

HP Forum Archive 15

[[Return to Index](#) | [Top of Index](#)]

Sample problems to evaluate RPL vs. RPN "objectively"

Message #1 Posted by [Gene](#) on 21 Feb 2006, 5:20 p.m.

I'd like to ask for input from our community here to see if there are some sample problems we could pose that would allow us to evaluate the effectiveness and efficiency of RPL vs. 4-level stack RPN.

For example, suppose we said one of the problems was to compute the roots to a quadratic equation given a, b, and c in the stack.

I'd like to see the RPN approach and the RPL approach to see if they can be remotely objectively evaluated for clarity, efficiency, and general usefulness to the average (NOTE THAT) user. :-)

I know that this is not a good sample problem given the built in quadratic function on the 49g+, but I'd like to see if any here have ideas for good sample problems.

It might be that we should have a set of problems that are RPN masterpieces vs. those that are easy to do in RPL but not RPN.

Thoughts? Gene

Re: Sample problems to evaluate RPL vs. RPN "objectively"

Message #2 Posted by [Howard Owen](#) on 21 Feb 2006, 6:25 p.m.,
in response to message #1 by Gene

I think it is interesting that you are asking the question. 8)

Edited: 21 Feb 2006, 6:26 p.m.

Re: Sample problems to evaluate RPL vs. RPN "objectively"

Message #3 Posted by [Gerson W. Barbosa](#) on 21 Feb 2006, 7:54 p.m.,
in response to message #1 by Gene

Hi Gene,

I fear the example below for the HP-15C is not what you'd expect because an RPN (or RPL) program using only the stack is far from meeting the 'clarify' requirement. Nevertheless, I think it meets the 'efficiency' requirement. Well, it is hard to have both.

Furthermore, the HP-15C is an exceptional calculator. Perhaps I should have written an HP-11C version (or even an HP-12C one!). That's one of the programs that are written in the last blank pages of my HP-15C manual. I wrote it in 1984 when I knew RPN even less than I know today.

The RPL version would be straightforward and quite easy to read.

Regards,

Gerson.

```

001- f LBL E
002- g CF 8      ; complex mode off
003- 4
004- *
005- g R^
006- *
007- CHS
008- x<>y
009- ENTER
010- Rv
011- g x^2
012- +
013- g TEST 2   ; x<0
014- g SF 8     ; if delta <0 then set complex mode on
015- SQR T
016- g R^
017- +
018- 2
019- /
020- x<>y
021- ENTER
022- Rv
023- /
024- 1
025- CHS
026- *
```

027- ENTER
028- g R^
029- g R^
030- x<>y
031- /
032- +
033- 1
034- CHS
035- *
036- g RTN

Assuming a, b and c in the stack, after running the program the stack will contain x1 and x2. Steps 24, 26, 33 and 35 are only necessary because CHS in complex mode does not change the sign of the argument (a bug or because of a reason unknown to me?). To achieve this it is necessary to multiply the argument by -1. The user will know the roots are complex because of the 'C' indicator.

Re: Sample problems to evaluate RPL vs. RPN "objectively"

Message #4 Posted by [Palmer O. Hanson, Jr.](#) on 21 Feb 2006, 10:52 p.m.,
in response to message #3 by Gerson W. Barbosa

I think that we need to agree on what "effectiveness" and "efficiency" mean and how we separate out the characteristics of the machine from the characteristics of the algorithm. For example, pages 205ff of the *HP-15C Advanced Functions Handbook* defines the quadratic as $c - 2bz + az^2 = 0$ which allows some efficiency in calculation at the cost of defining a quadratic in a way which is different from the traditional way. For effectiveness the 39 step routine on pages 205-206 of the *HP-15C Advanced Functions Handbook* uses a different algorithm which avoids destructive cancellation for difficult problems.

What if you used the old Mach Number problem?

Re: Sample problems to evaluate RPL vs. RPN "objectively"

Message #5 Posted by [Gerson W. Barbosa](#) on 22 Feb 2006, 8:18 a.m.,
in response to message #4 by Palmer O. Hanson, Jr.

