 PM **Post: #4**

Thomas Klemm
Senior Member

Posts: 1,447
Joined: Dec 2013

RE: [VA] SRC#002- Almost integers and other beasties

> **Valentin Albillo Wrote:** (12-13-2018 11:30 PM)
> no 3rd-party ones harvested on the Internet, please



PM FIND QUOTE REPORT

12-14-2018, 09:58 PM **Post: #5**

rprosperi
Senior Member

Posts: 4,439
Joined: Dec 2013

RE: [VA] SRC#002- Almost integers and other beasties

> **Gerson W. Barbosa Wrote:** (12-14-2018 04:14 PM)
>
> $$\ln\left(\frac{16\ln 878}{\ln(16\ln 878)}\right) = 3.14159265377$$

There is something beautiful and compelling about this one, at least for me!

Both you guys truly amaze me... in a good way, just to be clear....

--Bob Prosperi

EMAIL PM FIND QUOTE REPORT

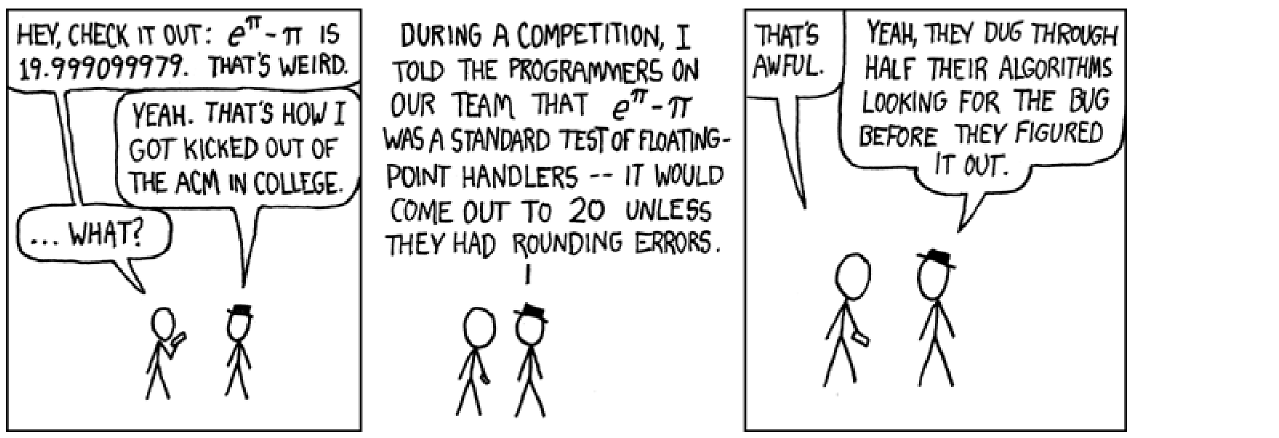
12-14-2018, 10:41 PM (This post was last modified: 12-14-2018 10:47 PM by Gerson W. Barbosa.) **Post: #6**

Gerson W. Barbosa
Senior Member

Posts: 1,361
Joined: Dec 2013

RE: [VA] SRC#002- Almost integers and other beasties

> **Thomas Klemm Wrote:** (12-14-2018 09:29 PM)
>
> > **Valentin Albillo Wrote:** (12-13-2018 11:30 PM)
> > no 3rd-party ones harvested on the Internet, please



I would humbly suggest an even more comprehensive test:

$$e^{\pi} - \pi + \left(\frac{9}{3\times 10^2 - \frac{2\sqrt{3}}{10^3 - \ln(2+\sqrt{2})}} \right)^2 = 20.000000000000000$$

EMAIL PM FIND QUOTE REPORT

12-15-2018, 01:19 AM **Post: #7**

Gerson W. Barbosa
Senior Member

Posts: 1,361
Joined: Dec 2013

RE: [VA] SRC#002- Almost integers and other beasties

Or this one, for 10-digit calculators:

$$\pi^2 + \frac{e^2}{33(\ln(\pi))^4} = 10.00000000$$

EMAIL PM FIND QUOTE REPORT

12-15-2018, 10:20 PM **Post: #8**

Valentin Albillo
Senior Member

Posts: 636
Joined: Feb 2015
Warning Level: 0%

RE: [VA] SRC#002- Almost integers and other beasties

.
Hi, ttw:

> **ttw Wrote:** (12-14-2018 05:41 AM)
> This suggests (if you are not doing this already, converting decimals to continued fractions and do comparisons by generating a single partial quotient at each step.

