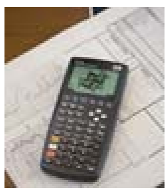
HP Solve

Calculating solutions powered by HP

» **[From the Editor]**
Learn more about current articles and feedback from the latest Solve newsletter including RPN tips, One Minute Marvels and Math problem challenges.

Your articles



» **HP Algebraic**
Palmer Hanson

What is an algebraic calculator? How do HP algebraic calculators compare with other calculators – RPN and Algebraic? Learn insights and answers in Palmer's latest article.

» **Better Problem Solving–Part I**
Richard J. Nelson

This is the first part in a series that will review how we solve problems with our calculators and how we might change the method or process to solve them more efficiently.

» **PROOT: A Blast From the Past!**
Namir Shammas

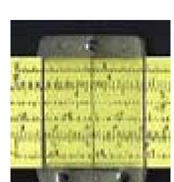
Finding the roots of an equation is one of the most difficult mathematical challenges. Namir has been studying HP's methods for many years and provides his valuable insight into this topic.

» **Commas in the HP Calculator Display**
Richard J. Nelson

In Richard's latest article, he explains the basic purposes of the comma delimiter in the display of an HP Calculator.

» **HP's Calculator Manuals**
Richard J. Nelson

Here is a review of nearly four



» **Fundamentals of Applied Math Series #5**
Richard J. Nelson

Issue 22
January 2011

Welcome to the twenty-second edition of the HP Solve newsletter. Learn calculation concepts, get advice to help you succeed in the office or the classroom, and be the first to find out about new HP calculating solutions and special offers.

» **Download the PDF version** of newsletter articles.

» **Contact the editor**

decades of HP calculator manuals with a personal list of desired topics for their content. A comparison of the most complete User's manual ever written for an HP calculator is made to illustrate the resource issue involved.

The fifth installment explains and explores all of the important applications of logs. It is also the first serious calculator function to be reviewed in this series with the preceding installments dedicated to numbers and pre-calculator problem solving.

From the Editor

HP Solve #22 page 3

Article – Next →

From The Editor – Issue 22

Winter is here in the northern hemisphere and many ***HP Solve*** readers are spending more time indoors because very cold weather is at hand - even here in the Sonoran Desert. The Consumer Electronics Show is January 6 – 9, 2011in Las Vegas. I will be attending and if you will be there let me know. Perhaps we can have Saturday night dinner together as a group of HP users (like we did last year?).

Here is the content of this issue.

S01 Regular Columns

This is a collection of repeating/regular columns.

- **From the editor.** This column provides feedback and commentary from the editor.
- **RPN Tip 22.** Here is an oldie but goodie reprint that explains the basics of RPN.
- **One Minute Marvels.** This OMM reverses the objects on the stack. It is short and fast and illustrates some of the criteria used to select OMM's.
- **Math problem challenge # 3.** Here is an equivalent resistance problem that is really very easy to solve, IF, you use the right approach. You must think "out of the box" on this one.

S02 - HP Algebraic by Palmer Hanson What is an algebraic calculator? How do HP algebraic calculators compare with other calculators – RPN and Algebraic? Palmer is very well known in the "algebraic" calculator community is one of those few individuals who are able to easily move between the two user interfaces. Personally I have to make my brain work "extra hard" when using an algebraic calculator.

S03 – Better Problem Solving Part I The scientific handheld calculator is 38 years old. Many of the basic functions haven't changed very much since the HP-35A started this product category in 1972. Technology has changed especially in terms of display quality and running speed. Convergence is ever lurking in terms of competition for the personal handheld calculator. The primary advantage, in my opinion, of the handheld scientific (or graphing) calculator is convenience and cost. Otherwise a calculator in your smart phone, laptop, or desktop computer will suffice. Reader feedback for some of these "strange" suggestions is solicited.

S04 PROOT: A Blast From the Past! by Namir Shammas Finding the roots of an equation is one of the most difficult mathematical challenges. Namir has been studying HP's methods for many years and he provides his valuable insight into this topic.

S05 Commas in the HP Calculator Display I was reading one of my electronically emailed newsletters and I was reminded of the changes in the way various HP calculators display numbers with commas. I consulted one of the experts in RPL, Dr. William Wickes, for some "official" insight into this topic. Personally I miss having numbers displayed like those in an HP-12C or HP-15C, or HP-41C. This is especially important to me because my favorite machine is the HP48GX.

S06 HP's Calculator Manuals Users of technical instruments need a "User's Manual" in order to best apply the many complex and not-always-obvious features of a product. Here is a review of the nearly four decades of HP calculator manuals with a personal list of desired topics for their content. A comparison of the most complete User's manual ever written for an HP calculator is made to illustrate the resource issue involved.

S07 #5 in the Fundamentals of Applied Math Series – Logs We all know what logs are but if you are like me you may not remember all of the important applications of logs. This is the first serious calculator function to be reviewed in this series with the preceding four installments dedicated to numbers and pre-calculator problem solving.

That is it for this issue. I hope you enjoy it. If not, tell me!

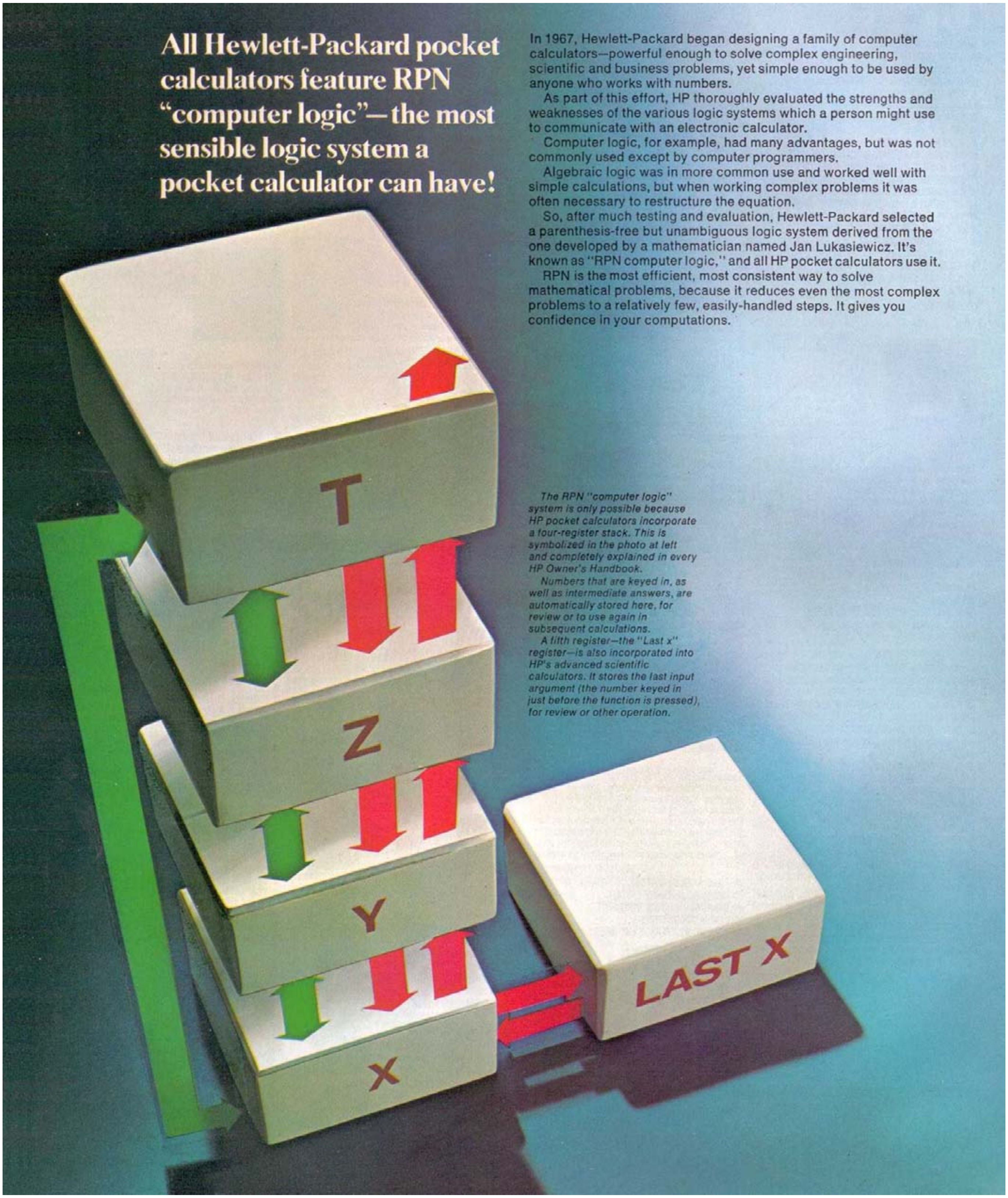
Also tell me what you liked, and what you would like to read about.

X < > Y,

Richard Email me at: hpsolve@hp.com

RPN Tip #22 – Basic Review

There are still a few ideas left to illustrate how RPN works so we will have a few more RPN tips. Here is a reprint of an article that was published in The Hewlett-Packard Calculator Catalog and Buying Guide, Spring, 1975. As with all things RPN it is timeless.



All Hewlett-Packard pocket calculators feature RPN "computer logic"—the most sensible logic system a pocket calculator can have!

In 1967, Hewlett-Packard began designing a family of computer calculators—powerful enough to solve complex engineering, scientific and business problems, yet simple enough to be used by anyone who works with numbers.

As part of this effort, HP thoroughly evaluated the strengths and weaknesses of the various logic systems which a person might use to communicate with an electronic calculator.

Computer logic, for example, had many advantages, but was not commonly used except by computer programmers.

Algebraic logic was in more common use and worked well with simple calculations, but when working complex problems it was often necessary to restructure the equation.

So, after much testing and evaluation, Hewlett-Packard selected a parenthesis-free but unambiguous logic system derived from the one developed by a mathematician named Jan Lukasiewicz. It's known as "RPN computer logic," and all HP pocket calculators use it.

RPN is the most efficient, most consistent way to solve mathematical problems, because it reduces even the most complex problems to a relatively few, easily-handled steps. It gives you confidence in your computations.

The RPN "computer logic" system is only possible because HP pocket calculators incorporate a four-register stack. This is symbolized in the photo at left and completely explained in every HP Owner's Handbook.

Numbers that are keyed in, as well as intermediate answers, are automatically stored here, for review or to use again in subsequent calculations.

A fifth register—the "Last x" register—is also incorporated into HP's advanced scientific calculators. It stores the last input argument (the number keyed in just before the function is pressed), for review or other operation.

No "=" key is needed

If you will look at the keyboard of any HP calculator shown in this catalog, you will see that none has an "=" key. Nor are there any keys for parentheses. None are needed.

Instead, all HP calculators have a key like this:

[ENTER↑]

Thanks to this key, and RPN logic, you get four major advantages you don't get with most other calculators:

1. **You work with only two numbers at a time, just as if you were solving the problem with paper and pencil. (Only incredibly faster.)**

Even the most complex problems are broken down into a series of easily-handled two-number problems, which you can solve in any order that's convenient—left to right, right to left, or from the middle of the equation outwards.

No matter what kind of problem it is, there's no restructuring to do . . . no rearranging of the equation as is so often necessary with other calculators, to conform to algebraic logic.

So there's less confusion and less chance for error.

2. **The function is immediately calculated.**

With an HP calculator, pressing the function key initiates the desired action, so you get your answer immediately.

For example, to find the square root of 16, simply press three keys . . .

[1] [6] [$\sqrt{x}$]

. . . and your answer immediately appears on the display . . .

4.00

And it's just as fast and easy to calculate squares, cosines, factorials or other functions.

3. **The intermediate answer is displayed.**

This enables you to check your calculation every step of the way . . . so you can do something about it if it doesn't look "right."

4. **The intermediate answer is automatically stored.**

So there's no need to store it manually, by keying in each digit, if the number is needed in the next calculation.

Obviously, this saves keystrokes and helps prevent errors. And you can easily recall the intermediate answer if need be.

Four major advantages — to give you *confidence in your computations.*

Just four simple steps

To use any Hewlett-Packard pocket calculator, just follow these four simple steps . . .

1. Key in the first number.
2. Enter it into the stack (press the "ENTER↑" key).
3. Key in the second number.
4. Press the function key.

And if your numbers are already stored in the calculator as intermediate answers, all you have to do is hit the function key.

Could anything be easier . . . or faster?

Here's an example

Let's take a simple problem—2.5 x 4—and solve it with an HP calculator, using the four steps shown above . . .

1. Key in the first number:

[2] [·] [5]

2. Enter it into the stack:

[ENTER↑]

3. Key in the second number:

[4]

4. Press the function key:

[x]

Your answer appears on the display:

10.00

Now let's try a slightly more difficult problem: (2 + 6) x (9 — 3.5).

If you were working this out with paper and pencil, you'd probably work from left to right and first solve for (2 + 6). Then you'd solve for (9 — 3.5). Finally, you'd multiply the two answers—8 x 5.5—and get 44.

Well, with an HP pocket calculator, you work the problem the same way.

(Or, if you prefer, you could work it right to left, or even—with more complex problems—from the middle outwards).

Working from left to right, press . . .

[2] [ENTER↑] [6] [+]

The display shows the intermediate answer:

8.00

To solve for (9 — 3.5), press . . .

[9] [ENTER↑]
[3] [·] [5] [−]

The display shows:

5.50

To multiply the two intermediate answers (which have been automatically stored), press . . .

[x]

And the displays shows:

44.00

Even if your problem were as complex as converting indicated air speed to the true mach number . . .

$$\sqrt{5\left(\left[\left(\left[\left(\frac{400}{661.5}\right)^2 .2+1\right]^{\frac{1.4}{.4}}-1\right)\frac{29.96}{15}+1\right]^{.286}-1\right)}$$

. . . you would still be able to solve it quickly, easily and without confusion if you used an HP calculator, thanks to RPN—the most sensible logic system a pocket calculator can have!

Have you been reading RPN Tips? Are you very familiar with RPN? Which of the statements below do not apply to RPN?

A. RPN is easy to use because your approach to every problem is always the same.
B. RPN solves problems with the minimum of "rules" to remember.
C. RPN does not require parentheses to solve problems, e.g. 1 + 2 x 3 =
D. RPN is more keystroke efficient than Algebraic, i. e. it requires fewer keystrokes to solve problems.

Answer: C because the problem is unclear. Algebraic assumes the multiplication is performed first.

HP 48 One Minute Marvel – No. 9 – Stack Reversal

One Minute Marvels, OMMs, are short, efficient, unusual, and fun HP 48 programs that may be entered into your machine in a minute or less. These programs were developed on the HP 48, but they will usually run on the HP 49 and HP 50 as well. Note the HP48 byte count is for the program only.

Suppose you have 100 objects (e.g. numbers) on the stack and you want to take a quick look at the top few? If you use the normal stack manipulation commands you could do this, but there are other ways that are far more efficient.

You could assemble the numbers into a list and use the list reverse command to see the objects more easily in the display OR you could just reverse the objects in place using **'SREV'**. This means the top four in a normal HP48 display and more if you are using an HP50g with a small font.

The utility (efficiency) of this OMM should be obvious but its primary utility here is to illustrate a fast and efficient technique that is worth studying. Here is how the routine works.

