♦MoHPC♠   *The Museum of HP Calculators*

HP Forum Archive 18

[Return to Index | Top of Index]

Lambert's W on the HP-33s

*Message #1 Posted by **Gerson W. Barbosa** on 10 Feb 2008, 9:16 p.m.*

```
W0001 LBL W
W0002 x<>y
W0003 STO A
W0004 x<>y
W0005 STO B
W0006 1
W0007 +
W0008 LN
W0009 STO X
W0010 FN= F
W0011 SOLVE X
W0012 RCL A
W0013 x<>y
W0014 RTN
F0001 LBL F
F0002 RCL X
F0003 eˣ
F0004 LASTx
F0005 *
F0006 RCL- B
F0007 RTN


W: LN=54; CK=C232
F: LN=21; CK=3529


-e⁻¹ <= x < 1E500
```

This basically uses the definition of Lambert's W function: W is the inverse function of $f(w)=we^{w}$. Its applications include solutions of equations involving exponentials. For instance, the solution of $x^{x}=2$ is $x=e^{W ln(2)}$

On the HP-33s:

```
2 LN XEQ W eˣ -> 1.55961046946
```

The program is equivalent to Valentin Albillo's one-liner for the HP-71B in HP-71B Math ROM Baker's Dozen (Vol. 1), but the HP-33s is slightly faster. I am not sure about the HP-35s.

Gerson.

Re: Lambert's W on the HP-33s

Message #2 Posted by John B. Smitherman on 10 Feb 2008, 10:05 p.m.,
in response to message #1 by Gerson W. Barbosa

Hi Gerson. It's about the same - 3 seconds on the 33s and 35s. Try moving between the 32sii, 33s and 35s. Maybe it's just me but I find it difficult!

Regards,

John

Re: Lambert's W on the HP-33s

Message #3 Posted by Gerson W. Barbosa on 11 Feb 2008, 6:33 a.m.,
in response to message #2 by John B. Smitherman

Hello John,

Thanks for testing it on the HP-35s.

> Quote:
> ———
> Try moving between the 32sii, 33s and 35s. Maybe it's just me but I find it difficult!
> ———

Switching from the HP-33s to the HP-50g is difficult as well. That's because the different placement of the ENTER key on both calculators. There was a time HP calculators shared a similar keyboard layout.

Regards,

Gerson.

Re: Lambert's W on the HP-33s

Message #4 Posted by John Keith on 10 Feb 2008, 10:34 p.m.,
in response to message #1 by Gerson W. Barbosa

My similar program takes about 0.5s on the HP 50g. Would be great to have it as a built-in function, though. Probably execute 10-20 x faster.

John

Re: Lambert's W on the HP-33s

Message #5 Posted by Gerson W. Barbosa on 11 Feb 2008, 6:17 a.m.,
in response to message #4 by John Keith

Yes, it could be placed under MTH SPECI(al functions). It seems most implementations are based on Newton's method with optimized initial guesses. Would you mind posting your HP-50g solution? Thanks!

Gerson.

Re: Lambert's W on the HP-33s

Message #6 Posted by Egan Ford on 12 Feb 2008, 12:57 a.m.,
in response to message #5 by Gerson W. Barbosa

Hello Gerson,

I couldn't resist. Some super fast missing special functions for the 50g: http://sense.net/~egan/hpgcc/misc/special.zip

Unzip that to the root of your 50g SD card. The following files will be extracted:

```
SPECIAL.LIB
SPECIAL/LAMBERTW
SPECIAL/LOGGAMMA
```

Put the SD card back in your 50g and type:

```
:3:SPECIAL.LIB
RCL
2
```

```
STO
ON-C
```

This library adds two missing special functions to the 50g. LogGamma and LambertW. Both take real or complex arguments. There is a 2nd LambertW (LAMBERTWB) that takes two arguments. The first is the branch (integer) and the 2nd is the argument.

LogGamma tests:

```
  z             LogGamma(z)      50g loggamma(z)   50g ln(gamma(z))


  1                    0                    0                  0
 100         359.134205370        359.13420537       359.13420537
1000         5905.22042321        5905.22042321      1151.2925465
5555.4444         42343.2         42343.1698379      1151.2925465
12+34i   -11.7232+102.052i  (-11.7232,102.0521)  (-11.7232,1.5212)
```

Bold answers are wrong. ln(gamma()) is not loggamma().

LambertW tests:

```
 b           z     function                 result


             5     LAMBERTW                1.32672
 0           5     LAMBERTWB               1.32672
-1           5     LAMBERTWB       (0.05663,-4.72438)
 3           5     LAMBERTWB       (-1.23846,17.20691)
            -5     LAMBERTW        (0.84484,1.97501)
    123.4+5.6i     LAMBERTW        (3.54958,0.03538)
    -898-44.5i     LAMBERTW        (5.06143,-2.61517)
```

And

```
2 LN LAMBERTW e^x = 1.55961046946
```

Neither is well tested for stability. A spot check of many values looked ok. I plan to clean up when HPGCC3 releases. For now you are stuck with the HPGCC2 version.