Thanks for your interest and comment. I do use a continued fraction algorithm to convert the arbitrary constant supplied by the user to a rational fraction, generating partial quotients one by one till the user-supplied accuracy is met.

Regards.
V.
.

Find All My HP-related Materials here: **Valentin Albillo's HP Collection**

PM WWW FIND EDIT QUOTE REPORT

12-15-2018, 10:56 PM (This post was last modified: 12-16-2018 04:48 AM by Valentin Albillo.) **Post: #9**

Valentin Albillo

Posts: 636

Senior Member

Joined: Feb 2015
Warning Level: 0%

RE: [VA] SRC#002- Almost integers and other beasties

.
Hi, Gerson:

> **Gerson W. Barbosa Wrote:** ⇒ (12-14-2018 04:14 PM)
> More or less in the same vein,

Thanks for your excellent findings, I was sure you'd never fail to contribute some amazing near-identities to this thread. As **Bob Prosperi** already pointed out, I too find this one particularly **beautiful**:

> **Quote:**
>
> $$\ln\left(\frac{16\ln 878}{\ln(16\ln 878)}\right) = 3.14159265377$$

Good finding indeed !

By the way, it's quite nice that the simple function **x/Ln(x)** sometimes gives *almost-integer* results for integer arguments (which means its graphic passes *extremely close* to integer-coordinates grid points), such as the following, in increasing order of *"closeness"*:

```
  x                x/Ln(x)
---------------------------------
  17              6.0002541...
  163            31.9999987...
  53453        4910.0000012...
  110673       9529.0000006...
  715533      53078.0000004...
```

so that we have, for instance,

53453/Ln(53453) = 4910.0000012...

In your case the argument **x=16*Ln(878)** results in **x/Ln(x)** being **23,140692636**9... which is almost the famous **Gelfond's constant = e^Pi** (the easiest transcendental number to compute to high precision) so its natural logarithm is very nearly **Pi** itself.

Nice catch ! 🙂

Have a fine weekend and best regards
V.
.

Find All My HP-related Materials here: **Valentin Albillo's HP Collection**

PM WWW FIND EDIT QUOTE REPORT

12-16-2018, 10:06 AM **Post: #10**

EdS2
Member

Posts: 255
Joined: Apr 2014

RE: [VA] SRC#002- Almost integers and other beasties

.

> **Valentin Albillo Wrote:** ⇒ (12-15-2018 10:56 PM)
> By the way, it's quite nice that the simple function **x/Ln(x)** sometimes gives *almost-integer* results for integer arguments...

Thanks, that leads to a rabbit hole of interesting links (OEIS and Mathoverflow.)

EMAIL PM FIND QUOTE REPORT

12-16-2018, 04:51 PM **Post: #11**

ttw
Member

Posts: 215
Joined: Jun 2014

RE: [VA] SRC#002- Almost integers and other beasties

Exp(Pi*Sqrt(163)) is one of the classic examples. The explanation is rather complicated. Other expressions, for example: ({[Sqrt(5)+1]/2}^n)/Sqrt(5) is close to the Fibonacci numbers; in fact ({[Sqrt(5)+1]/2}^n-{[1-Sqrt(5)]/2}^n)/Sqrt(5) is the well-known Binet formula for Fibonacci numbers. This sequence works by successive approximation to an integer the 163 sequence just seems to happen.

EMAIL PM FIND QUOTE REPORT

12-16-2018, 11:43 PM **Post: #12**

Valentin Albillo
Senior Member

Posts: 636
Joined: Feb 2015
Warning Level: 0%

RE: [VA] SRC#002- Almost integers and other beasties

> **EdS2 Wrote:** (12-16-2018 10:06 AM)
> Thanks, that leads to a rabbit hole of interesting links (OEIS and Mathoverflow.)

Interesting. I did not consult the **OEIS** for the results I gave above for **x/Ln(x)**, I simply obtained them myself by running this trivial **HP-71B** program I wrote in J-F Garnier's **Emu71** to quickly find them:

```
1  DESTROY ALL @ M=1 @ I=2
2  X=I/LN(I)  @ N=ABS(X-IROUND(X)) @ IF N<M THEN M=N @ DISP I;,X
3  I=I+1 @ GOTO 2

>RUN

2    2.88539008178
5    3.10667467281
9    4.09607651981
13    5.06832618827
17    6.00025410569
163    31.9999987385
53453    4910.00000122
110673    9529.00000068
715533    53078.0000004
1432276    101044
...         ...
```

Substituting **X=I/LN(I)** at line *2* by some other function and rerunning the program will result in a new set of almost-integer values, for instance:

```
2 X=I/TAN(I) ...

>RUN

 2    -.915315108721
 3    -21.0457576543
 7    8.03260795684
 37    -44.0072133321
 48    39.9957590124
 128    -123.004197859
 170    460.0010337
 1489    -12899.9995967
 2106    986.000155144
 11923    15493.9999873
```

i.e.: **1489/Tan(1489) =** -12899.9995967 ~ **-12900**

and so on and so forth. Trivial variations of this trivial program will produce an infinitude of almost-integer valued expressions of all kinds.