The first command, DEPTH, returns the number of values on the stack. If there are 100 objects on the stack level one would be 100. The number two is then placed on the stack because we want the loop to ROLL the stack from 2 to 100 times. SWAP orders the input for the correct input for the FOR loop as 2 through 100.

j is a local variable that is used only for the duration of the FOR NEXT loop. The first value of j is 2 and it is consumed. The second call of j returns 2 to the stack and ROLL is executed. j is incremented to 3 with the execution of NEXT. This repeats until the stack is rolled down 100 times and the program terminates.

Is this the only way to accomplish this task? Would more efficient methods be faster or use fewer bytes? Note that the timing information reverses the digits 1 to 100 in about 3/4ths of a second. You couldn't press a normal stack sequence of commands faster than this. Could the 50g do this with fewer commands?

What if you used a program that put the objects into a list, used the REVLIST command, and then exploded the list back onto the stack? Would this be a shorter program? Would it work as well as **'SREV'**? As a test include a program as an object on the stack. What happens then? If you understand the answers to these questions you will know why **'SREV'** is an OMM.

'SREV' << DEPTH 2 SWAP FOR **j j** ROLL NEXT >>

8 commands, 34.0 Bytes, #3115h. Timing: 1 to 100 in 0.767 seconds.

Note: If you don't get the same HP48 check sum verify that "j" is lower case and not upper case "J."

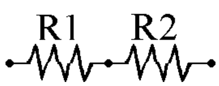
HP Solve Math Problem Challenge #3

This problem is the third in a series of real world practical or teaching problems offered as a challenge to ***HP Solve*** readers. Send your solution to the editor and if your solution is thought to be the most practical, clear, and using minimal math, it will be published in ***HP Solve***.

This problem is an equivalent resistance problem. The math involved is not very complex, but if you have any basic electrical experience you will understand Ohms law and equivalent resistance.

Here is the equivalent resistance idea.

Series Connected

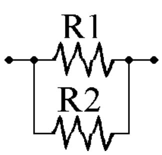


$R_T = R1 + R2$

$R_T = R1 + R2 + R3$

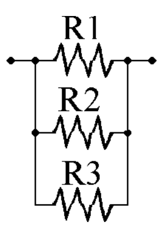
Two resistors in series, R1 & R2 have an equivalent resistance, $R_T = R1 + R2$. This means that the resistor may be replace with a single resistor of the value of R_T. If R1 = 13 Ω, R2 = 12 Ω, R3 = 25 Ω, R_T = 50 Ω. The current is the same through all series resistors.

Parallel Connected



$$RT = \frac{R1 * R2}{R1 + R2}$$

Two resistors in parallel, R1 & R2, have an equivalent resistance, $R_T = R1*R2/(R1 + R2)$. This is their product over their sum. If R1 = 13 Ω, R2 = 12 Ω, RT = 6.23 Ω.



$$RT = \frac{1}{\frac{1}{R1} + \frac{1}{R2} + \frac{1}{R3}}$$

Three resistors in parallel, R1, R2, & R3, have an equivalent resistance of the reciprocal of the sums of the reciprocals of each resistance.

Here is an example Problem. See Fig. 1. Calculate the equivalent resistance of the following points.

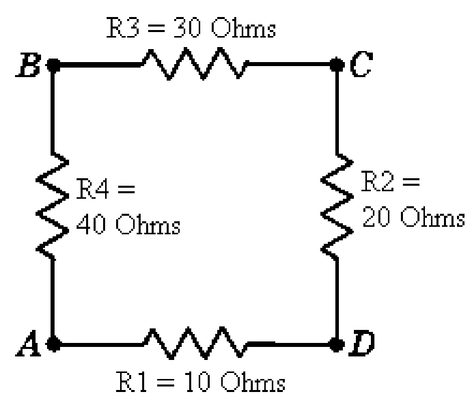


Fig. 1 – Example series parallel problem.

<u>A to B:</u> This is a 40-ohm resistor, R4, with three series resistors, R3, R2, R1 in parallel. The series equivalent is their sum or 60 ohms = 30 + 20 + 10. 60 ohms in parallel with 40 ohms is 24 ohms.

<u>A to C:</u> This is two sets of two series connected resistors (R3 & R4), (R1 & R2) connected in parallel. These equivalent resistors are 70 ohms and 30 ohms in parallel. The effective resistance is 21 ohms.

<u>A to D:</u> This is a 10-ohm resistor connected in parallel with three series connected resistors, R2, R3, & R4. The series equivalent is 20 + 30 + 40 = 90 ohms. The equivalent of 90 ohms in parallel with 10 ohms is 9 ohms.

B to C: This is a 30-ohm resistor, R3, connected in parallel to three series connected resistors, R2, R1, &

R4. The equivalent of the series resistors is their sum or 80 ohms. 80 ohms in parallel with 20 ohms is 21 ohms.

B to D: This is two series connected sets (R2 & R3) and (R1 & R4) in parallel. This is the equivalent of 50 ohms in parallel with 50 ohms which is 25 ohms.

C to D: This is a 20-ohm resistor, R2, connected in parallel to three series connected resistors, R1, R2, & R4. The equivalent of the series resistors is their sum or 70 ohms. 70 ohms in parallel with 20 ohms is 16 ohms.

Now that you have the basics of equivalent resistance here is the problem. Twelve equal resistors are connected together as if they were the edges of a cube. The equivalent resistance to be calculated is across the internal corners of the cube, i.e. from a to g in Fig. 2. The value of each resistor is 2,982 ohms.

The best solution idea is one that gives a clear step-by-step explanation of the solution. The best solution is based on the guidelines listed below

1. The printed solution is the decision of the editor and multiple solutions may be published.

The description and clarity of the solution is most important.

3. The use of graphics, if needed, should be used to make understanding the solution easier and clearer.
4. The use of minimal mathematics, i.e. algebra instead of calculus.

5. The use of an HP Calculator if helpful.

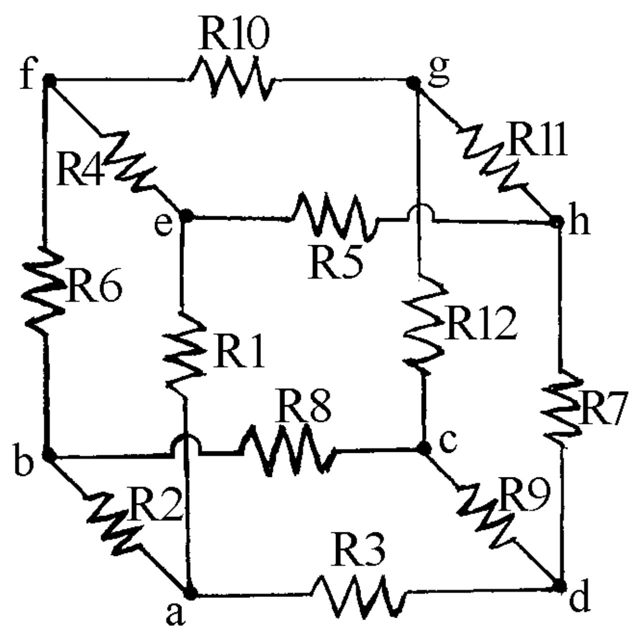


Fig. 2 – Resistors along the edges of a cube. Find the Equivalent resistance from a to g across the internal diagonal of the cube.

Extra "points" are possible if multiple solutions are provided, or if derivations of the solution equation/ ratio are provided.

Remember, just getting the answer is not enough to distinguish your result from everyone else. Send your entry to the Editor. The dead line is before the next issue is posted. There is a long publishing lead time so send in your solution as soon as possible.

Send you email solution to the editor, Richard J. Nelson at hpsolve@hp.com

A complete solution will be provided in this column in ***HP Solve*** Issue 23.

HP Algebraic

HP Solve #22 page 10

← Previous **Article** – Next →

HP Algebraic

Palmer Hanson

Ed. Note. I have seen many different calculator "logic systems" since the first HP RPN calculator started the product category we call a scientific (or financial) calculator. Each new model attempts to address a particular use in the market place. One observation I can make is simply that no machine is "pure" anything. All calculator designs are practical tradeoffs of features, functions, logic systems, and applications[0].

What does Algebraic Mean?

HP uses the term algebraic in a very generic way in the definition of operating modes -- if it isn't RPN then it is algebraic unless it is RPL. Thus, the typical options for the operating mode as offered in a menu on a non-RPL HP calculator will be RPN or ALG. A user will find that calling the ALG option can yield one of several different operating modes depending upon which machine is being used. For example, there is a "true algebraic" mode in which equations can be entered in a format very similar to the way they appear in a textbook, an "almost algebraic" mode which is very similar to that of the early machines manufactured by Texas Instruments, and an "adding machine arithmetic" mode which is characterized by a lack of operator precedence similar to that offered by old mechanical calculators..

True Algebraic

The algebraic mode in machines such as the HP-10s and HP-35s operate in a manner which is consistent with the time-honored "My Dear Aunt Sally"[1] form of precedence. Parentheses are allowed and even encouraged if needed for clarity. There is no intermediate output during equation entry. I call this mode true algebraic because an algebraic equation from a textbook can typically be entered directly step by step working from left to right. Richard Nelson prefers to call this Command Line Interface (CLI)[2]. Whatever the nomenclature this mode is very similar to the old higher order languages such as FORTRAN or BASIC, and to the equation entry system in graphical calculators manufactured by Texas Instruments and Casio. A true algebraic mode is also available as an equation entry mode with an HP-33s and as the equation entry for the Solve modes in machines such as the HP-17BII and HP-19BII. True algebraic will evaluate

2 + 3 x 5 =

as 17 because the multiply sign takes precedence over the plus sign. If the user wants to perform the addition before the multiplication then appropriate parentheses may be inserted so that

(2 + 3) x 5 =

will be evaluated as 25. Now consider another equation.

5 + 3^2 =

which will be evaluated as 14 because the exponentiation takes precedence over the plus sign.

Almost Algebraic

The algebraic mode in the HP-33s uses arithmetical precedence and accepts parentheses but operators such as the trigonometric functions, the square root, and the logarithmic functions are entered after the operand rather than before. Thus, I don't consider it to be a true algebraic system because a textbook equation typically cannot be entered directly from left to right. The mode is very similar to that which was used in TI scientific calculators from the TI-30 through the TI-95, but with important differences. With the TI machines and with the algebraic mode in the HP95LX all the user sees during the solution is the calculated result at each intermediate stage of the solution. With the HP-33s the sequence of steps is accumulated in the upper register and the calculated result at the current stage of the solution appears in the lower register. The HP-33s also offers implied multiplication. However, the implementation of the entry of parentheses as shifted functions is a substantial deterrent to the use of the machine in algebraic mode. This partly algebraic mode will evaluate the two examples 2 + 3 x 5 = and 5 + 3^2 = in the same manner as a "true algebraic" machine since the rules of precedence are similar.

Adding Machine Arithmetic

The algebraic modes in business-oriented machines such as the HP-10B, HP-17BII, and HP-19BII do not use arithmetical precedence. Since these machines perform arithmetic in a manner similar to the old mechanical adding machines I call this mode Adding Machine Arithmetic (AMA). Richard Nelson prefers to call it ATH[(2)]. Others call it chain algebraic. AMA machines evaluate 2 + 3 x 5 = ? as 25 because the arithmetic is performed in the order entered. If the user wants to perform the multiplication before the addition appropriate parentheses may be inserted so that 2 + (3 x 5) = ? will be evaluated as 17 or the equation can be changed to 3 x 5 + 2 = . which will be evaluated as 17. Now consider the equation 5 + 3^2 = ? which will be evaluated as 64 because AMA treats the exponentiation operator in the same manner as the arithmetic operators. Thus, when the exponential operator is entered the 3 is added to the 5 yielding 8 which is then squared. To obtain the sum of 5 and 3^2 the user can enter appropriate parentheses 5 + (3^2) = ? which will be evaluated as 14, or you can alter the equation to 3^2 + 5 = ? which will also be evaluated as 14. TI business oriented machines such as the Business Analysts and Money Manager and the Sharp EL-733A operate in the same manner. This effect is not illustrated in the owner's manual for the HP-19BII or Sharp EL-733A. It is illustrated in the manuals for the TI machines.

The Mach Number Equation

For further demonstration of the relative capabilities of the various modes of calculation I will look at the venerable old Mach number equation which appears in many of the manuals for the earlier HP RPN machines.

$$M = \sqrt{5\left[\left(\left\{\left[\left(1 + 0.2\left[\frac{350}{661.5}\right]^2\right)^{3.5} - 1\right]\left[1 - (6.875 \times 10^{-6})\,25{,}500\right]^{-5.2656}\right\} + 1\right)^{0.286} - 1\right]}$$

Fig. 1 – Mach number equation with values entered and ready to solve. How does your calculator stack up?

The manuals contend that a solution to a complex equation such as this demonstrates an important advantage of a system such as RPN; i.e., because you calculate one step at a time, you don't get 'lost' within the problem. You see every intermediate result, and you emerge from the calculation confident of your final answer[(3)]**.** The following is the listing of the keystrokes needed to solve the Mach Number problem in RPN on the HP33s.

Table 1 - Keystrokes used to Solve the Mach Number Equation[(3)]

Keys Pressed	Display	Comments
350 ENTER	350.00	
661.5 /	0.53	
x^2	0.28	Square of bracketed quantity
.2 x	0.06	

Keys Pressed	Display	Comments
1 +	1.06	
3.5 y^x	1.21	
1 -	0.21	Contents of left-hand set of brackets are in the stack.
1 ENTER	1.00	
6.875 E 6 +/- ENTER	6.88E-6	
25500 x	0.18	
-	0.82	
5.2656 +/- y^x	2.76	Contents of right-hand set of brackets are in the stack.
x	0.58	
1 +	1.58	
.286 y^x	1.14	
1 -	0.14	
5 x	0.70	
SQRT	0.84	Answer: Mach number of Daedalus' Harrier
	0.835724536 = ten digit machine answer.	

That is a total of 61 keystrokes on an HP-33s or HP-41C. I think that is the minimum possible on any HP RPN calculator because there are no requirements for use of a shift key. 62 keystrokes are required on an HP-35s because x^2 is a second function. 66 keystrokes are required for an RPN solution on an HP-17BII and HP-19BII because x^2, y^x and square root of x are second functions.

The advocates of algebraic tend to describe the solution to a textbook equation such as the Mach Number Equation as no more than simply entering the equation from left to right, pressing the equals key and voila!, without any agonizing or analysis to decide where to start the problem, the solution appears in the display. Actually it is typically a little more complicated than that because of idiosyncrasies of individual machines.

The true algebraic modes will permit the equation to be entered in a manner almost identical to that in the textbook equation with one exception, namely that an additional set of parentheses must be entered if the square root is to be taken of the entire expression. With an HP-10s the keystroke sequence is

SQRT (5 ((((((1 + .2 (350 / 661.5) x^2) y^x 3.5 - 1) (1 - (6.875 EXP (-) 6) x 25500) y^x (-) 5.2656) + 1) y^x .286 - 1)) =

where the notation (-) is the unary minus sign. That requires 74 keystrokes as written. Users who are familiar with the machine will reduce that to 70 by eliminating the parentheses surrounding the 6.875E-6 entry and eliminating the last two closing parentheses by using the characteristic that an equal sign closes all open parentheses. There is no easy way to examine intermediate results.