Edited: 12 Feb 2008, 2:01 a.m.

Re: Lambert's W on the HP-33s

Message #7 Posted by Gerson W. Barbosa on 12 Feb 2008, 6:29 a.m.,

in response to message #6 by Egan Ford

Hello Egan,

Thanks! They are really fast. The largest real argument to LAMBERTW appears to be around 1e154. LAMBERTW(1e155) returns 351.023231852 instead of 351.039789836. Larger real arguments return (0., 0.). This is not a problem, as long as the valid ranges are specified.

Keep up the excellent work! Looking forward to the next version.

Gerson.

Re: Lambert's W on the HP-33s

Message #8 Posted by Egan Ford on 12 Feb 2008, 11:20 a.m.,
in response to message #7 by Gerson W. Barbosa

That is odd, the range should have been closer to +/-1e308. That is the limit of IEEE 64-bit floats. The same code compiled on my workstation has a range of +/-1e306. For now I'll state the range as +/-1e154. IMHO, good enough for most. Not every 50g function can take 1e500 size arguments. E.g. Gamma.

Edited: 12 Feb 2008, 11:27 a.m.

Re: Lambert's W on the HP-33s

Message #9 Posted by Gerson W. Barbosa on 12 Feb 2008, 6:12 p.m.,
in response to message #8 by Egan Ford

> Quote:
>
> For now I'll state the range as +/-1e154. IMHO, good enough for most.

You're quite right, even more considering many times the argument is the natural logarithm of something. I can't imagine a number whose log is 1e154 :-)

Regards,

Gerson.

Re: Lambert's W on the HP-33s

Message #10 Posted by Egan Ford on 12 Feb 2008, 8:51 p.m.,
in response to message #9 by Gerson W. Barbosa

I found the problem. It is no coincidence that +/-1e154 is the R range for my Lambert W function: 154 + 154 = 308. LAMBERTW and LAMBERTWB support complex arguments. Since I was lazy and short on time I converted reals to complex. I followed the code to a complex division between two large numbers > 1e154. This complex division requires a multiplication that exceeds 1e308 (The limit of 64bit floats). My code does not check for NAN or INF (it should), so 0s are return incorrectly. BTW, my workstation did not have this problem. I suspect that my math library uses long doubles internally to multiple large doubles.

Options:

- Find a small arbitrary precision library with complex number support.
- Write a different routine for reals. This will limit complex numbers to +/-1e154 and reals to 1e308.
- Do nothing. +/-1e154 is good enough. Fix bugs to return error when exceeding c9x-complex library limits.
- Do nothing.

I must admit I had never heard of this function. None of my textbooks mention W. I usually shop for algorithms in NR--nothing.

Late last night I only lost about an hour of sleep to write and verify LAMBERTW. But, I lost another 2 hours of sleep being fascinated with W! Thanks for the entertainment.

A couple good reads:

High level: http://www.americanscientist.org/content/AMSCI/AMSCI/ArticleAltFormat/200521714030_306.pdf

Details with examples: http://www.cs.uwaterloo.ca/research/tr/1993/03/W.pdf

Re: Lambert's W on the HP-33s

Message #11 Posted by Egan Ford on 14 Feb 2008, 5:30 p.m.,
in response to message #5 by Gerson W. Barbosa

Gerson,

Here is an RPL real-number only version for the 50g:

```
<< NEG 'x*EXP(x)' + 'x' 0 ROOT 'x' PURGE >>
```

Re: Lambert's W on the HP-33s

Message #12 Posted by Gerson W. Barbosa on 14 Feb 2008, 6:43 p.m.,
in response to message #11 by Egan Ford

And it works for x=9.9999999999e499, in case I ever need it :-)

Thanks!

Gerson.

Re: Lambert's W on the HP-33s

Message #13 Posted by Valentin Albillo on 12 Feb 2008, 6:46 a.m.,
in response to message #1 by Gerson W. Barbosa

Hi, Gerson:

I'm truly glad you're interested in Lambert's W function, as it really is a pet function o'mine ! :-)

After reading tons about it for the past few years, I'm pretty much convinced that indeed it's a firm candidate for the next "elementary" transcendental function to be added to the classic trigonometric, exponential, and logarithmic ones (which, essentially, can all be reduced to exponentials and inverse exponentials (i.e. logarithms) of arbitrary complex values).

Why should it be considered elementary ? Well, a number of reasons:

- It has many interesting mathematical properties, such as the fact that both its integral and its derivative are expressible in closed form in terms of itself, not to mention the fact that its inverse is also closed-form, namely w*exp(w) !