Thanks for your interest and links.

V.
.

Find All My HP-related Materials here: **Valentin Albillo's HP Collection**

PM WWW FIND EDIT QUOTE REPORT

12-19-2018, 11:23 PM (This post was last modified: 12-20-2018 10:56 PM by Valentin Albillo.) **Post: #13**

Valentin Albillo
Senior Member

Posts: 636
Joined: Feb 2015
Warning Level: 0%

RE: [VA] SRC#002- Almost integers and other beasties

.
Hi, all:

> **I Wrote:** (12-16-2018 11:43 PM)
> Trivial variations of this trivial program will produce an infinitude of almost-integer valued expressions of all kinds.

A few additional, nice *almost-integer* results obtained that way:

```
5e^Acos(178/181)   = 6.0000000066

9e^Acos(538/541)   = 10.000000000023

8e^Acos(430/433)   = 9.000000000048

Ln 146 + Sin 614   = 4.0000080001 4

Ln 455 + Cos 188   = 7.0000003 4

Ln 231 + Tan 87    = 4.0000002 3

Gamma(314/709)     = 2.0000004 7
```

All trig functions, in radians.

V.
.

Find All My HP-related Materials here: **Valentin Albillo's HP Collection**

PM WWW FIND EDIT QUOTE REPORT

01-03-2019, 12:45 AM **Post: #14**

Gerson W. Barbosa
Senior Member

Posts: 1,361
Joined: Dec 2013

RE: [VA] SRC#002- Almost integers and other beasties

```
640000*x^5-768000*φ^2*x^4+3000+ln(2)=0
```

EMAIL PM FIND QUOTE REPORT

01-03-2019, 07:29 AM **Post: #15**

Valentin Albillo
Senior Member

Posts: 636
Joined: Feb 2015
Warning Level: 0%

RE: [VA] SRC#002- Almost integers and other beasties

.
Hi, **Gerson**:

> **Gerson W. Barbosa Wrote:** (01-03-2019 12:45 AM)
> `640000*x^5-768000*φ^2*x^4+3000+ln(2)=0`

I only have an iPad at hand right now so running this *extremely* quick'n'dirty Newton on it produces the intended root of your polynomial, namely:

```
10 def fnf(x)=640000*x^5-768000*p^2*x^4+3000+log(2)
20 def fnd(x)=(fnf(x+0.0001)-fnf(x-0.0001))/0.0002
30 p=(1+sqr(5))/2:input x0:home
35 for i=1 to 15
40 x1=x0-fnf(x0)/fnd(x0)
50 print x1;" ";fnf(x1)
60 x0=x1
70 next i

Run

?10
8.167851990851785 14317006965.841368
6.715802401810718 4653139559.305097
5.573461555087618 1501801189.590903
4.687642795302511 477761112.5232644
4.021032838991736 147136983.11165047
3.551962197641171 41803117.36241603
3.2712995361031516 9506051.502593396
3.1593486482128053 1132110.8530623894
3.141983056899174 24349.060661761047
3.141592847539331 12.090443311217141
3.1415926535896097 0.0000030975368739971643
3.14159265358956 -3.170697671084355e-8
3.1415926535895604 -1.904654323148236e-9
3.1415926535895604 -1.904654323148236e-9
```

3.1415926535895604 -1.904654323148236e-9

which is a nice approximation to Pi, congrats and thanks for sharing. Perhaps it's even more accurate than what the iPad produces but right now I can't tell ...

Regards.
V
.

Find All My HP-related Materials here: **Valentin Albillo's HP Collection**

PM WWW FIND EDIT QUOTE REPORT

01-03-2019, 07:48 AM (This post was last modified: 01-03-2019 07:49 AM by Paul Dale.) **Post: #16**

Paul Dale
Senior Member

Posts: 1,662
Joined: Dec 2013

RE: [VA] SRC#002- Almost integers and other beasties

The underline finishes one digit too far:

> **Valentin Albillo Wrote:** ⇒ (01-03-2019 07:29 AM)
>
> **3.1415926535895604** -1.904654323148236e-9

3.1415926535897932

Learning some leading digits of π is useful (for trolling).