You are right. I remember I saw the quadratic solver on the HP-15C Advanced Functions Handbook mentioned here some time ago and how it handled difficult examples. Too bad at the time the complementary handbook was not available here, neither was it available in Portuguese I think. I need to remember to have the 15C Advanced Functions on the Museum DVD printed out.

The difference between "effectiveness" and "efficiency" is something I still have to learn. I was asked this at least a couple of times at college but I never remember the subtleties. ('Eficiência' vs. 'Eficácia' - Prof. Vieira?)

Re: Sample problems to evaluate RPL vs. RPN "objectively"

Message #6 Posted by [Valentin Albillo](#) on 22 Feb 2006, 7:57 a.m.,
in response to message #3 by Gerson W. Barbosa

Hi, Gerson:

Gerson posted:

"[...] CHS in complex mode does not change the sign of the argument (a bug or because of a reason unknown to me?)."

More the latter than the former. Have a look at pp 124-125, "*Changing Signs*" of *Section 11: "Calculating With Complex Numbers"* of the *HP-15C Owner's Handbook* where your "bug" is discussed in gory detail.

There's an acronym for this, IIRC, something like **RTFM** ... :-)

Best regards from V.

Re: Sample problems to evaluate RPL vs. RPN "objectively"

Message #7 Posted by [Gerson W. Barbosa](#) on 22 Feb 2006, 10:35 a.m.,
in response to message #6 by Valentin Albillo

Hi Valentin,

Yes, as I have recognized in an earlier post, I should have read the other handbook. But, in this case, the answer seems to be in the standard manual. I have surely read this section already, I just may have forgotten about the content.

Anyway, there's no problem in reminding me this: I always translate **RTFM** as *Read the Fine Manual*, which, in the case of HP manuals, is always true! :-)

Best regards,

Gerson.

Re: Sample problems to evaluate RPL vs. RPN "objectively"

Message #8 Posted by [Valentin Albillo](#) on 22 Feb 2006, 10:39 a.m.,
in response to message #7 by Gerson W. Barbosa

Hi again, Gerson:

Gerson posted:

"I always translate RTFM as Read the Fine Manual"

Of course, of course ! Is there any other way to translate this !? ;-))

Best regards from V.

Re: Sample problems to evaluate RPL vs. RPN "objectively"

Message #9 Posted by [Les Bell](#) on 21 Feb 2006, 7:59 p.m.,
in response to message #1 by Gene

The one that interests me, and I've been trying to find time to write up, is section 9-6 of "The Feynman Lectures on Physics". This is simply a numerical solution of a first-order differential equation (mass on spring) and produces Table 9-1, which lists displacement, velocity and acceleration against time.

I've done it on the 41, and I've done it in Perl. I haven't had time to tackle it on the 48 yet - at least, numerically/programmatically. On the other hand, I was able to enter the equation into the differential equation solver on the 48, and get a nice graph, too.

Which method is better probably depends on why you are solving the problem. The goal might be a numerical solution to the problem, or it might be to develop some intuition as to what is going on in the physical system.

Which approach works better for which goal is left as an exercise for the reader.

;))

Best,

--- Les

[<http://www.lesbell.com.au>]

Re: Sample problems to evaluate RPL vs. RPN "objectively"

Message #10 Posted by [James M. Prange \(Michigan\)](#) on 21 Feb 2006, 10:25 p.m.,
in response to message #1 by Gene

Does this "average" user ordinarily use Classic RPN, or RPL, or perhaps neither?

Regards,
James

Re: Sample problems to evaluate RPL vs. RPN "objectively"

Message #11 Posted by [Gene Wright](#) on 21 Feb 2006, 10:43 p.m.,
in response to message #10 by James M. Prange (Michigan)

Well, "average" user is tough to define.

I can write/use 4-level RPN and RPL, but there are times when I'm much happier using a normal RPN machine than an RPL one.

So, I'm just wondering if we can come up with some sample problems that really, really point out the differences between them.

What are some areas where the 4-level stack / classic RPN really shows off?

What are some areas/problems where the RPL system would be very difficult to do in the 4-level stack.