A different keystroke sequence is required with a true algebraic machine such as the HP-35s which uses a single parentheses key to simultaneously enter an opening and a corresponding closing parenthesis and does not offer implied multiplication. .(A similar methodology was used for parentheses, brackets and braces in earlier machines such as the HP48, HP49 and HP50.) The keystroke sequence is:

SQR 5 x P P P P P 1 + .2 x f SQ 350 / 661.5 > > y^x 3.5 - 1 > x P 1 - P 6.875 E 6 +/- > x 25500 > y^x 5.2656 +/- + 1 > y^x .286 - 1 > > ENTER

where I used the letter P to indicate the entry of the combined parentheses key, > to indicate the use of the right arrow key to move past a closing parenthesis and f to indicate the use of a second function. 75

keystrokes are required. Note that the SQR and SQ functions automatically insert necessary parentheses so that the expression is evaluated properly. An experienced user will reduce that to 71 by eliminating the parentheses surrounding the 6.875E-6 entry and eliminating the last two closing parentheses by using the characteristic that an ENTER closes all open parentheses. I admit that I struggled with this because of unfamiliarity with the double parenthesis methodology. There is no easy ability to examine intermediate results.
The almost algebraic mode of the HP-33s combines a left to right entry of the textbook equation except that the square root operator must come after the entry of the operand, not before. The mode also offers visibility into many of the intermediate results; however, recognition of the results is more difficult than with an RPN system. The following table shows the steps in a left-to-right solution of the Mach Number problem and the contents of the lower display at each step with the machine in FIX 2 mode. The comments in quotation marks are the same as those in the HP-67 Owner's Handbook and Programming

Table 2 - Keystrokes Used to Solve the Mach Number Equation in Algebraic on the HP-33s.

Press	Display	Comments	Press	Display	Comments
5	5		(	0.21	
x	5.00		1	1_	
(	5.00		-	1.00	
(	5.00		6.875	6.875_	
(	5.00		E	6.875E_	
(	5.00		6	6.875E6_	
(	5.00		+/-	6.975-6	
1	1_		x	6.88E-6	
+	1.00		25500	25500_	
.2	0.2		)	8.82	RPN
x	0.20		$Y^{\wedge x}$	0.82	
(	0.20		5.2656	5.2656_	
350	350_		+/-	-5.2656	
/	350.00		)	0.58	RPN
661.5	661.5_		+	0.58	
)	0.53	RPN	1	1_	
$X^{\wedge 2}$	0.28	RPN “Square of bracketed quantity”	)	1.58	RPN
)	1.06	RPN	$Y^{\wedge x}$	1.58	
$Y^{\wedge x}$	1.06		.286	0.286_	
3.5	3.5_		-	1.14	RPN
-	1.21	RPN	1	1_	
1	1_		)	0.14	RPN
)	0.21	RPN “Contents of left hand set of brackets are in the stack	ENTER	0.70	RPN
X	0.21		sqrt	0.84	RPN “Mach Number of Dacdalus’ Harrier”

Guide. The notation RPN in the comments column indicate displayed values which are the same as those indicated as appearing in the display of an HP-67 in that handbook.

There are 73 steps; however, 14 additional keystrokes are required because the entry of parentheses are second functions so a total of 87 keystrokes are required. Nearly all of the intermediate results which appear in an RPN solution are also available in the HP-33s solution. This should not be surprising. Examination of the keyboard sequence and the resulting output in the display in the table above shows that the HP-33s actually solves the problem from the inside out, not because the user asks it to, but because that is how it must be done. With RPN the user must examine the equation and find the proper inside point at which to begin. With ALG the machine essentially does the thinking for the user. Other "true algebraic" mechanizations also do the thinking for the user.

Solving the Mach Number equation on a calculator in adding machine arithmetic mode requires the appropriate insertion of additional sets of parentheses to circumvent the lack of precedence. For the HP-10B the steps in the solution will be:

```
5 x ( ( ( ( ( 1 + ( .2 x ( 350 / 661.5)SQ ) ) ^ 3.5 - 1 ) x ( ( 1 - ( 6.875 E - 6 x 25500 ) ) ^ 5.2656 +/- ) ) + 1 )
              1                1                    3  2                    2            3
^ .286 - 1 ) ) SQRT
```

Fig. 2 – HP-10B keystrokes for the Mach number equation.

where 79 steps are required. However, parentheses, E, x^2, y^x and square root of x are second functions with the result that the number of keystrokes required is 106. The numbers below the keystroke listing indicate the extra parentheses which must be added when solving with an AMA machine. The parentheses labeled 1 cause the value 0.2 to be multiplied by the square of 350/661.5 rather than be added to the 1. The parentheses labeled 2 cause the value 6.875E-6 to be multiplied by 25500 rather than be subtracted from the 1. The parentheses labeled 3 avoid the problem with y^x acting like an arithmetic operator and multiplying the result from the solution to the left hand set of brackets by the solution to the right hand set of brackets rather than simply raising the contents of the right hand set of brackets to the -5.2656 power.

Of course, it may not be entirely fair to use the solution of the Mach Number equation as a test of a machine that Wlodek's book[4] describes as "... a low cost business model ..." For the HP-17BII the input will be:

5 x (((((1 + (.2 x (350 / 661.5) SQ)) ^ 3.5 - 1) x ((1 - (6.875 E - 6 x 25500)) ^ 5.2656 +/-)) + 1) ^ .286 - 1)) SQRT

where 79 steps are required. However, E, x^2, y^x and the square root of x are second functions so the number of keystrokes required is 85. The negative sign for the exponent of 6.876E-6 inside the second bracket must be entered by using the minus arithmetic key before the value 6 is entered, not with +/- after the 6 is entered, because +/- after the 6 changes the sign of the mantissa not the sign of the exponent. The negative sign for the exponent -5.2656 at the end of the second bracket may be input either by using the minus arithmetic key before the value is entered or by using the +/- key after the value is entered. For the HP-19BII the input will be

5 x (((((1 + (.2 x (350 / 661.5) SQ)) ^ 3.5 - 1) x ((1 - (6.875 E - 6 x 25500)) ^ - 5.2656)) + 1) ^ .286 - 1)) SQRT

As with the HP-17BII 79 steps involving 85 keystrokes are required. The input sequence is very similar to that used with the HP-17BII except that the exponentiation key is labeled ^ not y^x. The rules for entering the negative signs associated with the exponent in scientific notation and the exponent with the y^x function are the same as with the HP-17BII.

Example #2

Now consider the solution of another relatively complex equation

[(3 + 1) (4 + 3) + (2 + 6) (4 + 6)] / [(2 + 3) (2 + 1) + (3 + 5)(4 + 2)] = 108/63 = 1.71428 57142 9 ...

which has been used to illustrate one of the disadvantages of the 4-level RPN mode offered in many HP calculators because its solution is not possible without storing an intermediate result. Actually, the real disadvantage is NOT the necessity to store an intermediate result, but rather the necessity to recognize that the intermediate result must be stored before it is pushed up and off the stack.

The keystrokes for solution on a true algebraic machine can be:

((3 + 1) x (4 + 3) + (2 + 6) x (4 + 6)) / ((2 + 3) x (2 + 1) + (3 + 5) x (4 + 2)) =

which involves 52 keystrokes as written. However, features provided by some machines may make a smaller key count possible. On an HP-35s the entry sequence is:

P P 3 + 1 > x P 4 + 3 > + P 2 + 6 > x P 4 + 6 > > / P P 2 + 3 > x P 2 + 1 > + P 3 + 5 > x P 4 + 2 > >
ENTER

Fig. 3 – Keystrokes for the example #2 problem on the HP-35s.

where as before I used the letter P to indicate the use of the combined parentheses key and > to indicate the use of the right arrow cursor key to move past a closing parenthesis. The keystroke count is 52. On a machine like the HP-10s which offers implied multiplication the number of keystrokes can be reduced to 48 as in:

((3 + 1)(4 + 3) + (2 + 6) (4 + 6)) / ((2 + 3) (2 + 1) + (3 + 5)(4 + 2)) =

With the almost algebraic mode on an HP-33s the solution can be

f (f (3 + 1 f) f (4 + 3 f) + f (2 + 6 f) f (4 + 6 f) f) / f (f (2 + 3 f) f (2 + 1 f) + f (3 + 5 f) f (4 + 2 f) f) ENTER

Fig. 4 – Keystrokes for the example #2 problem on the HP-33s.

where the machine accepts adjacent parentheses (i.e., implied multiplication) from the keyboard and inserts the multiply sign into the display of the equation in the upper line. The keystroke count is 68 which illustrates how severe the penalty can be for having parentheses as second functions. Users who are familiar with the feature where an = or an ENTER closes the outstanding parentheses can reduce the number of keystrokes for the HP-10s and HP-35s by two keystrokes and for the HP-33s by four keystrokes.

The keystroke sequence for the AMA machines is:

(((3 + 1) x (4 + 3)) + ((2 + 6) x (4 + 6))) / (((2 + 3) x (2 + 1)) + ((3 + 5) x (4 + 2))) =

where additional parentheses must be inserted to circumvent the lack of precedence. 60 keystrokes are required with an HP17BII or HP19BII. 28 additional keystrokes are required with an HP10b where parentheses are second functions.
The perception that the problem can not be solved on a 4 level RPN machine without storage of an intermediate result is incorrect. Without even using the LAST X function the following sequence will solve the problem

2 ENTER 3 + 2 ENTER 1 + x 3 ENTER 5 + 4 ENTER 2 + x + 3 ENTER 1 + 4 ENTER 3 + x 2 ENTER 6 + RollDown x<>y / RollDown / x<>y RollDown 4 ENTER 6 + x +

Fig. 5 – Keystrokes for the example #2 problem using RPN.

which is hardly a credible solution for most users. The solution requires only 45 keystrokes on an HP-41,

The solution requires 50 keystrokes on an HP-17BII or HP-19BII where RollDown and x<>y are second functions. This problem is an interesting exercise but I think that a knowledgeable RPNer who even suspected that there might be any difficulty with stack overflow could be expected to solve the denominator in a straightforward manner, store the result, solve the numerator, recall the denominator and divide. That can easily be done in 45 keystrokes on an HP-41 and in only 43 keystrokes on an HP-17BII or HP-19BII. The problem can be solved directly without intermediate storage on machines such as the HP-28S, HP-48 series, HP-49 and HP-50G which do not use a limited stack.

Limitations on the Number of Parentheses and Pending Operations

A little study of the solution of the Mach Number equation on the AMA machines will suggest that the need for a third set of additional parentheses to avoid an error being introduced by the y^x at the end of the right hand brackets could be avoided by interchanging the material in the left and right hand set of brackets. That turned out to be true for the HP-17BII and the HP-19BII. But, with the HP-10B the calculation would proceed satisfactorily until after the entry of the 350 value where an "Error - Full" message would occur when the divide was entered. On the HP-10B that problem can be avoided by moving the multiplication by 5 from the beginning to the end of the calculation. Of course, rearranging the equation for entry into the machine defeats the claimed advantage of algebraic modes, namely the ability to enter the equation in textbook form.

The "Error - Full" message is associated with a limit on what are called pending operations with algebraic implementations. The classic test to determine that limit for a given machine is to enter the sequence

(1 + (2 + (3 + (4 + (5 + (6 + ...

When I did that with my HP 10B the "Error - Full" message appeared when the plus after the 4 was entered indicating that the limit on pending operations was four. I did the test with my HP35s, HP-17BII and HP-19BII. When I did not receive any error when 20 was entered I stopped the test and pressed = which closes all the open parentheses and completes the calculation. All three machines yielded the correct sum of 210.

When I did the test with my HP33s I received the following display:

(1 + (2 + (3 + (4 + (5 + (6 + (7 + (8 + (9 + (10 + (11 + (12 + (13 + 14 + 15 + ...
where the machine stopped accepting parentheses after thirteenth one. The old adage says "When all else fails, read the instructions." so that is what I did. Page C-5 of the HP33s Scientific Calculator User's

Guide says "In ALG mode, you can use parentheses up to 13 levels. If I press = after the entry of 13 I get the correct sum of 91 indicating that at least 13 pending operations are available.
I did the test with my HP-10s and did not receive any error when the entry reached 20. But when I pressed = the message "Stack ERROR" appeared. I pressed the left arrow and the + sign after the 11 in the series flashed. I deleted the entries after the 11, pressed = and the correct sum of 66 appeared. This all suggests that the HP10s has a limit of ten pending operations. Unhappily, the machine lets the user go merrily on his way with the entry of more pending operations and does not indicate a problem until the = sign is pressed.

How do they Compare?

One measure of calculator usage is "efficiency" based on the number of keystrokes required for solving a problem. Table 3 summarizes the keystroke counts from solutions of the problems described above.

Table 3 – Keystroke Counts for the Two Examples Discussed

Scientific Models	Mach Number	Example Two
HP10s	74	48
HP33s	87	68
HP35s	75	52
Business Models	-	-
HP-10B	105	88
HP-17BII	85	60
HP-19BII	85	60
RPN HP-41	61	45

The information in the table translates into an efficiency advantage for RPN of ten to thirty per cent depending on which type of algebraic is being used. I don't find that to be a persuasive reason to choose RPN over AOS. My indifference to keystroke efficiency is related to the way that calculators and computers are used by engineers.

My perception is that engineers are paid to analyze and solve problems not to enter data or programs into machines. Thus, keystroke entry tends to be a small, bordering on insignificant, part of the typical day, the typical week, and the typical year of an engineering career. I look at it this way. If a typical engineering day involves seven hours of thinking and problem formulation and solution and one hour of data entry would an employer care or even notice if one machine were ten per cent or even twenty-five per cent more efficient? I admit that there must be some careers where keystroke efficiency is important. I suspect that those careers are closer to that of a clerk-typist than to that of an engineer.

Additional Algebraic Mechanizations

A discussion of algebraic mechanizations in calculators manufactured by HP would not be complete if it did not address user preference; i.e., whether to use RPN or algebraic. On page 39 of his book *A Guide To HP Handheld Calculators and Computers,* Wlodek Mier-Jedrzejowicz wrote; "Many users -- and not only those brought up on Friden calculators -- prefer RPN. Others find it a total mystery -- or at least profess to do so; I do not really believe it -- ... when HP-35A calculators were introduced in the university where I studied I found it took about 5 minutes to teach other students how to use an HP-35." That is the way I was introduced to RPN. There was a reference to how easily RPN could be used to solve a complicated problem such as the Mach number equation. There was no mention of stack limitations that could lead to an incorrect answer to a problem as innocuous as the one discussed here.

Of course, Wlodek's comment was written from the viewpoint of a proponent of RPN. I am a long timeproponent of algebraic. For eight years I was the editor of a newsletter for algebraic machines such as the TI-59, TI-95, the TI CC-40, and the Casio fx-7000G. As a result of that experience I suggest that an appropriate corollary to Wlodek's statement would be: Many users prefer algebraic notation including the use of parentheses because it allows them to enter textbook equations directly into their machines. Many RPN users profess to find the use of parentheses a total mystery -- or profess to do so. I do not really believe that either. I do admit the proponents of algebraic have a long standing prejudice in favor of direct textbook entry; e.g., page 1-8 of TI's manual for the use of the TI-30, *The Great International Math on Keys Book* (1976) presents an algebraic expression and states "You can key the above problem directly, left to right, into your (TI-30 type) calculator with AOS and you'll get the correct answer. (Not all calculators will do this.) ..." Similar words appear in their manuals for all of the calculators manufactured by TI. An unsuspecting user will be unhappily surprised when first experiencing an error condition due to limitations on the number of open parentheses or pending operations.