Pauli

PM FIND QUOTE REPORT

01-03-2019, 08:36 AM **Post: #17**

Valentin Albillo
Senior Member

Posts: 636
Joined: Feb 2015
Warning Level: 0%

RE: [VA] SRC#002- Almost integers and other beasties

> **Paul Dale Wrote:** ⇨ (01-03-2019 07:48 AM)
> The underline finishes one digit too far:

Nope, it ends exactly where I intended it to end.

And yes, you are trolling.

V.
.

Find All My HP-related Materials here: **Valentin Albillo's HP Collection**

PM WWW FIND EDIT QUOTE REPORT

01-03-2019, 12:56 PM **Post: #18**

Gerson W. Barbosa
Senior Member

Posts: 1,361
Joined: Dec 2013

RE: [VA] SRC#002- Almost integers and other beasties

> **Paul Dale Wrote:** ⇨ (01-03-2019 07:48 AM)
> **3.1415926535897932**
>
> 🙂 Learning some leading digits of π is useful (for trolling).
>
> Pauli

Or for spotting mistakes like this one (YouTube video).
I used to remember the first 16 digits when I used my HP-200LX on a regular basis (its screen has gone dark and I have no spare part to replace it again). Anyway, I don't remember any "747" sequence occurring so early in π.

Gerson.

EMAIL PM FIND QUOTE REPORT

01-03-2019, 01:51 PM **Post: #19**

Paul Dale
Senior Member

Posts: 1,662
Joined: Dec 2013

RE: [VA] SRC#002- Almost integers and other beasties

The first "747" occurs at position 740 (thanks to the π searcher).

In the fullness of π, this is right near the start (of course).

Pauli

PM FIND QUOTE REPORT

01-03-2019, 11:33 PM **Post: #20**

Valentin Albillo
Senior Member

Posts: 636
Joined: Feb 2015
Warning Level: 0%

RE: [VA] SRC#002- Almost integers and other beasties

> **I Wrote:** ⇨ (12-19-2018 11:23 PM)
> A few additional, nice *almost-integer* results obtained that way:

For completeness, I forgot to include this remarkable one which I discovered and posted here last March:

`Sin(9*(Sin 1 + Cos 40)) =` **0.9999999999999998**30826985368...

which differs from the integer **1** by about *1e-16.*

V.
.

Find All My HP-related Materials here: **Valentin Albillo's HP Collection**

PM WWW FIND EDIT QUOTE REPORT

01-23-2019, 11:58 PM **Post: #21**

Valentin Albillo
Senior Member

Posts: 636
Joined: Feb 2015
Warning Level: 0%

RE: [VA] SRC#002- Almost integers and other beasties

Hi again,

Dealing with some search theory, I recently came across this short expression:

$$\frac{1}{12} + \frac{\Pi^2}{6Ln^2(2)} + \frac{2}{Ln(2)} \sum_{k=1}^{\infty} \frac{(-1)^k}{k(2^k - 1)}$$

which this straightforward **HP-71B** program *(a command-line expression would work as well)* quickly evaluates to the *apparently* exact *integer* value **1**:

```
1     DESTROY ALL @ V=0 @ FOR K=1 TO 40 @ V=V+(-1)^K/K/(2^K-1) @ NEXT K
2     V=V*2/LN(2)+1/12+PI^2/6/LN(2)^2 @ DISP V

>RUN
        1
```

But it's not !
See if you can get the actual, more accurate *almost-integer* value with your HP. It certainly qualifies as a most amazing almost-integer one ! :)

V.
.

Find All My HP-related Materials here: **Valentin Albillo's HP Collection**

PM WWW FIND EDIT QUOTE REPORT

02-09-2019, 10:53 PM **Post: #22**

Gerson W. Barbosa
Senior Member

Posts: 1,361
Joined: Dec 2013

RE: [VA] SRC#002- Almost integers and other beasties

> **Valentin Albillo Wrote:** (01-03-2019 11:33 PM)
> For completeness, I forgot to include this remarkable one which I discovered and posted here last March:
>
> `Sin(9*(Sin 1 + Cos 40)) = ` **0.9999999999999998**`30826985368...`
>
> which differs from the integer **1** by about *1e-16.*

Beautiful one! Hope you don’t mind if I uglify it a bit:

Sin(9*(Sin 1 + Cos 40) + 5/(e*10^8))

Best regards,

Gerson.