- It can be used to solve a vast number of important exponential and logarithmic equations in closed form in terms of itself, such as the ones featured in my article and many others.

- Same applies to a number of important differential and integral equations, which can be solved in closed form in terms of Lambert's W and actually do appear in real-world applications.

- Practical applications for it in everyday's engineering and technical life (and even in the arts, demographics, military, etc) are being discovered all the time. You can find dozens of references in the Web, but as an example of what uses it can be put to, have a look at these three interesting PDF documents:

A new elementary function for our curricula ?

Exact expressions for the range and the optimal angle of a projectile with linear drag

Where to hang a painting ?

If it continues to grow in importance, which seems pretty likely at this rate, I think it's not at all preposterous to contemplate the possibility that standard software (such as Excel) will eventually include it along with the trigonometric and exponential functions (math packages such as Mathematica and Maple already include it to arbitrary precision and for arbitrary complex values).

After that, the next logical step would be for advanced scientific calculators to include it as well, most specially since it's so very easy to compute and its inverse is already immediately available.

In any case, if I were to develop some kind of calculator software for a PC or a Palm or similar, I would certainly include Lambert's W (called simply "W(x)") as an standard function.

Note:

Of course, many important special functions are also used in engineering and technical applications all the time (Bessel & elliptic functions in electronics, for instance) but they aren't considered elementary nor will you find any calculator with a key reserved for them.

On the other hand, the Gamma function is indeed extremely important in applications and appears very frequently as well, so it's given its own key or menu entry in advanced scientific calculators from the HP-34C onwards and previously in some other brand's models as well. It's not considered "elementary" either but "transcendental", because it can be proven that it does not satisfy any linear differential equation with algebraic coefficients.

Best regards from V.

Edited: 12 Feb 2008, 6:53 a.m.

Re: Lambert's W on the HP-33s

Message #14 Posted by Gerson W. Barbosa on 12 Feb 2008, 9:07 a.m.,
in response to message #13 by Valentin Albillo

Hello Valentin,

Thanks for your always interesting comments. Thanks also for the links, especially the first one which is easier to understand.

A reason Excel should include this function should be because apparently it is not so difficult. It can be implemented even on the HP-12C, at least for the real values, as you can se below. I took the liberty of using Sir Isaac Newton's ingenious solver :-) Time Voyager (page 9)

I have to leave for work in about half an hour, so sorry for being rather brief.

Best regards,

Gerson.

```
01 x<>y
02 STO 3
03 x<>y
04 STO 4
05 1
06 +
07 LN
08 STO 0
09 CLx
10 STO 1
11 RCL 0
12 GTO 42
13 x=0
14 GTO 31
15 RCL 1
16 x=0
17 GTO 34
18 /
19 1
20 -
21 STO/ 2
22 RCL 0
23 RCL 0
24 RCL 2
25 -
26 STO 0
27 -
28 x=0
29 GTO 31
30 GTO 09
31 RCL 3
32 RCL 0
33 GTO 00
34 x<>y
35 STO 1
36 RCL 0
37 EEX
38 5
39 CHS
40 STO 2
```

```
41 +
42 e^x
43 LSTx
44 *
45 RCL 4
46 -
47 GTO 13


2 R/S  ->  0.852605502 (after 15 seconds)
```

Edited: 12 Feb 2008, 9:11 a.m.

Re: Lambert's W on the HP-33s

Message #15 Posted by John B. Smitherman on 12 Feb 2008, 7:22 p.m.,
in response to message #13 by Valentin Albillo

Another Lambert W-function reference:

http://mathworld.wolfram.com/LambertW-Function.html

Regards,

John

Re: Lambert's W on the HP-33s

Message #16 Posted by Egan Ford on 12 Feb 2008, 8:26 p.m.,
in response to message #15 by John B. Smitherman

Some history:

http://www.americanscientist.org/template/AssetDetail/assetid/40804?&print=yes

---

Re: Lambert's W on the HP-33s

Message #17 Posted by Werner on 14 Feb 2008, 1:59 a.m.,
in response to message #1 by Gerson W. Barbosa

Here's a 42S version, using but a single alpha label:

```
00 { 36-Byte Prgm }
01*LBL "W"
02 FC? 45
03 GTO 00
04 RCL "X"
05 E^X
06 LASTX
07 *
08 RCL- "W"
09 RTN
10*LBL 00
11 PGMSLV "W"
12 STO "W"
13 CLX
14 STO "X"
15 10
16 SOLVE "X"
17 END
```

Cheers, Werner

Re: Lambert's W on the HP-33s

Message #18 Posted by Gerson W. Barbosa on 14 Feb 2008, 6:50 p.m.,
in response to message #17 by Werner

Thanks for providing an HP-42S version!

Cheers,

Gerson.

[Return to Index | Top of Index]

GTO Go back to the main exhibit hall