A comparison of the advantages of RPN and algebraic onpages 12-4 of the HP-33s User's Guide lists the strengths of RPN to Use less memory, & Execute a bit faster and the strengths of Equations and ALG operations as Easier to write and read & can automatically prompt.

The same material appears on page 13-5 of the HP35s User's Guide. The idea that RPN uses less memory than algebraic has not been seriously challenged as far as I know. Even in the days of the so-called "friendly competition" the TI community admitted that 100 steps inn an HP RPN program offered more computing power than 100 steps in a TI AOS program. I obtained a current comparison by entering the Mach Number equation into the HP33s as a program and reading out the program length. The RPN version used 270 bytes. The algebraic version used 303 bytes.

Relative speed of the two methodologies was more difficult to assess in the days of the friendly competition and was strongly associated with the algorithm and processor speed. The newer machine was typically faster. A challenge by Gene Wright in the Forum section of The Museum of HP Calculators gave me an idea for a more direct comparison, again using the HP-33s. I added a simple GTO loop to the Mach Number programs and counted the number of times the programs could be completed in one minute The RPN version completed 191 loops. The algebraic version completed 171 loops.

But it's all about what the user is used to. Richard Nelson reminded me of Dr. Wicke's question "How do you differentiate between friendly and familiar?" A user who has learned a keystroke sequence by using it many times may be able to find and press several keys in less time than with a less familiar, fewer keystroke sequence. A similar effect is an important aspect of speed in touch typing. Familiar and frequently used words are typed in response to the appearance of the word without really thinking about the individual letters in the word.

Conclusions and Recommendations

During the preparation of this article I had an epiphany of sorts. I have always been amused and bemused by the insistence of the RPN community that solution of relatively complex equations such as the Mach Number equation was difficult, if not downright impossible, with algebraic machines. I now suspect they were using machines with AMA algebraic. As discussed in this article with AMA version of algebraic it is NOT POSSIBLE to simply enter the Mach Number equation from left to right but IS NECESSARY to insert additional parentheses to circumvent the peculiarities (at least to scientific people) of the AMA mechanization. I struggled with that when preparing for this article. If I hadn't had the incentive of preparing for the article I most certainly would have given up.

The major deficiency of the algebraic mode in the HP33s and HP35s is in the handling of the input of parentheses. Part of the problem is a shortage of keyboard space. RPNers complained about the omission of keyboard functions that they wanted in order to provide parentheses input at all. Algebraic users complained about parentheses as shifted functions or as shared on a single key.

Here's a possible solution for calculators which offer both RPN and algebraic: RPNers frequently use RollDown and x<>y. Algebraic users rarely, if ever, do. Why not share the RollDown function with the left parenthesis function and the x<>y function with the right parenthesis function with the software deciding which option to use depending on which mode the calculator is in.

Glossary

Algebraic – The algebraic mode in machines such as the HP-10s and HP-35s which operate in a manner which is consistent with the time-honored "My Dear Aunt Sally"**(1)** form of precedence.

AMA – Adding Machine Arithmetic. No user logic implemented as exemplified by a mechanical adding machine. Math operations are implemented as they occur. Also known by ATH.

BASIC – per Wikipedia: (an acronym for **Beginner's All-purpose Symbolic Instruction Code**) is a family of high-level programming languages. The original BASIC was designed in 1964 by John George Kemeny and Thomas Eugene Kurtz at Dartmouth College in New Hampshire, USA to provide computer access to non-science students.

CLI – Command Line Interface typically used by the majority of graphing calculators.

FORTRAN – per Wikipedia: **Fortran** (previously **FORTRAN**; blends derived from **IBM Mathematical Formula Translating System**) is a general-purpose, procedural, imperative programming language that is especially suited to numeric computation and scientific computing.

Operand – The values that an operator operates with.

RPN – Reverse Polish Notation. A legacy calculator logic system as advocated by Hewlett-Packard.

RPL RPN – Reverse Polish Lisp Reverse Polish Notation. A calculator logic system as implemented by Hewlett-Packard on newer RPL machines. Pressing ENTER on these machines does not replicate X into Y (Level one into level two of the stack).

HP Algebraic Notes

*(0) There is, however, a very strong case made by Joseph K. Horn for the HP-71B as being one model that most closely approaches the "ideal." See the excellent article The HP-71B 'Math Machine that describes the HP-71B in **HP Solve** Issue 17. Joseph makes the case for the "ideal" scientific calculator user interface.*

(1) My Dear Aunt Sally is a popular memory aid taught in schools and textbooks to remember Multiply, Division, Addition, and Subtraction as the order of precedence for algebraic expressions.

*(2) See **HP Solve** Issue #4 RPN Tips for a description of the four basic calculator logic systems (user interfaces).*

(3) The words in the table are quoted from page 107 of the HP-67 Owner's Handbook and Programming Guide.

(4) A Guide to HP Handheld Calculators and Computers, 5th Ed., by W.A.C Mier-Jedrzejowicz, Ph.D. for additional details see: http://www.hpcalculatorguide.com/

About the Author

Palmer Hanson worked on autopilot and bombing systems on F-100 and F-101 aircraft and on inertial navigation systems for aircraft such as the A-11 and YF-12 Blackbirds, the X-15, the B-52, and the AV-8. He authored six technical papers on inertial navigation. He met his first computer in the Navy in 1952 -- the electromechanical MK 1A which was part of the Mk 37 Gun Fire Control System. He met his first digital computer at the University of Minnesota in 1960 - the RemRand 1103. He met his first programmable calculator at Honeywell -- the TI-59. He authored five articles and numerous letters to the editor on the use of portable computers and programmable calculators in publications such as TRS-80 News, Byte, and PPX Exchange. From 1983 through 1991 he was the editor and publisher of TI PPC Notes, a newsletter for users of hand-held programmable calculators. That work was recognized by inclusion in the Who's Who in the South and Southwest. Since retirement he spends about half the year in Largo, Florida and half in Brevard, North Carolina.

Better Problem Solving–Part I

HP Solve #22 page 21

← Previous **Article** – Next →

Better Problem Solving Part I

Richard J. Nelson

Introduction

The basic nature of human beings is to do things in the simplest and easiest way possible. One measure of societies "advancement" is its use of tools. We move about using our vehicles, and we eat our food with utensils. Practically everything we do everyday involves the manipulation of the physical objects that make up our lives. The tools and methods we use represent our ability to solve problems. If I break a leg I am able to still be mobile by using crutches or a wheel chair.

The tools we use must provide two functions be truly useful.

(1) They must do the required job easily, and
(2) they must do the job quickly.

We humans are always in a hurry to get the job done so we are able to tackle the next job, especially if it is one that we really enjoy doing. Our brains work so fast that we often feel limited by our tools in terms of getting the job done. Often this limitation is measured by the amount of time we spend in the process.

This series of articles will address the various aspects of problem solving related to calculators. This series should not be confused with the Math Review series found elsewhere in this issue of ***HP Solve***. The objective of this series is to review how we solve problems with our calculators and how we might change the method or process so that we might solve them more easily and faster.

Problems with Two Possibilities – Δ%

A regular column in ***HP Solve*** is One Minute Marvels, OMMs. These are short (being able to key them into your machine in a minute or less) HP48/49/50 programs or routines that are examples of an elegant solution or the solution of an unusual problem. In ***HP Solve*** issue 14 an OMM program was described to calculate what is often called delta percent. Given two values, e.g. your taxes for last year, and your taxes for this year, you are either to calculate the percentage increase or the percentage decrease. To keep the math simple let's assume that the two numbers are \$ 1,000 and \$1,150 for last year and this year respectively. First you have to key the two numbers and then subtract the two values. Next you need to divide the difference by either last year (1,000) or this year (1,150) depending upon which you wanted, the % decrease or the % increase. The result of the division must then be multiplied by 100 to convert the value into percent[(1)].

The basic mechanics of the solution are as described above. You, the problem solver, have to do a bit of thinking in order to get the right answer. Suppose, however, you didn't pay as much this year as you did last year and the two tax values were reversed?

Some HP calculators have a delta percent function built in and this function will save you a little time.

Because I am human with the traits of wanting to keep life and problem solving as simple as possible I will suggest that there is a better way. Because this "better way" is a different way you may resist its approach. I guarantee, however, that your basic nature will take over if you actually use this approach for a while. After ten years of using this method of calculating Δ% I will bet you lunch that you will want to have the delta percent function work this way. Here is the process as described in ***HP Solve*** Issue 14.

- ♦ Step 1 – Key in the first number (the order doesn't matter).
- ♦ Step 2 – Key in the second number.
- ♦ Step 3 – Press the Δ% key.

That is all there is to solving Δ% problems. It is as easy as One-two-three. No thinking is require to solve the problem, none! This Δ%, however, is different in that it gives you two answers and you will easily and naturally choose the one you want. On an RPN or RPL machine there are two numbers on the stack. See fig. 1 below.

Y:	L2	15.00
X:	L1	-13.04

Fig. 1 – Δ% calculation example

The numbers for our example tax problem are 15% and -13.04%. The negative number is on the lower level (X register or level one). This represents the percentage the smaller number is compared to the larger number and the next level up (Y register or level two) is the percentage the larger number is over the lower number (the increase in taxes). These two values will always be in this order and you will quickly and instinctively know which number you need for your problem. This approach is so unconventional that most mathematics types will immediately rebel. I know, however, that if you use the program a few dozen times with real world data that you will soon realize how much easier Δ% problems are solved with this proposed approach.

Problems with Two Possibilities – conversions

The same approach may be used for conversions. Suppose you want to convert temperatures. How many degrees is 72 degrees Fahrenheit in degrees Celsius? Some HP calculators have this conversion on a key labeled →F or →C. Most modern calculators use a dot matrix display which facilitates alphanumeric characters so this suggestion for temperature conversion should also take advantage of this capability.

The input is 72 and a single function converts this value to both Fahrenheit and Celsius. The order will always be the same as shown below.

Y:	L2	72.00° F is 22.22° C°
X:	L1	72.00° C is 161.60° F

Fig. 2 – New Temperature Conversion

This proposed approach is suggested as a possible improvement for entry level or midrange machines. The more advanced unit conversion systems of the high-end (48/49/50) machines with their large number of conversions may not benefit from this approach.

Unit conversions may be compared to provide perspective using this approach. An example is a unit conversion, "1", such as the avoirdupois ounce and the troy ounce is illustrated in Fig. 3.

Y:	L2	1.00 oz is 0.91 ozt
X:	L1	1.00 ozt is 1.10 oz

Fig. 3 – New oz conversion proposal

Problems With Two Possibilities – Function & Inverse

Fig. 4 – HP-35s Trig. Keys.

Adding labeling to answers to take full advantage of an alphanumeric display may also use the "both answers" solution to other functions such as Sin, Cos, Tan, LOG/LN, etc. See typical key designations in Fig. 4. 30 degrees as an input to TAN would result in the display of Fig. 5. This application is less appealing than the examples above and it is included only to further illustrate the possibilities.

Y:	L2	ATAN 30.00 is 88.09
X:	L1	TAN 30.00 is 0.58

Fig. 5 – Tan inverse/function example.

What about chain calculations? If the answer you need is on level 2 or in the Y register SWAP or X⇄Y may be used. Alternately a key that eliminates the bottom answer only leaving the top answer could be used. This "function" would be used for all two-answer results.

Keyboard clutter would be reduced using this approach because a single function essentially replaces two functions.

Problems With Two Possibilities – Function & Inverse alternate

Another aspect of making problem solving simpler and easier is making the keyboard as simple and uncluttered as possible. Have you ever been in a very cluttered room looking for something? There is so much "stuff" everywhere that your eyes are not able to "process" every detail in a reasonable amount of time. The person who lives in the room knows where things are and will find things easily.

This aspect of calculator usage is familiar vs. friendly (experienced vs. first look). Reducing keyboard clutter has been addressed with ideas such as soft menus, shift keys, and even audio feedback. Ideas that seem strange today may become commonplace in the future as more people get used to a particular way the user interacts with the machine. One method is to use the timing of key pressing. This may be implemented in at least two ways.

1) Press and hold the key when another key is pressed. This is an old "shift" key concept going back to the mechanical typewriter days.

2) Press the same key twice in succession. If the time between pressings is too long the alternate function is not executed.

Fig. 6 – HP-15C Trig. Keys.

Double pressing a key for its inverse function is a simple idea that reduces keyboard clutter. See example of Trig. function keys in Fig. 6. The blue notations could be omitted and replaced with other functions. If you need the arcsine function you simply press the SIN key twice within ¼ of a second. If the time exceeds the ¼ second time the SIN function is executed.

What I am proposing is to replace the shifted inverse function keystroke with a ½-keystroke. In its simplest form pressing a calculator key involves two parts. First your eye has to search for and find the key. Second the key must be pressed. If you do not have to do the first part you save ½ of a keystroke and the inverse notation doesn't need to be on the keyboard.

Do you have an idea for a regular scientific calculator function and how it may be "improved" to make it easier, faster, or simpler to use? If so send it in to the editor for possible inclusion in this series of articles.

What do you think? Have you tried using the Δ% program in OMM ***HP Solve*** Issue #14? Send an email to the editor with your thoughts on how classical problems may be better solved on future calculators.

Email ***HP Solve*** at: hpsolve@hp.com

Better Problem Solving Part II will suggest a method of implementing a function that requires a single key press and three inputs. The function then decides which of 19 solutions is the correct one and then calculates four outputs. If this function is added to all mid range and high end scientific calculators it would save hundreds of hours of involved calculations for just about every technical user. The problem is a very common everyday type problem.

Better Problem Solving Part I Notes

(1) You may also calculate the value by {(Last Year) / (This Year)} -1 x 100. This method is useable on any calculator and it will save keystrokes. It still requires that you key the values in the proper order.

About the Author

Richard J. Nelson has written hundreds of articles on the subject of HP's calculators. His first article was in the first issue of *HP 65 Notes* in June 1974. He became an RPN enthusiast with his first HP Calculator, the HP-35A he received in the mail from HP on July 31, 1972. He remembered the HP-35A in a recent article that included previously unpublished information on this calculator. See http://holyjoe.net/hhc2007/Remembering%20The%20HP35A.pdf He has also had an article published on HP's website on HP Calculator Firsts. See http://h20331.www2.hp.com/Hpsub/cache/392617-0-0-225-121.html.

PROOT: A Blast From the Past!

HP Solve #22 page 26

← Previous **Article** – Next →

PROOT: A Blast From the Past!

Namir Shammas

The PROOT function in the graphing calculators, starting with the HP-48G/G+/GX models, has its roots in the HP-71B handheld BASIC computer. To be exact, the PROOT function first appeared in the Math Pac for the HP-71B. This ROM offered support for the following features:

1. Complex math, functions, variables, and arrays.
2. Hyperbolic functions.
3. Matrix/vector operations.
4. Solution to systems of equations.
5. Root calculations for nonlinear functions.
6. Root calculations for polynomials.
7. Numerical integration.
8. Finite Fourier transforms.

The PROOT function in the HP71B Math Pac handled polynomials with only real coefficients. The function returned real and/or complex roots. HP developers added support to complex polynomial coefficients when they ported the PROOT function to the graphing calculators.