EMAIL PM FIND QUOTE REPORT

02-09-2019, 11:33 PM **Post: #23**

ttw
Member

Posts: 215
Joined: Jun 2014

RE: [VA] SRC#002- Almost integers and other beasties

There are quite a few of these strewn about mathematical sites on the internet. Some have easy explanations. Some have very deep explanations. Some have no seeming reason at all.

Sin(11) = about -1
Exp(pi)-pi = about 20
Cos(pi*Cos(pi*Cos(Ln(pi+20)))) = about 1
Exp(pi*Sqrt(163)) = about 262537412640768744

EMAIL PM FIND QUOTE REPORT

02-09-2019, 11:56 PM **Post: #24**

Valentin Albillo
Senior Member

Posts: 636
Joined: Feb 2015
Warning Level: 0%

RE: [VA] SRC#002- Almost integers and other beasties

.
Hi, **Gerson**:

> **Gerson W. Barbosa Wrote:** (02-09-2019 10:53 PM)
>
> > **Valentin Albillo Wrote:** (01-03-2019 11:33 PM)
> >
> > For completeness, I forgot to include this remarkable one which I discovered and posted here last March:
> >
> > `Sin(9*(Sin 1 + Cos 40)) = 0.99999999999999830826985368...`
> >
> > which differs from the integer **1** by about *1e-16.*
>
> Beautiful one! **Hope you don't mind if I uglify it a bit:**
>
> **Sin(9*(Sin 1 + Cos 40) + 5/(e*10^8))**

I don't understand ... :-(

Also, could you accurately evaluate the expression I gave in my post #21 in this thread (the previous one to yours) and if so, what accurate result did you get ? The proximity to the exact integer 1 is *not* a coincidence, there are deep mathematical reasons for it.

Best regards and have a nice weekend.
V.
.

Find All My HP-related Materials here: **Valentin Albillo's HP Collection**

PM WWW FIND EDIT QUOTE REPORT

02-10-2019, 03:41 AM **Post: #25**

Gerson W. Barbosa
Senior Member

Posts: 1,361
Joined: Dec 2013

RE: [VA] SRC#002- Almost integers and other beasties

> **Valentin Albillo Wrote:** (02-09-2019 11:56 PM)
>
> .
> Hi, **Gerson**:
>
> > **Gerson W. Barbosa Wrote:** (02-09-2019 10:53 PM)

Beautiful one! **Hope you don't mind if I uglify it a bit:**

Sin(9*(Sin 1 + Cos 40) + 5/(e*10^8))

I don't understand ...

Hi, Valentin,

The factor of *e* doesn´t appear to fit nicely in the trigonometric expression, but it makes the result get a bit closer to 1:

Sin(9*(Sin 1 + Cos 40) + 5/(e*10^8)) = **0.9999999999999999999999999**7740056822767

Valentin Albillo Wrote: ⇨ (02-09-2019 11:56 PM)

Also, could you accurately evaluate the expression I gave in my post #21 in this thread (the previous one to yours) and if so, what accurate result did you get ? The proximity to the exact integer 1 is *not* a coincidence, there are deep mathematical reasons for it.

Best regards and have a nice weekend.
V.
.

```
%%HP: T(3)A(R)F(.);
\<< 126 40 'DIGITS' STO 0 1 ROT
  FOR k 1 FNEG k FY\|^X k FDIV 2 k FY\|^X 1 FSUB FDIV FADD
  NEXT DUP FADD 2 FLN FDIV 1 12 FDIV FADD FPI FSQ 6 FDIV 2 FLN FSQ FDIV FADD ZZ\<-\->F DROP \-
>STR DUP HEAD "." + SWAP TAIL +
\>>

EVAL ->

"1.00000000000123741257573611022871961 0648"
```

The RPL program above requires the LongFloat library.

Number of iterations = Ceil(W(10^n*ln(2))/ln(2)), where n = number of digits and W(x) is the Lambert W function.

Best regards,

Gerson.

EMAIL PM FIND QUOTE REPORT

02-10-2019, 01:48 PM (This post was last modified: 02-10-2019 01:56 PM by Gerson W. Barbosa.) **Post: #26**

Gerson W. Barbosa
Senior Member

Posts: 1,361
Joined: Dec 2013

RE: [VA] SRC#002- Almost integers and other beasties

Gerson W. Barbosa Wrote: ⇨ (02-10-2019 03:41 AM)

```
%%HP: T(3)A(R)F(.);
\<< 126 40 'DIGITS' STO 0 1 ROT
  FOR k 1 FNEG k FY\|^X k FDIV 2 k FY\|^X 1 FSUB FDIV FADD
  NEXT DUP FADD 2 FLN FDIV 1 12 FDIV FADD FPI FSQ 6 FDIV 2 FLN FSQ FDIV FADD ZZ\<-\->F DROP \-
>STR DUP HEAD "." + SWAP TAIL +
\>>

EVAL ->

"1.00000000000123741257573611022871961 0648"
```

The RPL program above requires the LongFloat library.