We can argue about it all day, but I'm wondering if we can come up with some good illustrations to help.

Evaluating RPL vs. RPN: Area between two curves

Message #12 Posted by [Karl Schneider](#) on 22 Feb 2006, 2:01 a.m.,
in response to message #1 by Gene

Hello, Gene --

A very illustrative example, IMO, was discussed two years ago:

[Area between two curves](#)

This problem was revisited in a stupendously lengthy thread that ran during November 2005 in Archive #15.

Regards,

-- KS

Edited: 22 Feb 2006, 9:39 p.m.

Re: Sample problems to evaluate RPL vs. RPN "objectively"

Message #13 Posted by [Thomas Radtke](#) on 22 Feb 2006, 3:27 a.m.,
in response to message #1 by Gene

Quote:

I'd like to ask for input from our community here to see if there are some sample problems we could pose that would allow us to evaluate the effectiveness and efficiency of RPL vs. 4-level stack RPN.

I have a very simple example: Add I% interest of A to A ($=A(1+I\%)$). That is on RPN

```
A
ENTER
I
%
+
```

while on RPL, it is required to do something like

```
A
ENTER
ENTER
I
%
+
```

See? You save one keypress, that is nearly 17% of the total work. If you do nothing else 8 hours a workday, you save more than one hour per day. Clearly, this shows the superiority of RPN;).

Thomas

Re: Sample problems to evaluate RPL vs. RPN "objectively"

Message #14 Posted by [Palmer O. Hanson, Jr.](#) on 22 Feb 2006, 4:08 a.m.,
in response to message #13 by Thomas Radtke

The interest problem is a terrible example. It says that if you are a key punch operator then RPN is clearly the way to go. What if you are an engineer?

Re: Sample problems to evaluate RPL vs. RPN "objectively"

Message #15 Posted by [Thomas Radtke](#) on 22 Feb 2006, 9:30 a.m.,
in response to message #14 by Palmer O. Hanson, Jr.

Quote:

What if you are an engineer?

Then I would have the insight that $Q=1+I\%$ can easily be calculated in mind and end up with typing

```
A
ENTER
Q
*
```

which works on both systems and is not only as much as effective but also highly efficient;P.

(My post was probably not the most serious response to Gene's request, sorry for that:).

Thomas

Re: Sample problems to evaluate RPL vs. RPN "objectively"

Message #16 Posted by [Han](#) on 22 Feb 2006, 2:39 p.m.,
in response to message #13 by Thomas Radtke

I only recently read about the differences between RPN and RPL (and by no stretch am an expert in either). I think that the example below is not quite 'objective' enough. For one, not all calculators have a key dedicated to the % operation. On some implementations of RPL (for example, the HP48 series), there are commands that are accessed via menus.

Taking a look at just the keyboard layout of the 41C (I've never used one myself), I do not see a % operation there. Presumably, this is accessible either by menu or catalog of some sort. Likewise, the DUP command on an HP48 can be accessed via the stack operations menu. Thus A ENTER ENTER I % + could easily be A DUP I % + where DUP is from the menu key. Comparisons in terms of the number of keystrokes (because this will be highly dependent on the keyboard layouts of the implementations of RPN and RPL) might not be the most fair comparison. In fact, comparing RPN on one machine with RPN on a different machine may produce a different number of keystrokes.

Also, when doing routine calculations, we're more likely to write a program to do the repetitive tasks.

Quote:

I have a very simple example: Add I% interest of A to A (=A(1+I%)). That is on RPN

A
ENTER
I
%
+

while on RPL, it is required to do something like

A
ENTER
ENTER
I
%
+

See? You save one keypress, that is nearly 17% of the total work. If you do nothing else 8 hours a workday, you save more than one hour per day. Clearly, this shows the superiority of RPN;).

Thomas

Edited: 22 Feb 2006, 2:43 p.m.

Re: Sample problems to evaluate RPL vs. RPN "objectively"

Message #17 Posted by [Massimo Gnerucci \(Italy\)](#) on 22 Feb 2006, 11:32 p.m.,
in response to message #16 by Han

Quote:

Taking a look at just the keyboard layout of the 41C (I've never used one myself), I do not see a % operation there.