The PROOT function is based on the Laguerre method. I have discussed this method is an earlier ***HP Solve*** issue. The PROOT function has the following features:

1. Works with polynomials that have real coefficients.
2. Solves for all real and complex roots.
3. There is no need to supply guesses for the roots or tolerance limits for the answers. The function PROOT internally determines the initial guesses and works to maximize the accuracy of the answers.

Why Should I Care About the HP-71B?

The HP-71B is one of the vintage HP computing machines that has a dual aspect. It is a calculator and a handheld BASIC computer. The HP-71B spearheaded other handheld computers by implementing the IEEE floating point math standard (still in proposal stage at the time) which included such new concepts as infinity and NAN (not-a-number). Armed with a Math Pac and a Curve Fitting Pac (which also offers a nice optimization engine), the HP-71B is a formidable machine that works with an HP version of legacy BASIC. In general, coding in BASIC has been and remains easy to follow and read. If you feel nostalgic in reading this article and have no HP-71B at hand, you can download the EMU71 software from the internet (http://www.jeffcalc.hp41.eu/emu71/index.html) which emulates the HP-71B in a DOS box. This emulator also comes with a Math Pac and other goodies! So you are set to go!

Demonstrating The PROOT Function

Using the PROOT function with the BASIC-driven HP-71B is easy. Just keep the following simple rules in mind:

1. Create an array used to store the polynomial coefficients. The array should have n+1 (where n is the polynomial order) elements, if you want to take advantage of the MAT commands in the Math Pac.
2. Store the polynomial coefficient in a floating-point array. The first element of that array should contain the coefficient for the term with the highest order, and so on. This also means that the

3. constant coefficient is stored in the highest-index array element.
4. Create an array of complex numbers that has n elements.
5. Call the PROOT function, which stores the calculated real and complex polynomial roots in an array of complex numbers. As far as the machine is concerned all the roots are complex. The real roots are the ones with the imaginary part being zero or a very small number.

Here is a short BASIC program that prompts you to enter the order and the coefficients for a polynomial. The program displays the roots using

```
10 OPTION BASE 1
20 INTEGER N
30 INPUT "ENTER POLYNOM ORDER? ";N
40 DIM A(N+1)
50 COMPLEX B(N)
60 MAT INPUT A
70 MAT B=PROOT(A)
80 FOR I = 1 TO N
90 DISP B(I) @ PAUSE
100 NEXT I
110 DISP "DONE"
```

The program performs the following main tasks:

1. Sets the Option Base to 1 so that the lowest index for arrays and matrices is 1 and not the default of 0. This indexing scheme is merely convenient for the above program.

2. Prompts you to enter the polynomial order, N.

3. Creates the array A to have N+1 elements. The program uses this array to store the polynomial coefficients.

4. Creates the complex array B to have N elements. The program uses this array to store the polynomial roots.

5. Prompts you for the polynomial coefficients using the MAT INPUT command. You can enter each coefficient individually, as prompted, or type multiple coefficients on the input line. In the latter case, you need to separate the values using commas.

6. Invokes the PROOT function to calculate the polynomial roots. The program stores the result of PROOT in the array B.

7. Displays the elements of complex array B using a FOR loop. Each loop iteration has a PAUSE statement, allowing you to examine the roots at your own pace as the program pauses. Invoke the CONT command from the keyboard to resume program execution and view the next root. You can replace the FOR loop, in lines 80 to 100, with the single command MAT DISP B. However, this command tends to display the array elements rather quickly.

8. Displays the word DONE when the program reaches the end.

Let's use the above BASIC program to find the roots of the following polynomial:

$$Y = x^3 + 2*x^2 + 3*x + 4$$

Here the session to calculate the roots for the above polynomial. Your input appears in bold and underlined characters. The command keys appear in bold and are enclosed in square brackets. For example [ENDLINE] refers to the ENDLINE key:

```
[RUN]
ENTER POLYNOM ORDER? 3[ENDLINE]
A(1)? 1,2,3,4[ENDLINE]
(-.17468540428,-1.54686888723)[f][+]
(-.17468540428,1.54686888723)[f][+]
(-1.65062919144,0)[f][+]
DONE
```

The key sequence **[f][+]** triggers the CONT command that allows you to resume program execution. The program calculates and displays two complex roots and a real root for the given polynomial.

Analyzing PROOT Errors

How are the errors in calculating the duplicate roots affected by their values and their count? This a question that I raised and answered for the graphing calculators' PROOT functions in a previous HP Solve issue. In this article, I ask the same question for the HP-71B Math Pac.

First I will define a domain of values for the duplicate roots and the number of duplicates. I studied the roots of 1, 2, 3, 4, 5, 10, 20, 30, 40, 50, and 100. As far as the number of duplicates, I considered the values in the range of 5 to 40, in increments of 5. This is the same domain of numbers I used for the graphing calculator analysis.

I used the following BASIC program to calculate a single value that summarizes the errors in the obtained roots:

```
5 REM OPTION BASE 0 IS IN EFFECT BY DEFAULT
10 INTEGER N
20 INPUT "ORDER? ";N
30 INPUY "DUPLICATE ROOT? ";R0
40 DIM C(N) @ COMPLEX R(N - 1), R1(N - 1)
50 MAT R1 = ((R0,0))
60 MAT C = ZER @ C(N) = 1
70 FOR L = 1 TO N
80 FOR K = N - L TO N - 1
90 C(K) = C(K+1)-C(K)*R0
100 NEXT K
110 C(N) = -C(N)*R0
120 NEXT L
130 MAT R = PROOT(C)
140 MAT R1 = R - R1
150 E = FNORM(R1) / SQR(N)
160 DISP E @ PAUSE
170 MAT DISP R
```

Store the above program object in the variable PC. This program takes the value of the duplicate root and the number of duplicates from levels 2 and 1, respectively. The program performs the following tasks:

1. Prompts the user to enter the polynomial order and the duplicate root value.

2. Declares the real and complex arrays needed by the program. The array C stores the polynomial coefficients. The complex array R stores the calculated polynomial roots. The complex array R1 stores the value of the duplicate root.
3. Assigns the values of the duplicate root to all the elements of array R1 (in line 50).
4. The statements in lines 60 to 120 calculate the polynomial coefficients and store them in array C.
5. Calls the function PROOT, in line 130, to calculate the roots of the polynomial, storing them in array R.
6. Subtracts the array of calculated roots from the array of supplied roots in line 140. This task uses a MAT command to subtract the values in arrays R and R1 in a single swoop.
7. Calculates the norm of the array of root and divides it by the square root of the number of duplicated roots. The result, stored in variable E, is a measure for the square root of the mean squared errors. I will call this result NormErr.
8. Displays the NormErr value, in variable E, and then pauses program execution.
9. Displays the array of calculated roots after you invoke the CONT command. You are not obligated to view the roots. You can instead skip this task and start again with task 1 to study another duplicate root.

Here is a sample session with the above BASIC program. Let's run the program to calculate the roots of the polynomial (x – 1)^5 and its errors. This is polynomial of order 5 with 1 as the duplicated root:

```
[RUN]
ORDER? 5[ENDLINE]
DUPLICATE ROOT? 1[ENDLINE]
7.54717951647E-7[f][+]
(.999998938682,0)
(1.00000026533,0)
(1.00000026533,0)
(1.00000026533,8.4852813742E-7)
(1.00000026533,-8.4852813742E-7)
```

The NormErr value is 7.547E-07 and the five roots are very close to 1 (with the imaginary parts being 0 or very small values). Table 1 shows the results for the domain of values I chose--the square root of the mean squared errors in a two dimensional table of duplicate root values and number of duplicate roots.

Table. 1- The resulting square roots of the mean squared errors (NormErr).

		Number of Duplicate Roots						
Root	5	10	15	20	25	30	35	40
1	7.54718E-07	0.0083082	0.115248156	0.312344571	0.52104063	0.640049866	0.783125845	0.427179263
2	1.11055E-06	0.047485168	0.300285767	0.681751198	0.818744126	1.487085026	1.90840792	2.280244037
3	2.13388E-06	0.104815717	0.361728784	1.017550377	1.953900504	2.560174349	3.058388786	3.463778361
4	3.14669E-06	0.012121824	0.601958039	1.833659839	2.606713423	3.366875598	3.933409313	4.548226861
5	1.87971E-06	0.214440152	0.723073044	2.413900034	3.207609566	4.272291389	4.962886843	5.813349385
10	6.16712E-06	0.083573062	1.571792692	3.078166377	5.081298458	6.400264524	7.950693395	9.184036127
20	7.83154E-06	0.375879626	3.178523712	6.28979807	9.801778486	14.89794146	19.08430516	22.80205451
30	1.82493E-05	0.30583898	4.404651736	9.794744761	19.53871975	25.60169507	30.58419254	34.63779514
40	3.11929E-05	0.12232013	6.02397928	18.33662058	26.06771717	33.66914546	39.33407894	45.48218349
50	4.27488E-05	2.144177774	5.856138001	24.14111473	32.07754071	42.72342353	49.62855435	58.1334889
100	7.54718E-05	0.831348223	11.45957919	30.78155686	50.29599742	64.19602825	79.75180325	91.29228563

Figure 1 shows a linear plot for the NormErr results. The graph includes a legend. Series 1 refers to 5 duplicate roots, series 2 refers to 10 duplicate roots, and so on. The lines are not perfectly linear because of the rounding errors. The curve for 10 duplicates shows the most variations. These variations (or

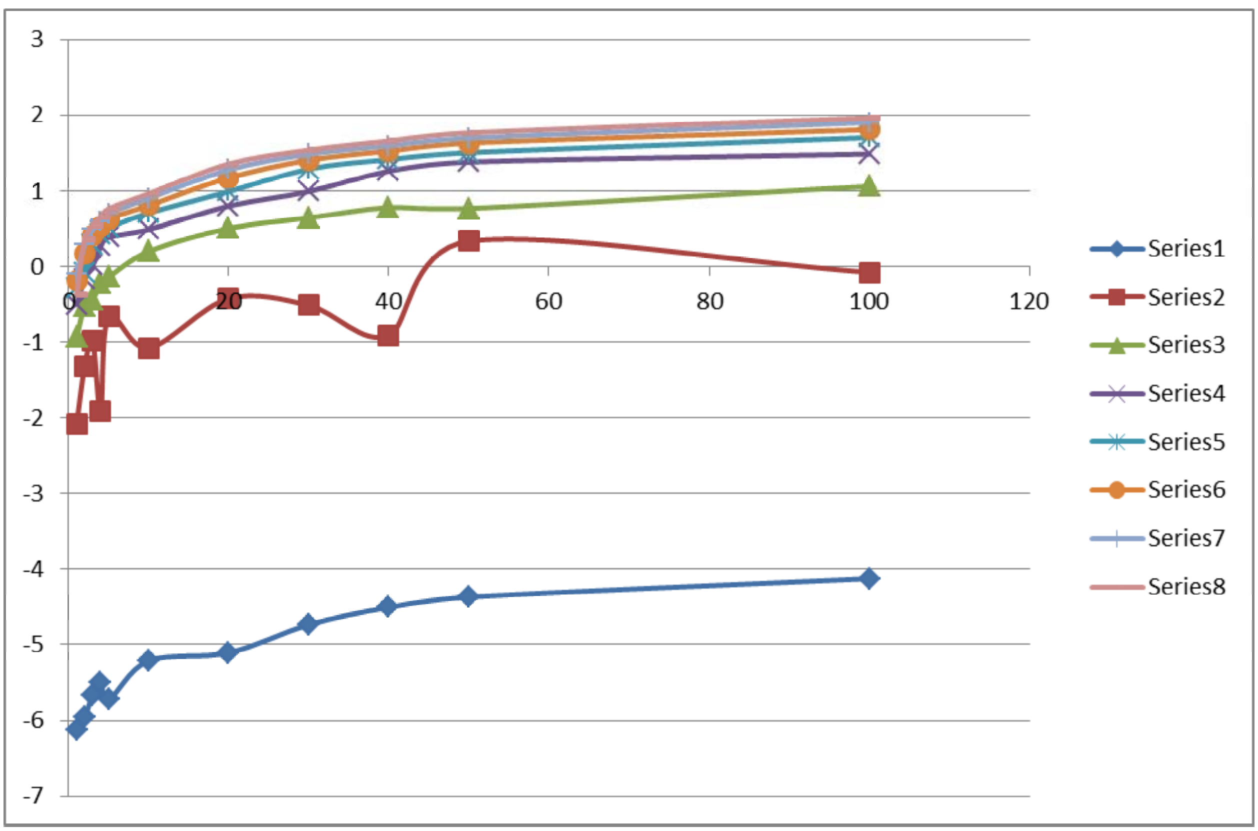


Fig. 1- A linear plot for the results.

deviation from an expected smooth curve) may be due to special schemes used to obtain accurate answers that seem to be most effective for the polynomials of order 10. Figure 2 plots the same data using log-log scales. The series labeling is the same as in Figure 1.

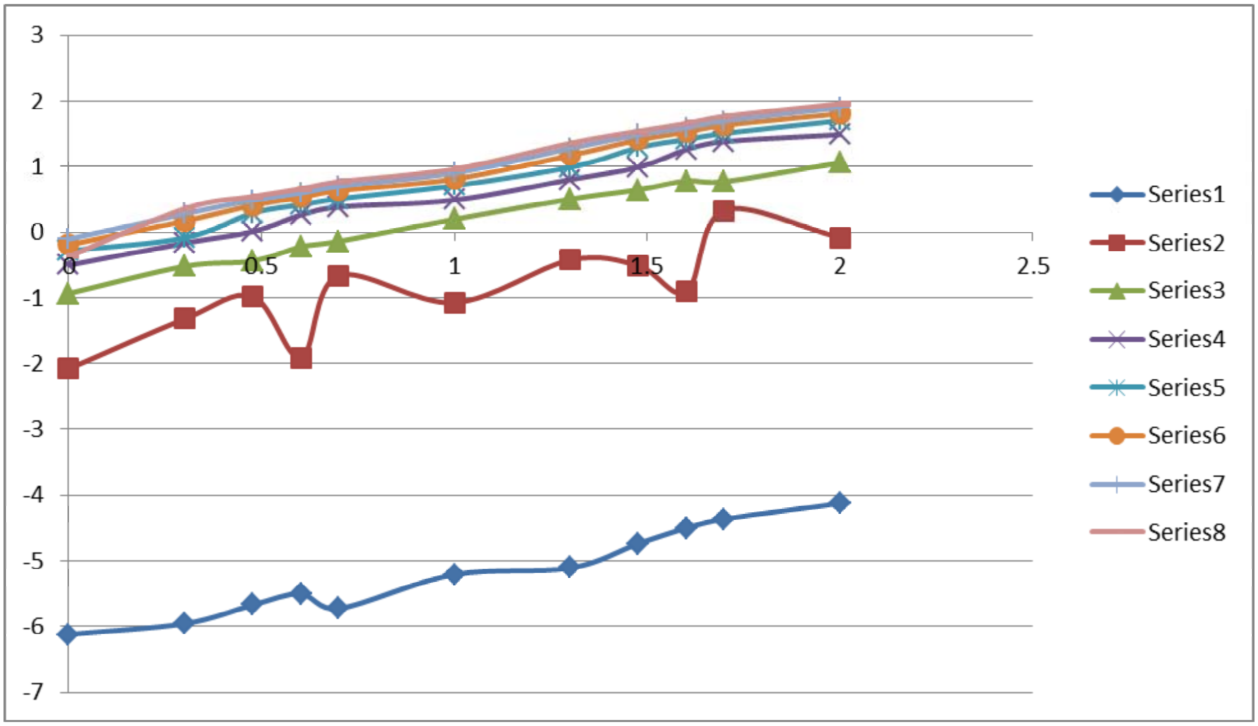


Fig. 2- A log-log plot for the results.