Number of iterations = Ceil(W(10^n*ln(2))/ln(2)), where n = number of digits and W(x) is the Lambert W function.

This is a more generic version that takes the desired number of digits, *n*, as an argument. As I am using an approximation for W(x), the estimation of the required number of iterations might not be exact for small values of n.

Both programs are basically a straightforward conversion of Valentin's HP-71B program in post #21.

```
100

%%HP: T(3)A(R)F(.);
\<< RCLF SWAP -105 CF -3 CF DUP 'DIGITS' STO ALOG 2 LN * LN DUP LN - LASTARG SWAP / + 2 LN / CEIL 0
1 ROT
  FOR k 1 FNEG k FY\|^X k FDIV 2 k FY\|^X 1 FSUB FDIV FADD
  NEXT DUP FADD 2 FLN FDIV 12 FINV FADD FPI FSQ 6 FDIV 2 FLN FSQ FDIV FADD ZZ\<-\->F NEG SWAP \->STR
DUP SIZE ROT \=/ -51 FC? { "." } { "," } IFTE UNROT { DUP TAIL SWAP HEAD } { "0" } IFTE UNROT + +
SWAP STOF
\>>

EVAL ->

1.0000000000012374125757361102287196106466728742977320481965484438441718256405304288509138855861935 25
```

`1.00000000000123741257573611022871961064667287429773204819654844384417182564053042885091388558619352 5` (132.59 seconds on the HP 50g)

```
 5 -> 0.99990

 6 -> 1.00000

 7 -> 1.000000

12 -> 1.00000000001

20 -> 1.0000000000012374126

40 -> 1.000000000001237412575736110228719610648

50 -> 1.0000000000012374125757361102287196106466728742977
```

EMAIL PM FIND QUOTE REPORT

02-10-2019, 04:26 PM (This post was last modified: 02-10-2019 05:19 PM by Albert Chan.) **Post: #27**

Albert Chan
Senior Member

Posts: 1,226
Joined: Jul 2018

RE: [VA] SRC#002- Almost integers and other beasties

> **Gerson W. Barbosa Wrote:** (02-10-2019 01:48 PM)
>
> Number of iterations = Ceil(W(10^n*ln(2))/ln(2)), where n = number of digits and W(x) is the Lambert W function.
>
> `50 -> 1.0000000000012374125757361102287196106466728742977`

Above 50 digits numbers are confirmed corrrect.
Can you explain how the iteration count formula is derived ?

> **Valentin Albillo Wrote:** (01-23-2019 11:58 PM)
>
> $$\frac{1}{12}+\frac{\Pi^2}{6Ln^2(2)}+\frac{2}{Ln(2)}\sum_{k=1}^{\infty}\frac{(-1)^k}{k(2^k-1)}$$

I see that the sum gained about 1 bit precision with each iteration, so I use n / log10(2) ~ 3.322n
It is slightly over-estimated, but not by much.

Edit: the difference is the effect of 1/k, iterations ~ 3.322n - log2(3.322n)

EMAIL PM FIND QUOTE REPORT

02-10-2019, 05:06 PM (This post was last modified: 02-10-2019 05:14 PM by Gerson W. Barbosa.) **Post: #28**

Gerson W. Barbosa
Senior Member

Posts: 1,361
Joined: Dec 2013

RE: [VA] SRC#002- Almost integers and other beasties

> **Albert Chan Wrote:** (02-10-2019 04:26 PM)

> **Gerson W. Barbosa Wrote:** (02-10-2019 01:48 PM)
>
> > Number of iterations = Ceil(W(10^n*ln(2))/ln(2)), where n = number of digits and W(x) is the Lambert W function.
> >
> > ```
> > 50 -> 1.0000000000012374125757361102287196106466728742977
> > ```
>
> Above 50 digits numbers are confirmed corrrect.
> Can you explain how the iteration count formula is derived ?