[shift]Rolldown (G key)

Greetings,
Massimo

Re: Sample problems to evaluate RPL vs. RPN "objectively"

Message #18 Posted by [Thomas Radtke](#) on 23 Feb 2006, 4:08 a.m.,
in response to message #16 by Han

Quote:

For one, not all calculators have a key dedicated to the % operation. On some implementations of RPL (for example, the HP48 series), there are commands that are accessed via menus.

In that case, the function key to which % is assigned to becomes functionally a % key, right?

Quote:

Likewise, the DUP command on an HP48 can be accessed via the stack operations menu.

That would mean to switch between stack and math/real menus all the time. Effective but not very efficient:).

Quote:

Also, when doing routine calculations, we're more likely to write a program to do the repetitive tasks.

I dare to say that you can overcome nearly any shortcoming of one of the systems by programming a handy shortcut. But you actually have to do this and you have to take that work into account when estimating something like 'efficiency'.

Re: Sample problems to evaluate RPL vs. RPN "objectively"

Message #19 Posted by [Valentin Albillo](#) on 22 Feb 2006, 6:48 a.m.,
in response to message #1 by Gene

Hi, Gene:

Gene posted:

"I'd like to ask for input from our community here to see if there are some sample problems we could pose that would allow us to evaluate the effectiveness and efficiency of RPL vs. 4-level stack RPN."

Though interesting, I'd say this is an exercise in futility, because the RPN vs. RPL topic, which has been discussed *ad nauseam*, probably belongs to the same class as politics, religion, nationality, local mores, etc. The ones who idolize one will despise the other no matter how many examples you care to provide or no matter how rationally you want to discuss the subject, and you're certain to make no converts no matter how good your arguments. For each example favorable to one of them, someone will provide an example favorable to the other, and at the end we all will agree to disagree, if at that.

"For example, suppose we said one of the problems was to compute the roots to a quadratic equation given a , b , and c in the stack. I'd like to see the RPN approach and the RPL approach to see if they can be remotely objectively evaluated for clarity, efficiency, and general usefulness to the average (NOTE THAT) user: :-)"

Actually, irrelevant. The average user would be best served by the simple HP-71B version (or any vintage SHARP handheld for that matter)

$$D=\text{SQR}(B*B-4*A*C) @ (-B+D)/2/A, (-B-D)/2/A$$

which can run rings around any RPN or RPL version in terms of efficiency and simplicity, requires no mental conversion, and requires no programming, just a command-line statement.

Anyway, if you still insist in comparing a once useful but now obsolete system to perform calculations to an aberrant, obfuscated system to do the same, you should focus in programs which show off their differences to the best. For instance, if you want to highlight RPN, you must seek examples which:

- o Make good use of automatic stack replication, i.e., T is replicated whenever the stack drops. A great example for this would be arbitrary (and *not* specified in advance!) degree polynomial evaluation.
- o Make good use of the fact that the stack is always 4 registers deep so it's *never* 'nonexistent'.
- o Make good use of LASTX and register arithmetic and manipulations directly or indirectly upon stack registers (STO+Y, STO-L, RCL Z, RCL IND ST T, X<> IND ST L, XEQ IND ST X, etc). A good example would be performing complex number multiplication directly on the stack, as well as converting numeric values in the ALPHA register to their full equivalent numeric value in X, just using the stack.
- o Make good use of numerically addressed registers, specially in combination with register arithmetic and indirection.
- o In the case of the HP42S, make good use of the fact that the whole numerically addressable register set can be dealt with *en masse* as a matrix variable, so you can have multiple sets of independent numerically addressable register banks and the possibilities are almost limitless. A good example for this would be to provide recursion in a simple way, as well as working with any number of matrices simultaneously.

etc, etc, etc. I'm glad if any of these comments of mine can be of any help to your quest, but for the reasons mentioned at the beginning, I wouldn't waste much time on this.

Best regards from V.

Edited: 22 Feb 2006, 6:59 a.m.

[[Return to Index](#) | [Top of Index](#)]



Go back to the main exhibit hall