Applying simple linear and multiple linearized regressions to the data, I found the following empirical model:

Number of observations = 88
F statistic = 1839.17
Adjusted R-Square = 0.97688
ln(NormErr) = –0.0896701802129991+ 1.01052899010609 * ln(Root) – 359.972416836311/N^2

The above model agrees with Figure 2 which shows plots that are somewhat linear on the log-log scale. What makes the above model interesting is that it not only came out as the best model (among hundreds of competing models) but it also is similar to the best model for the PROOT errors on the HP-50G calculator:

Number of observations = 88
F statistic = 4229.83
Adjusted R-Square = 0.98982
ln(NormErr) = –0.112444241487657 + 1.01931043509749 * ln(Root) – 357.981953542852 /N^2

The corresponding coefficients in the above two models have values that are fairly close to one another. This is truer for the slopes. This observation leads me to conclude that the algorithms (and its special strategies to handle special cases and strive to obtain very good accuracy) used with the HP-71B Mat Pac and with the graphing calculators are basically the same.

Conclusion

The Math Pac of the HP-71B offers the first incarnation of the PROOT function. This initial implementation lacks the ability to handle complex polynomial coefficients. Other than that, it does a very good job in finding unique and duplicate roots. HP has done well by adding the ability to handle complex polynomial coefficients when it moved the PROOT to the graphing calculators.

References

1. Hewlett-Packard Company, "Math Pac Owner's Manual for the HP-71", Edition 2, September 1984.
2. Hewlett-Packard Company, "HP-71 Owner's Manual", Revision E, March 1987.
3. Hewlett-Packard Company, "HP-71 Reference Manual", Edition 4, October 1987.
4. Wikipedia, "The Laguerre's Method", March 2010.
5. W. M. Press et al, "Numerical Recipes: The Art of Scientific Computing", third edition, 2007. Cambridge University Press.
6. A. Ralston and P. Rabinowitz, "A First Course in Numerical Analysis", second edition, 1978, Dover Publications.
7. Hewlett-Packard Company, "HP 50g/49g+/48gII graphing calculator advanced user's reference manual", Edition 2, July 2009.
8. W. Kahan, "Personal Calculator Has Key to Solve Any Equation f(X)=0", Hewlett-Packard Journal, December 1979.

About the Author

Namir Shammas is a native of Baghdad, Iraq. He resides in Richmond, Virginia, USA. Namir graduated with a degree in Chemical Engineering. He received a master degree in Chemical engineering from the University of Michigan, Ann Arbor. He worked for a few years in the field of water treatment before focusing for 17 years on writing programming books and articles. Later he worked in corporate technical documentation. He is a big fan of HP calculators and collects many vintage models. His hobbies also include traveling, music, movies (especially French movies), chemistry, cosmology, Jungian psychology, mythology, statistics, and math. As a former PPC and CHHU member, Namir enjoys attending the HHC conferences. *Email me at: nshammas@aol.com*

Commas in the HP Calculator Display

HP Solve #22 page 34

← Previous **Article** – Next →

Commas in the HP Calculator Display

Richard J. Nelson

The comma delimiter in the display of an HP Calculator serves two basic purposes[(1)].

The first, and most important, purpose is differentiating the two parts of a number – the number and its decimal part, e.g. **1618033988,75**. US ***HP Solve*** readers will look twice at this because such a number looks strange to them. The comma used for this example is called the radix mark and the radix mark used in the US and many other English-speaking countries is a period. HP calculators have the feature of setting the radix mark to either a comma (used in Europe) or a period (usually the default).

The second basic purpose of having a comma in the display is as a thousands separator e.g. **1,618,033,988.75**. The thousands separator makes reading large numbers much easier. In fact trying to read some numbers without thousands separators is what inspired this article. See Fig. 1.

Fig. 1 – Comma delimiter use example.

Many HP calculators[(2)] prior to the introduction of RPL in mid 1986 with the HP28C would show thousands separators in the display. With the RPL change in the operating system the display control was changed and the only time you may see large numbers displayed with commas for the easiest reading is in the Fix display mode.

I asked Dr. William Wickes, one of the HP architects of RPL about this. Here is his reply. "The HP28C was HP's first multi-line RPN calculator, so the design team had to rethink the formatting of numbers when two or more numbers would be viewed on the stack at the same time. We considered STD mode by definition as meant for unformatted display of numbers. In an STD display of a multi-line stack of numbers, neither the decimal points nor the separators necessarily will line up vertically, so the display will be a bit of a jumble anyway and adding digit separators wouldn't improve matters. In FIX modes, there are always the same number of digits to the right of the decimal point; with right-justification of the numbers the vertical columns correspond to powers of ten and the separators line up nicely. So one would use FIX for a formatted, nicely aligned stack of numbers with consistent significant digits, or STD for an unformatted show-all-digits stack."

Because many of the current crop of HP's machines are based on the RPL operating system most newer HP calculator users don't have the nice feature of the thousands separators in the display. The HP48/49/50 series of machines will only display the thousands separators in the FIX mode. So will the HP35s.

There are current machine examples of the "old way" of using the thousands separators in the display. See Fig. 2 below.

10C 11C 12C 12Cpl 12Cpr 12C[25th] 15C 16C

Fig. 2 – The long reigning Voyager Series in model number order. The financial versions, HP-12C's, are current.

If you look closely you will see commas in the displays of these HP stock photos. The thousands separator commas are used by all Voyager series[(3)] calculators (sometimes called the '10 Series' or the 'Slimline Series'). Table 1 summarizes the radix mark and where it is most commonly used.

Table 1 – Radix and 1,000's Separator Usage Examples

Display Example	Radix[1]	Examples of where used[2]
1,618,033,988.75	Period[3]	USA, UK, South Africa, languages of Interlinguas and Esperanto, computer languages such as C, Java, and FORTRAN
1.618.033.988,75	Comma	Asia, France, Italy, (most of mainland Europe), and ISO international blueprints

[1] The thousands separator is "opposite" to the Radix mark.

[2] This is not an extensive or complete list. Some newer standards specify the use of a thin space (half space) as a group separator. Examples are SI/ISO 31-0 and the International Bureau of Weights and Measures.

[3] Default for most HP calculators.

Look again at the differences in the numbers shown in fig. 1. Which are easier to read? Did you immediately recognize that the two largest numbers are millions? There are three aspects to this issue.

1. The reading of numbers from the source that is reason for using the calculator. Reading the number is much easier and less error prone if the comma delimiter is used.

2. Keying in the number. Keying errors will be reduced if the display shows commas as you key them in as later the pre-RPL machines do.

3. The answer is easier to read during and after the calculation e.g.

$$1234 \times 5678 = 7006652 \quad \text{vs.} \quad 1{,}234 \times 5{,}678 = 7{,}006{,}652$$

If you usually calculate with a 10-digit finance calculator of the Voyager Series dealing with currency conversions that involve large values of a foreign currency, you really appreciate having the comma delimiters. If you use a 48/49/50 graphing calculator you have a 12-digit display, but you must use FIX mode to have the number displayed with commas. Most users of these machines keep their machine in STD mode and it is inconvenient to switch between modes just to read large numbers correctly.

Have you used more than one model of an HP calculator and noticed how the comma in the various calculators is used? Do you frequently use 8 – 12 digit numbers?

Do you think that entering and displaying numbers with thousands separators is important?

Do you want HP to bring back the thousands separator in the RPL based machines if it is practical? This would be best applied to single line displays.

Send your comments to the ***HP Solve*** editor at:

hpsolve@hp.com

Commas in the HP Calculator Display - Notes

(1) Ordered pairs (complex numbers) are shown, entered, with a comma between them on some machines. The comma is used worldwide as a separator between the real and the imaginary parts of a complex number, and there is a space after the comma and before the imaginary part, e.g. (3, 4).

(2) The HP Classic series did not provide this feature. The HP-71B in 1984 did not. Reader question: When were the display delimiters first added to HP calculators? Was it with the Spice/Spike series?

(3) To change (toggle) the radix on the Voyager machines press and hold the "." key when turning on the calculator. You may then see the two displays shown in Table 1.

HP's Calculator Manuals

HP Solve #22 page 37

← Previous **Article** – Next →

HP's Calculator Manuals

Richard J. Nelson

Legacy users of HP's Calculators usually remember the manuals that came with their machines in a positive way. Popular opinion seems to suggest that newer manuals are not as good as the older manuals, and certainly every manual user, old or new, seems to complain that there are never enough examples[1].

Most commonly the manual that describes the machine and how to use it is called the owner's manual. This is a generic term for a manual that comes with your car, refrigerator, or calculator. The owner's manual is assumed to contain the important information related to the product and how to use it.

HP didn't start out calling their calculator manuals, "Owner's Manuals", and they don't call them by that name today. The HP-35A (1972) came with an "*HP-35 operating manual*" (36 pp.) The lower case is HP's choice. The HP-65A (1974) came with an "*HP-65 Owner's Handbook*" (107 pp.) The HP-67A (1976) came with an "*HP-67 Owner's Handbook and Programming Guide*" (353 pp.). The HP-41C/41CV came with an "*Operating manual, A Guide for the Experienced User*" (71 pp.). The HP48G Series machines (1993) came with a "User's Guide" (592pp.). All of these manuals were printed, most spiral bound, and heavy. User's Guide makes sense because it is simple, short, and descriptive.

After 16 years of HP calculator manuals the eleven Pioneer Series of machines (1988-1991) came with an "*Owner's Manual*" (larger size, ~250 pp.). More recent machines now come with a "*User's Guide*" or "*Quick Start Guide*." The name has changed from manual to handbook to guide.

Another often-remembered "feature" of HP's manuals is the use of humor when illustrating examples that include people. Especially memorable are the unusual names for people used in the example. One reason that users tend to remember these things is that they desperately need examples in manuals. The "*PPC ROM Manual*" demonstrated this need by providing extensive examples for every one of the 153 routines described in the 500 page 8-1/2"x 11" (US letter size) 2-1/2 pound (1.1 Kilo) tome.

Why is this completely obvious aspect of the user's need ignored by HP when it produces its manuals? After over 35 years of writing about this issue, especially after producing the *PPC ROM Manual*, I believe that I know the answer. The resources required to include lots of examples for any publication is simply beyond the resources that are available. The *PPC ROM Manual* was able to provide the much-needed examples because of the donated 100+ man-years – 876,528 man-hours – that the User Community put into the project. Can you imagine HP, or any calculator manufacturer, spending these kinds of resources on any similar project? At a value of $20/hr. for the time, that would be $17.5 million dollars or $35,000 dollars per document page to produce the content. Even the world's largest technology company does not have that level of resources for a calculator product.

I mention these things because of the "chatter" often seen on the various HP web sites that users complain about HP's manuals. I don't disagree with many of these complaints because I have been writing about (and complaining about) HP's manuals for 35 years. I have probably met with (in person, face to face) most of the manual writers at HP for most of their machines.

Back in the days when HP had all of their operations – engineering, marketing, and manufacturing – in one location in Corvallis, I would visit the "factory" for three days meeting with many different teams dedicated to calculators. I especially remember one meeting with the documentation group (at a mutual request because of some of the things I had written[2]) that mentioned that they thought that their manuals were some of the best in the industry. After all, they had won awards for their manuals. Of course I was coming from a perspective of what is desired from the user's perspective and these experiences drove the decision to include extensive examples (and a formal format of section headings, extensive index, etc.) in the *PPC ROM Manual*.

Manuals for computer related products have always, and will always, fall short from the user's perspective simply because of ignorance, resources, complexity, cost, and time. The needs of the user don't change, it is the manufacturer's continuous attempts to competitively meet those needs that changes.

"If resources were unlimited", is a prerequisite that every complainer should use when writing about manuals. From the user's perspective I personally would like to see a manual approach similar to the following.

1. A **photograph** of the keyboard with each and every key identified with a description of what the notations and symbols mean.

2. A **key response description** of what happens when any key is pressed. Presenting this matrix of key responses will require that the user understand that keys change in their meanings depending on the mode or environment at the moment the key is pressed.

3. A **reference organization** that recognizes that the manual is used at least five times more frequently for reference than it is used for explanation (initial) reading.

4. **Each page is numbered**[2] sequentially. Bill Wickes addressed this issue using Section numbering AND page numbering in his famous trilogy of HP48 books. If you can't look at the last page in the manual and know the total number of pages in the document you don't understand how important page numbering is. Numbering individual sections may be convenient for the document writer, but it is certainly not convenient for the reader. The argument that it is easier to revise a section and not impact the whole document (very much) isn't justified because so few updates are ever produced.

5. **Examples** – practical, real world, timely, and meaningful - are important and this is discussed above.

6. **Consistent Writing Style**. Manuals should be written following a published style[3] that dictates what must be covered in an owner's manual, handbook, or guide.

7. A **two tier index**[4]. While examples are omitted because of necessity the lousy indexes in most calculator manuals are a classic example of simple ignorance. Using modern computer software it is an easy task to produce a document with an index index of 3 or higher (Number of index entries divided by the number of text pages). Every technical document that is to be used as a reference has at least three words, terms, or important ideas per page. The document must be proofed. Simply write down these index items with the page number as the proofing is being done. Word and most other document software have an index capability.

8. **Format.** Most of HP's recent User's Guides follow a "standard" format. Section one is usually "Getting Started" and the last part is "Appendixes and References." If the manual is printed it is a soft cover using a "perfect bound" style of binding.

HP calculator teams have reinvented themselves three times and while a historic review is interesting, how should the manual issue be viewed using today's business outsourced model? Many of you reading this have watched HP up close and personal during their active participation at HP Handheld

Conferences, HHC's, since 2002 when HP GM Fred Valdez broke the mold for HP User Community relationships. The record trend reversing attendance at HHC 2007 witnessed the changes that are taking place – slowly but consistently. A calculator GM, Wing, first met at HHC 2007 with the support of the post Carly president, Hurd, demonstrated that HP is changing in ways that brings a smile to the face of the legacy user. Product indicators of this very positive change are the HP50g and the HP35s. Regarding the latter, the HP35s manual is available to be down loaded at:

http://h10010.www1.hp.com/wwpc/pscmisc/vac/us/product_pdfs/user_guide.pdf

Printed manuals is still an HP challenge and there have been several solutions proposed to solve this "issue." After a long period of working on an agreement with at least one third party they could not agree to go forward. Business volume is the primary issue. I am so tempted to get into the HP printed manual business. Still, having a manual in a useful downloadable form as illustrated at the link above is most helpful. If you must have a printed copy you may print it yourself, or have it done at a copy store, and bind it using the method described in my HHC 2004 paper Personal Low Cost Binding System. It is clear that manuals will become ever smaller with the primary User's guide available on a CD and/or downloaded from HP's web site.