Starting with $\log_{10}(k.2^k) = n$, I solved $k.2^k = 10^n$ for k. Well, actually W|A did :-)

$k = W(10^n.\ln(2))/\ln(2)$

BTW, it's better to replace `ALOG 2 LN * LN` with `10 LN * 2 LN LN +` just in case you want to evaluate it to one thousand digits or more:

```
1000

\<< RCLF SWAP -105 CF -3 CF DUP 'DIGITS' STO 10 LN * 2 LN LN + DUP LN - LASTARG SWAP / + 2 LN / CEIL
0 1 ROT
FOR k 1 FNEG k FY\|^X k FDIV 2 k FY\|^X 1 FSUB FDIV FADD
NEXT DUP FADD 2 FLN FDIV 12 FINV FADD FPI FSQ 6 FDIV 2 FLN FSQ FDIV FADD ZZ\<-\->F NEG SWAP \->STR
DUP SIZE ROT \=/ -51 FC? { "." } { "," } IFTE UNROT { DUP TAIL SWAP HEAD } { "0" } IFTE UNROT + +
SWAP STOF
\>>

EVAL ->

1.000000000001237412575736110228719610646672874297732048196548443844171825640530428850913885586193524976268453340086191658374509030019046729786005370140207590865397221066886209167246612158255597136947833662811711180501522046958297318386956749813586119403326983996836799698362386464361717810944715248515847063950123049027855289479337807074973722174863007602234598952082713436126867407223085711221417206013336683950248036912034243322848607544096465559742710057944068020597818546946376873631661338090760132715563114425400886965240835824220034845681146540332945848091156055661073808986770237768671181359710868112079802546002171398844199048674600407150411381977070159608769770037395721001869135492839448159377839257477067778776337799415286212226231921875049198549974749265675547171167195366657491492695699893916926664962342406045357897998136027548661020448361327035579555228205809418530092189232789163297481121766653027554098532310918458342580878445369891507372744436069036208883146409368525831685839774710
```

(363.97 seconds on the emulator)

B529h, 363 bytes

EMAIL PM FIND QUOTE REPORT

02-10-2019, 09:42 PM **Post: #29**

Gerson W. Barbosa

Senior Member

Posts: 1,361
Joined: Dec 2013

RE: [VA] SRC#002- Almost integers and other beasties

> **Albert Chan Wrote:** (02-10-2019 04:26 PM)
>
> Edit: the difference is the effect of 1/k, iterations ~ 3.322n - log2(3.322n)

Quite good estimation! It allows me to save 10 bytes by replacing **`LN + DUP LN - LASTARG SWAP / + 2 LN`** with **`SWAP DUP2 / LN + SWAP SQ`**

Now # 62Fh, 353 bytes.

EMAIL PM FIND QUOTE REPORT

02-13-2019, 02:49 PM (This post was last modified: 02-13-2019 02:51 PM by Albert Chan.) **Post: #30**

Albert Chan

Senior Member

Posts: 1,226
Joined: Jul 2018

RE: [VA] SRC#002- Almost integers and other beasties

Hi, Gerson W. Barbosa

Your code might be shortened and faster by removing (-1)^k factor.
Since it pre-calculated iterations required, summing backwards may be more accurate.
Example: With 4 digits precision, sum 4 terms,

t = |sum| = 1 - 1/(2*3) + 1/(3*7) - 1/(4*15)

Steps:
t = 0
t = 1/60 - t = 0.01667
t = 1/21 - t = 0.04762 - t = 0.03095
t = 1/6 - t = 0.1667 - t = 0.1358
t = 1 - t = 0.8642

EMAIL PM FIND QUOTE REPORT

02-15-2019, 05:29 PM **Post: #31**

Gerson W. Barbosa
Senior Member

Posts: 1,361
Joined: Dec 2013

RE: [VA] SRC#002- Almost integers and other beasties

> **Albert Chan Wrote:** (02-13-2019 02:49 PM)
> Your code might be shortened and faster by removing (-1)^k factor.
> Since it pre-calculated iterations required, summing backwards may be more accurate.

Sure I am aware of these, but initially I was not interested in optimizing the code for speed, only for size. Anyway, since the summation can be evaluated without resorting to exponentiation, as you've pointed out, there's no reason to use it, at least inside the loop:

```
enter k;
sign := -2*mod(k, 2) + 1;
sum := 0;
a := 2^k - 1;
for i = k to 1 step -1;
  sum := sum + sign/(a*i);
  sign := -sign;
  a := a div 2
next i;
display sum
```