Manuals will always be a topic of discussion by users, young and old, ignorant and thoughtful, newbie and experienced. What is most important, however, is to understand the issues involved, sharing your desires with HP, and being prepared to reach ever deeper into your wallet for printed good quality manuals, handbooks, or guides. What is missing in HP's manuals? How could the manuals be improved.? Send your comments to:

Email ***HP Solve*** at: hpsolve@hp.com

Notes:

1. *Technology changes the way we use information. Manuals are provided on CD's. Examples, as mentioned here, are usually associated with a manual; printed, downloaded, or provided by HP on a CD. What is an example? It is a step by step procedure, process, or algorithm that shows you how to solve a problem or use a process. Today HP is using video technology in the form of Training Guides for their machines The Training Guides may be found on their web site and they are being produced by experienced users of their machines.*

2. *One of the comments I had written regarding poor HP documentation was the fact that at least the pages should be numbered on all documents over four pages in length. I was severely taken to task for saying that HP didn't number their pages. I then mentioned a 72 page document that wasn't page numbered. They appeared to be shocked when I pulled a copy from my briefcase*

3. *When I asked if an HP Style Manual existed most of the people in the room were hesitant to talk about it. They implied that they did, but when I asked if I could get a copy I was stone walled. Privately a writer later explained that each writer used his or her own references, and that they did not have an official HP Style Manual. This older HP confident writer explained that there was an official HP Style manual, but it hadn't been updated in many years – at that time – and the younger writers didn't know about it. This was during the CVD days and much has changed. Perhaps there is an HP Style Manual being required at HP today. My CVD photocopy really looks old.*

4. *In the famous numerical calculation philosophy of Prof. William Kahn there are avoidable errors and unavoidable errors. Perhaps the lack of examples could be justified to be in the latter category, but having a poor index is most certainly in the former category. An index value of three or higher is most desirable and it greatly extends the usefulness of the User's Guide at a very low production cost.*

Fundamentals of Applied Math Series #5

HP Solve #22 page 41

← Previous **Article**

Logs

Richard J. Nelson

Introduction – What are logs?

Fig. 1 – LOG Keys.

Which item does not belong on this list: Amplifier, Earth quake, Antenna, Loud noise, Catenary, Redwood, Slide rule, Stellar brightness, Weber's law. Hint: Think about Fig. 2.

Fig. 2 – A precursor to learning about logs? [3].

Other items that should not be on the list[1] are: Holland's Rule, Fabian's Rule, and The Bangor Rule.

Log is short for logarithm[2]. Most students learn about logarithms as part of their algebra class when they study powers, e.g. Y^X.

Many algebraic and RPN, calculators have a key marked Y^X. More strongly algebraic oriented calculators such as the HP39gs or HP40gs will calculate powers with a X^Y key. When we use the power key for logs we assume that we are working with normal or common numbers, i.e. base ten numbers. See Table 1 below.

Table 1 - Powers of Ten

Y	X	Y^X
10	0	1
10	1	10
10	2	100
10	3	1000
base	log	number

From a practical perspective logarithms are most often used to express numbers that span a very large range of values. Table 1 illustrates how logarithms are related to the power of a number. Logarithms having a base of ten are called common logarithms simply expressed as log. More on the name later. From table 1 the log of 100 is 2, the log of 10 is 1, and the log of 1 is 0. The log value is the exponent of the power of ten that the number must be raised to. The log of 3 must be between 0 and 1.

Calculating the log of 3 is a difficult task and most people couldn't do it. We didn't learn this procedure in school when we learned to multiply and divide numbers. Fortunately we have calculators that are able to do this with a key identified as .

Calculating logs: To calculate the log of a number use the LOG key. To calculate the inverse or antilog use the 10^X key. See figure one above. **(1)**

The log of 3 is approximately 0.47712125472. Enter 10 and 0.47712125472 and use the power key to verify that $10^{0.47712125472}$ is indeed 3. The log of a base 10 number is the exponent of ten that represents the number.

---

1. *According to a USDA Forrest Service General Technical Report there are at least 95 log (timber) rules bearing 185 different names.*
2. *Logarithm is a term coined by John Napier of Merchistoun (1550 – 1617). His 1614 book "Mirifici Logarithmorum Canonis Descripti," provided log tables and methods for numerical calculation.*
3. *Mathematics building of the University of Maryland.*

Logarithm: The log value is the exponent of the power of ten that the number must be raised to. The log of 3 is approximately 0.47712125472 which means that $10^{0.47712125472}$ is equal to 3. Only positive real numbers have real-valued logarithms. **(2)**

Where logs are used

One of the classical applications of logarithms is the slide rule. The primary means of making calculations for over 350 years (the early 1600's to the mid 1970's) were logarithmic tables. Three place log tables were printed onto a convenient slide rule "calculator." This device was the standard means of making numerical computations before the first scientific calculator; the HP-35A, appeared in January 1972. See figure 3a below.

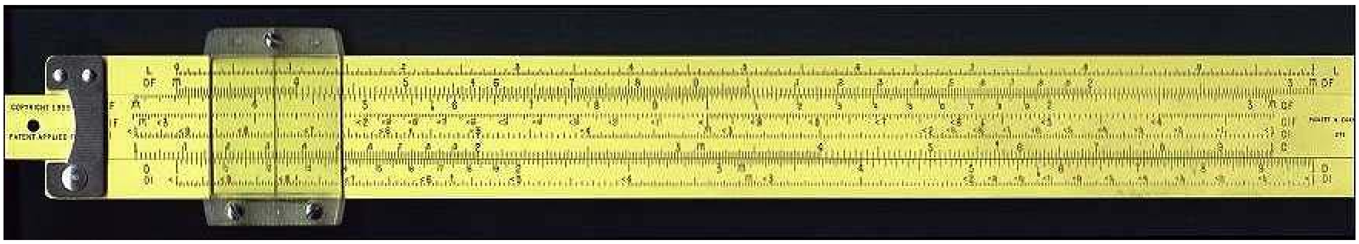


Fig. 3a – Typical slide rule used by science students and engineers prior to the mid 1970's.

The slide rule[4] uses scales that are spaced according to the logarithm of the scale value.

The image in figure three "b" was cropped from the photograph of a slide rule (fig. 3a) and printed landscape for maximum size. The C (and D) scale values (used to multiply numbers) were measured and compared with the log of their respective scale values (a decimal number) and multiplied by 259.

These values corresponded to the printed image measured values (as shown) to two decimal places plus or minus 2 counts. In most cases they were identical to three places. I just didn't take the time to measure to a fraction of a millimeter. For example, the scale length for the number "2" on the scale is 78.0 mm. The ratio of 78.0 mm/259 mm is 0.301. The log of 2 is approximately 0.301029995664.

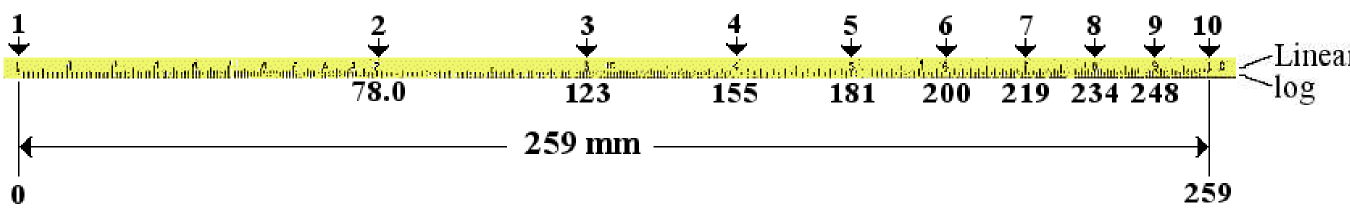


Fig. 3b – C slide rule scale of figure 3a cropped and annotated with printed image measurements in millimeters.

Another example of how logarithms are used is to plot data using a log scale instead of the normal linear scale. The slide rule scale of fig. 3b shows how linear & log scales compare. If the linear scale were used half the length, (5 of 10) of the 259 mm it would be 129.5 mm. On the log scale the corresponding value of 5 is 181 mm. Note how the log scale tends to expand the lower values and compresses the higher values (in each decade) in terms of scale lengths.

From this exercise it is clear that the distance spacing of the slide rule numbers is based on the logarithm and proportioned to the total scale length. Many slide rules have 10 inch scales. The spacing of "2" is the

- -

4. Personal note. When I used my slide rule in high school I often wondered why someone didn't make the scales on motor driven Mylar tape that could be very long and greatly increase the number of calculated digits. How many digits would be readable if a nine foot scale (10 times longer than the average slide rule) were used?

log of "2" and equal to 3.01 inches, "5" is 6.99 inches, etc. Note that the half value of the decade range is about 70 percent of the of the scale length. The greatest difference is that the first 10% of the scale value which is 30.1% of the scale length. This property of logs may be used to advantage in many situations.

Here is an another example of logarithmic scaling. An experiment was performed to determine the growth rate of crickets and the following data was recorded from a starting pair of crickets under ideal conditions. See table two.

Table 2 – Cricket Growth from a Single Pair

Start	30 Days	60 Days	90 Days	120 Days	150 Days	180 Days
2 (one pair)	2	10	20	100	1,000	10,000

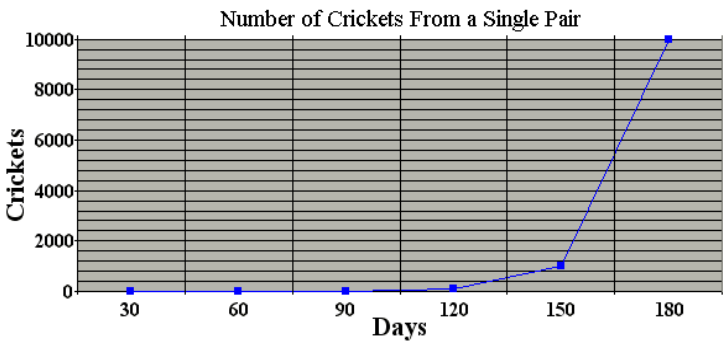


Fig. 4 – Cricket growth plotted using a linear "Y" scale.

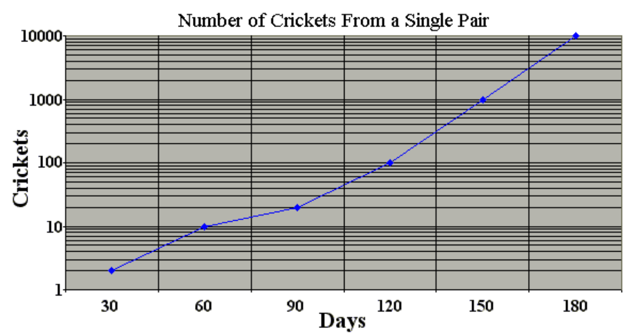


Fig. 5 - Cricket growth plotted using a log "Y" scale.

The purpose of any plot is to provide a meaningful visual representation of the data and the log scale used for the cricket count does a better job than the linear scale. See figures four and five. Note that the cricket plots are very close to the same size vertically. Which better represents the data? Benford's Law[5] is a related reason this is especially useful for easier data plotting and reading.

An additional consideration for a useful plot, aside from the ability to read the data values from the plot, is to be able to extend the plot (predict values) when additional data is not available. Figure six on the next page shows all four combinations of linear and logarithmic x and y scales for three function plots. The choice of scales will strongly influence the shape of the plot. It is easier and more accurate to extend a straight line plot than a curved line plot.

Electronics semiconductor components will often have specifications plotted using a semi-log or log-log plot. Figure seven shows three examples. Ohm's law, i.e. I = E/R, assumes that the resistance doesn't change with current or applied voltage. Figure 7a shows that the resistance of a semiconductor does effectively change with a change in current because the voltage drop changes.

Logarithms have applications in fields as diverse as astronomy, chemistry, computer science, economics, engineering, music, physics, and statistics.

Logarithmic scale: A scale of measurement in which an increase or decrease of one unit represents a tenfold increase or decrease in the quantity measured.	**(3)**

- -

*5. See One Minute Marvels in **HP Solve** Volume 15 for an illustration of Benford's law which describes the distribution of numbers used for "natural" data.*

- Bels, decibels, and nepers are used to measure sound intensity because the ear responds approximately logarithmically to sound pressure. If you need to adjust something related to sound intensity (loudness) you should do it at the lowest possible level in order to hear and detect the smallest change. Electronics technicians learn this when adjusting tuned circuits that carry an audio signal.
- Amplifiers and antennas use a ratio of logs to calculate the Db gain value.
- A logarithmic scale is used for pH measurements.
- The Richter scale measures earthquake intensity using a logarithmic scale.
- Logarithms are used in information theory as a measure of quantity of information.

Continued on page 5.

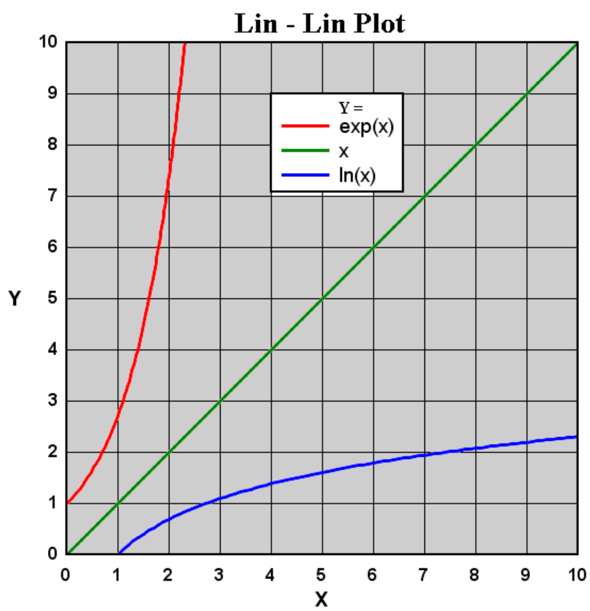


Fig. 6a – Standard linear plot – Linear Y & X.

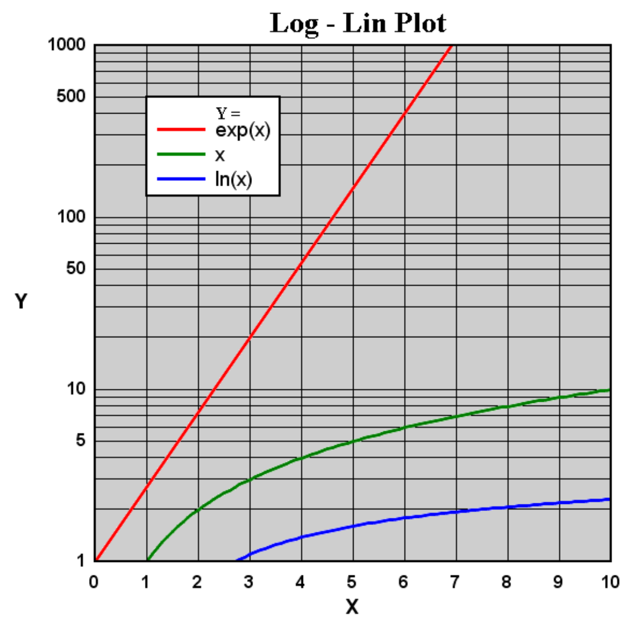


Fig. 6b – Y axis Log and X axis linear (semi-log).