The RPL code, however, is now slightly slower, because either it's not properly optimized yet or the LongFloat FY^X function is very efficient for integer arguments.

```
%%HP: T(3)A(R)F(.);
\<< RCLF SWAP -105 CF -3 CF DUP 1 + 'DIGITS' STO 10 LN * 2 LN SWAP DUP2 / LN + SWAP / CEIL 2 OVER ^
1 - OVER 2 MOD -2 * 1 + SWAP 0 4 ROLL 1
  FOR i OVER i FMULT 4 PICK FMULT FINV FADD ROT NEG ROT 2 FDIV FIP ROT -1
  STEP UNROT DROP2 DUP FADD 2 FLN FDIV 12 FINV FADD FPI FSQ 6 FDIV 2 FLN FSQ FDIV FADD ZZ\<-\->F NEG
SWAP \->STR DUP SIZE ROT \=/ -51 FC? { "." } { "," } IFTE UNROT { DUP TAIL SWAP HEAD } { "0" } IFTE
UNROT + + DUP SIZE " " REPL " " "" SREPL DROP SWAP STOF
\>>
```

1A8h, 428.5 bytes

141.34 seconds on my HP-50g, for 100 (previously, 132.59 seconds)

```
  1 -> 0.9

  2 -> 1.0

  3 -> 0.999

  4 -> 0.9999

  5 -> 0.99998

  6 -> 1.00000

  7 -> 0.9999997

 12 -> 1.00000000000

 20 -> 1.0000000000012374125

 40 -> 1.000000000001237412575736110228719610646
```

```
 50 -> 1.0000000000012374125757361102287196106466728742977
```

```
100 -> 1.000000000001237412575736110228719610646672874297732048196548443844171825640530
428850913885586193525
```

Regards,

Gerson.

EMAIL PM FIND QUOTE REPORT

02-15-2019, 07:50 PM **Post: #32**

Albert Chan
Senior Member

Posts: 1,226
Joined: Jul 2018

RE: [VA] SRC#002- Almost integers and other beasties

> **Gerson W. Barbosa Wrote:** (02-15-2019 05:29 PM)
> ```
> a := 2^k - 1;
> ...
> a := a div 2
> ```

Despite RPL does not see speedup, this is a great optimization !
Can the sign flipping code be removed, and possibly gain some speed ?

Code:
```
>>> sum, k = 0, 35       # for 12 digits accuracy
>>> a = (1<<k) - 1       # k bits of 1
>>> while k > 0:
...     sum = 1/(a*k) - sum
...     a, k = a>>1, k-1
...
>>> print sum
0.868876652659
```

EMAIL PM FIND QUOTE REPORT

02-15-2019, 11:48 PM (This post was last modified: 02-16-2019 02:15 AM by Gerson W. Barbosa.) **Post: #33**

Gerson W. Barbosa
Senior Member

Posts: 1,361
Joined: Dec 2013

RE: [VA] SRC#002- Almost integers and other beasties

> **Albert Chan Wrote:** (02-15-2019 07:50 PM)
> Can the sign flipping code be removed, and possibly gain some speed ?
>
> **Code:**
> ```
> >>> sum, k = 0, 35       # for 12 digits accuracy
> >>> a = (1<<k) - 1       # k bits of 1
> >>> while k > 0:
> ...     sum = 1/(a*k) - sum
> ...     a, k = a>>1, k-1
> ...
> >>> print sum
> 0.868876652659
> ```

Oh, I see!

```
%%HP: T(3)A(R)F(.);
\<< RCLF SWAP -105 CF -3 CF DUP 1 + 'DIGITS' STO 10 LN * 2 LN SWAP DUP2 / LN + SWAP / CEIL 2 OVER ^
1 - 0 ROT 1
  FOR i OVER i FMULT FINV SWAP FSUB SWAP 2 FDIV FIP SWAP -1
  STEP FSUB DUP FADD 2 FLN FDIV 12 FINV FADD FPI FSQ 6 FDIV 2 FLN FSQ FDIV FADD ZZ\<-\->F NEG SWAP
\->STR DUP SIZE ROT \=/ -51 FC? { "." } { "," } IFTE UNROT { DUP TAIL SWAP HEAD } { "0" } IFTE UNROT
+ + DUP SIZE " " REPL " " "" SREPL DROP SWAP STOF
\>>
```

10B6h, 393.5 bytes

Now 129.59 seconds for 100 digits.

P.S.: Replaced NIP FNEG with FSUB at loop exit. I'd imagined the former were faster, but it turns our it's actually 0.05 slower (although this is not conclusive after one measurement only).

EMAIL PM FIND QUOTE REPORT

02-23-2019, 05:58 AM **Post: #34**

ttw
Member

Posts: 215
Joined: Jun 2014

RE: [VA] SRC#002- Almost integers and other beasties

For things like (-1)^k in a list with k running over a large range, one can process by unrolling the loop twice. There needs to be either cleanup at the end or pre compute the first step for odd range.

EMAIL PM FIND QUOTE REPORT

User(s) browsing this thread: Valentin Albillo*