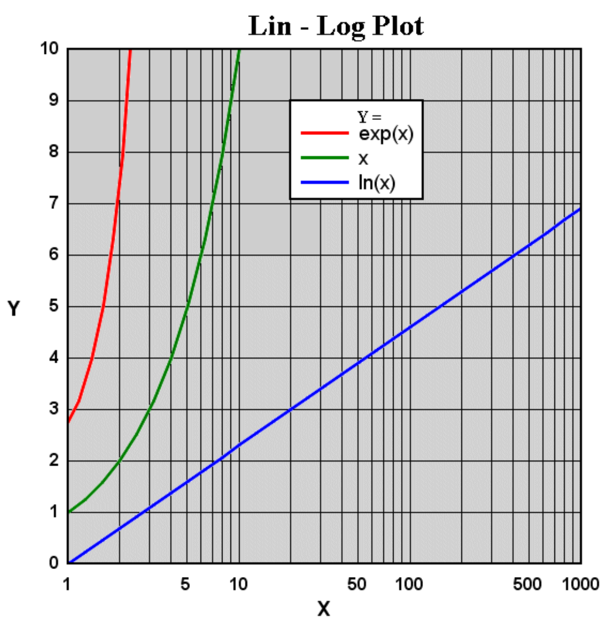


Fig. 6c – Y axis Linear and X axis Log (semi-log).

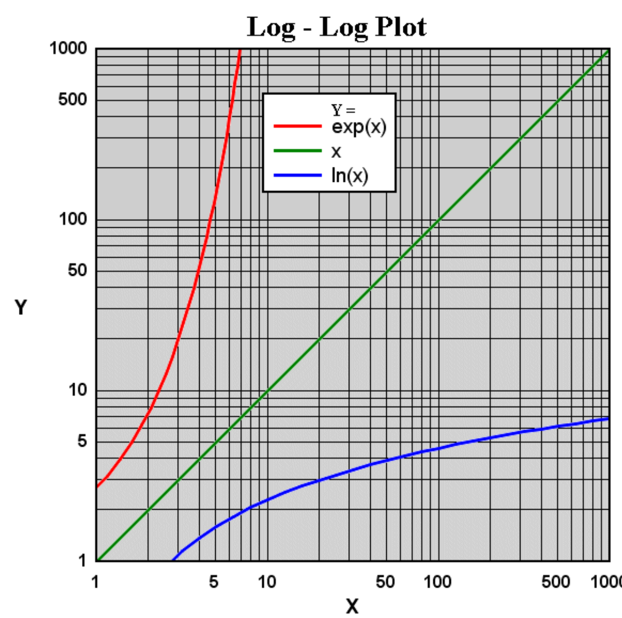


Fig. 6d – Standard log-log plot .

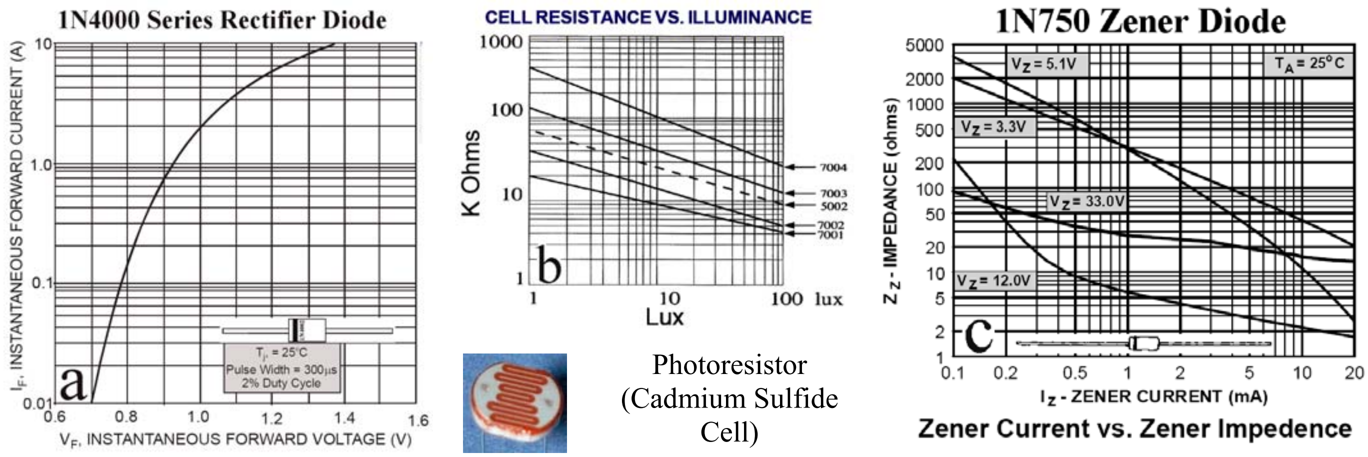


Fig. 7 – Electronic semiconductor component specifications frequently utilize log and semi-log scales.

- The apparent magnitude of stars measures the brightness logarithmically, because the eye responds approximately logarithmically to brightness.
- Semitones in music (e.g. music intervals cent, minor second, major second, and octave) are measured logarithmically.
- Compounding of interest uses logarithmic relationships.
- Radioactive decay is measured logarithmically.
- Measuring the efficiency of computer algorithms (especially sorting) is done logarithmically.
- Logarithms are used to describe the fractional dimensions for fractals.

Figure eight is a plot of the log of x. This is a similar version of the plot shown in figure 5a (see next section and box 8 below for the differences) and it better shows the values of the log function between zero and one.

Will the blue curve cross +1 at x = 10? What is the value of the curve at -2? What would be a better set of scales for this plot? *Hint: See similar blue plot of figure six "c."* The curve is not very clear for the value of -2. What is the antilog value of -2? What is the value at -10? Does the anti-log of x ever reach zero?

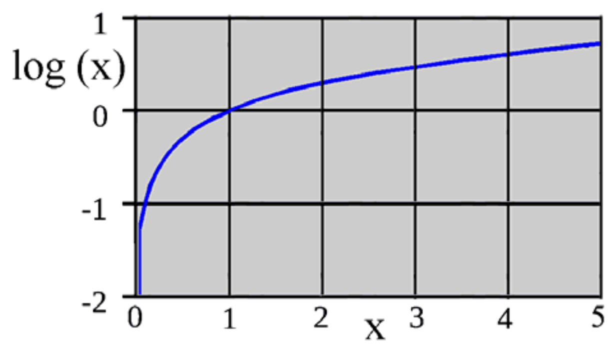


Fig. 8 – Log(x) plot for values near zero.

Logs that use different bases

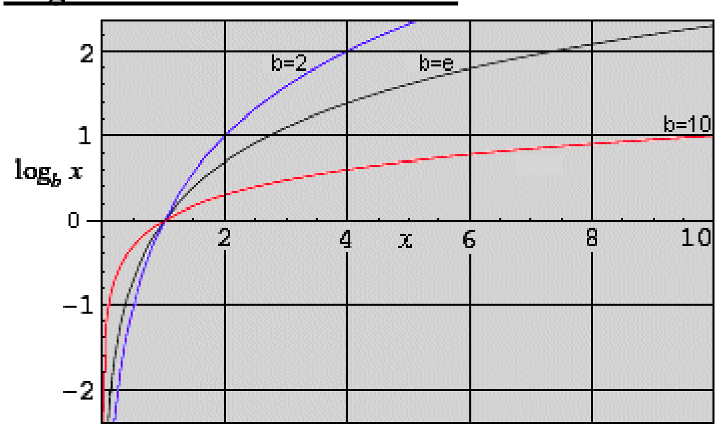


Fig. 9 – Plot of logs of three of the most commonly used bases.

If n = base^{Log}, the base may be a number that provides convenience or mathematical meaning. The logs of base ten numbers have been used to illustrate how logs are used for computation, measurement values, and plot scaling.

Figure nine shows a plot of x vs. log (x) for three different bases for comparison. The red plot is to base 10, the black plot is to base e (see footnote 6) **,** and the blue plot is to base 2. Logarithms of all bases pass through the point

(1,0), because any non-zero number raised to the power 0 is 1. The plots also pass through the points (b,1) for base b, because a number raised to the power 1 is itself. All the curves approach the y-axis, but do not reach it because at x=0 the value is a vertical asymptote. Computers use base 2 numbers and logs that use two as the base are quite common.

As figure nine shows the three most common logarithmic bases are 2, e, and 10. Logarithms to the base e are called Naperian or natural logarithms for their inventor John Napier (1550-1617) using the notation LN. Most scientific calculators have natural log and anti-log keys. An example is the two HP35s keys as shown in figure 10. These two keys, with the six functions shown, are the most mathematically power packed keys on the HP 35s keyboard.

Fig. 10 – HP35s LN keys.

The three most common logarithmic bases are 2, e, and 10.	**(4)**

Natural logs: The natural logarithm of a number *x* is the power to which *e* would have to be raised to equal *x*. To calculate the natural log, base e, of a number use the LN key. To calculate the inverse or antilog use the e^X key. See figure ten above.	**(5)**

Logarithm notations

All during our discussions of logarithms we have been using traditional calculator keyboard notations, i.e. log x is base ten and LN x is base e. Every technical field has its own notations and those within a given field may use log x to be LN x as an understanding among themselves. For example:

- Engineers, biologists, and astronomers often define "log(x)" to be the common logarithm, $\log_{10}(x)$.
- Mathematicians generally define "log(x)" to be the natural logarithm, $\log_e(x)$.
- Computer scientists often choose "log(x)" to be the binary logarithm, $\log_2(x)$.
- In most commonly used computer programming languages the "log" function returns the natural logarithm, $\log_e(x)$.

In an attempt to avoid logarithmic notation confusion the US Department of Commerce National Institute of Standards and Technology, NIST, recommends to follow the International Organization for Standardization, ISO, standard titled *Mathematical signs and symbols for use in physical sciences and technology, IS 31-11:1992.* The standard suggests these notations:

Logarithmic ISO notations: The notation "lb(x)" means $\log_2(x)$
The notation "ln(x)" means $\log_e(x)$
The notation "lg(x)" means $\log_{10}(x)$ **(6)**

Calculator keyboard notation usage is not expected to change, however, and the reader should be aware of other notations being used for logs. Perhaps HP could add base 2 log keys and then change the keyboard notations to comply with international standards.

- -

6. The constant e is known as Euler's number which has a value of 2.71828182846… *This number appears throughout mathematics and its discussion is beyond the scope of this review. Euler is also known for Euler's identity, named after Leonhard Euler, is the equality:* $e^{i\pi} + 1 = 0$. *It is considered the most beautiful theorem in all of mathematics. The reader may learn more at:* http://en.wikipedia.org/wiki/Euler%27s_identity

Let's do a mental experiment: you have a calculator key that is supposed to be a logarithm key, but the base is unknown. How would you determine its base?

Logarithm Base Determination		
1. Key a number, press ENTER twice.	Try 153.	
2. Press the unknown log key.	(use LN) = 5.03043792139	**(7)**
3. Press the 1/x key.	0.198789850034	
4. Press the Y^x key.	2.71828182846	

This method may also be used to test a programming language log function to verify the base.

You could save steps (assuming you also had an anti-log key for the unknown base log key) by pressing 1 and the anti-log key, e.g. 1, e^x = 2.71828182846; 1, 10^x = 10. If you had a base two log key you would press 1 and the log key to return 2.

The mathematics (equalities) of logarithms

The equations that follow may be found in most math text or reference books and only the basic equalities are given for reference. Proofs and derivations are not included here.

Logarithmic relationships apply to all logarithms regardless of the base because they are mathematically similar. **(8)**

Logarithms are especially useful in solving equations in which exponents are unknown. For the equation $b^n = x$; b can be determined with radicals, n can be determined with logarithms, and x can be determined with exponentials.

Logarithm tables were calculated by John Napier, Henry Briggs, Jost Burgi, Adriaan Vlacq, Jurij Vega, Leonhard Euler, François Callet, and Gaspard de Prony in the early 1600's to make calculations easier.

Multiplication:

To multiply two numbers take the anti-log of the sum of their logs. $\log(cd) = \log(c) + \log(d)$ ***(e1)***

Division:

To divide two numbers take the anti-log of the difference of their logs. $\log(c/d) = \log(c) - \log(d)$ ***(e2)***

Raise to a power:

To raise a number to a power take the anti-log of the power multiplied by the log of the number;

$$\log(c^d) = d\log(c) \qquad (e3)$$

Extracting a root:

To extract the d[th] root of a number c take the anti-log of the log of the number divided by the desired root;

$$\log(\sqrt[d]{c}) = \log(c)/d \qquad (e4)$$

Combining the division:

$$\log(\log(\sqrt[d]{c})) = \log(\log(c)) - \log(d) \qquad (e5)$$

Changing the base (base c to base a)

$$\log_a b = \frac{\log_c b}{\log_c a} \qquad (e6)$$

Equation 6 (*e6*) is useful to evaluate logarithms using other bases on calculators. Most calculators have LOG and LN keys, but none for $\log_2$. To find $\log_2(153)$, you calculate LOG(153) / LOG(2) = 7.2573878427. As reference box 8 reminds us the LN keys may also be used in place of the LOG keys.

Log of 1 = 0 regardless of the base. $\log_b(1) = 0$ ***(e7)***

Log of base = 1 $\log_b(b) = 1$ ***(e8)***

Logarithms and exponentials are inverse operations similar to multiplication and division.

$$b^{\log(x)} = x, \text{ and } \log_b(b^x) = x \qquad \textbf{(9)}$$

References (links) for logarithms

1. Calculator for calculating logarithms of any base: (see figure 11) http://rechneronline.de/logarithm/

2, Euler's identity: http://en.wikipedia.org/wiki/Euler%27s_identity

3. Excellent free 109 page logarithm reference in a PDF file: http://www.mathlogarithms.com/

4. Also see the errata for #3 printed version at: http://www.mathlogarithms.com/images/ErrataBoundCopyMay2008.pdf

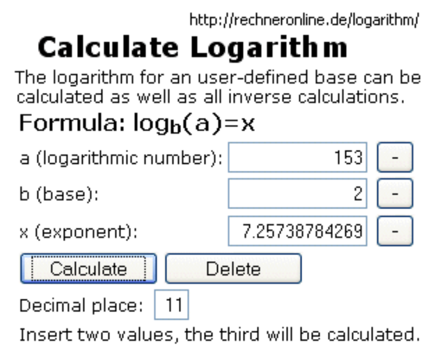


Fig. 11 – Website Log calculator.

Advanced topics

For the reader who wants to expand their logarithmic knowledge further here are a few terms that are suitable to search on the internet.

Complex logarithm
Discrete logarithm (theory of finite groups, and difficult to calculate)
Double logarithm (iterated logarithm)
Imaginary-base logarithm
Indefinite logarithm
Iterated logarithm
Log-normal distribution
Logarithms of complex numbers
Logarithm of a matrix is the inverse of the matrix exponential.
Logarithm of a quaternion
Logarithm of a octonion
logarithm derivative
Richter Scale
Super logarithm
Weber's law
Zech's logarithms

Reader challenge. Suppose you take the average of a set of logarithmic values. What is the antilog of this result called?

The average of the logarithmic values, converted back to a base 10 number is the geometric mean.

Summary and conclusion

A logarithm is a simple concept that most people have heard of, or remember from high school. The log of a number, x, is the exponent value, p, the base, b, is raised. For example; log 100_{10} = 2 because 10^2 = 100. In general terms; Log x = b^p. The log values of numbers are used for computation convenience, measurement values, and plot scaling. Examples of each of these primary applications are provided.

The three most common bases are 2, e, and 10. Logarithms are used in fields as diverse as astronomy, chemistry, computer science, economics, engineering, music, physics, and statistics. Logarithms are especially useful for dealing with exponents in equations. The normal textbook mathematics (equalities) of logarithms is also described. Reference links (including an Internet calculator) and a research topics list are provided for further study.