♦ MoHPC ♠     *The Museum of HP Calculators*

# HP Forum Archive 18

[ Return to Index | Top of Index ]

---

## 35s Programming (need help)

*Message #1 Posted by Chuck on 4 Feb 2008, 9:18 p.m.*

Hi all. I'm covering arc-length in my Calc II class tomorrow, and need some help with the 35s. Consider the problem below:

```
6 = Int[sqrt(1 + 4t^2),t,0,x)
```

That is, the integral of root(1+4t^2) from 0 to x equals 6: solve for x. This can easily be done on various TI calculators. But on page 15-11 of the 35s manual, it says "SOLVE cannot call a routine that contains an Integral(FN) instruction" and visa-versa (or am I missing something here?)

So, is there an elegant way to "solve" an integral equation on the 35s? It pains me to take in a TI. I may have to resort to my HP 50g, but I'd rather do this on a non-graphing machine. I'll work on it later tonight after my rehearsal.

Thanks in advance, CHUCK

*Edited: 4 Feb 2008, 9:18 p.m.*

---

## Re: 35s Programming (an HP-34C/15C/41 solution)

*Message #2 Posted by Karl Schneider on 5 Feb 2008, 1:56 a.m.,*
*in response to message #1 by Chuck*

Hi, Chuck --

> Quote:
>
> ---
>
> ... But on page 15-11 of the 35s manual, it says "SOLVE cannot call a routine that contains an Integral(FN) instruction" and visa-versa (or am I missing something here?)
>
> ---

You're not missing anything. The ability to run SOLVE within INTEG or vice-versa was not provided on the HP-32S in 1988, and that shortcoming has been carried forward on all its successors -- the HP-32SII, the HP-33s, and the HP-35s.

From my only HP Forum article:

http://www.hpmuseum.org/cgi-sys/cgiwrap/hpmuseum/articles.cgi?read=556

> Quote:
>
> _The SOLVE and INTEG functions were introduced on the HP-34C in 1979, and later directly ported to the HP-15C in 1982. For the HP-41C/CV/CX, the same functions were adapted with only slight modification in the Advantage Pac introduced in 1985. Consequently, the protocol for using SOLVE and INTEG on these models is almost identical ... Moreover, the program may include a SOLVE or INTEG instruction, thus allowing INTEG to be executed within SOLVE, or vice versa. This can be useful for solving a limit of integration, or a parameter of an integrand._
>
> _..._
>
> _Neither the HP-32S, the HP-32SII, the HP-33s, nor the HP-35s provide the capability to execute INTEG within SOLVE, or vice versa. The HP-42S, however, retained that capability of the original implementation._

```
6 = Int[sqrt(1 + 4t^2),t,0,x)
```

Calculation of the arc length of a function is a standard calculus problem, e.g.: sqrt[(dx)^2 + (dy)^2] = sqrt[1 + (dy/dx)^2].dx

This function is apparently parameterized as

$x = t$; $y = t^2$, so

sqrt[(dx/dt)^2 + (dy/dt)^2].dt = sqrt[1 + (2t)^2].dt

The integrand will be modestly larger than 2t. Integrating 2t.dt from 0 to x such that the integral will be equal to 6.0 yields x = sqrt(6) = 2.449. The actual solution will be lower than that.

Here's a program and result returned by my 1982-made LED HP-34C, which is executable without modification on an HP-15C, and as adapted (quite minimally) for the HP-41 with Advantage ROM:

```
Program:
```

```
HP-15C/34C        HP-41 with Advantage


LBL A             LBL "AA"
0                 0
x<>y              x<>y
INTEG B           "AB"
6                 INTEG
-                 6
RTN               -
LBL B             RTN
x²                LBL "AB"
4                 x²
*                 4
1                 *
+                 1
SQRT              +
RTN               SQRT
                  RTN


Execution:


FIX 3             FIX 3
2                 2
ENTER             ENTER
3                 3
SOLVE A           "AA"
                  SOLVE


(Answer: 2.305419905)


FIX 4             FIX 4
2.305             2.305
ENTER             ENTER
2.307             2.307
SOLVE A           "AA"
                  SOLVE


(Answer: 2.306097460)
```

This is a good application of INTEG within SOLVE, the mechanics of which I've done before. However, I've yet to encounter or come up with a realistic and practical application of SOLVE within INTEG.

-- KS

*Edited: 9 Feb 2008, 1:46 a.m. after one or more responses were posted*

## Re: 35s Programming (need help)

*Message #3 Posted by Antonio Maschio (Italy) on 5 Feb 2008, 2:51 a.m.,*
*in response to message #2 by Karl Schneider*

K = G

now, if

K is a variable that stands for Karl
G is a constant that represents Genius,

then the above equivalence is true.

-- Antonio

## A link to real genius...

*Message #4 Posted by Karl Schneider on 6 Feb 2008, 12:31 a.m.,*
*in response to message #3 by Antonio Maschio (Italy)*

Hi, Antonio --

*Well! Gosh!*

:-)

I'm certainly pleased that you've appreciated my posts, in particular about the HP-15C.

In truth, though, my response to Chuck's integration question was only a straight-ahead presentation of prior knowledge, without exceptional insight. Original work that some very smart people have presented and offered here is far more impressive.

*True* genius is a very rare thing; the term ought to be applied with discretion. Here's one example from science and electrical engineering:

[http://en.wikipedia.org/wiki/Oliver_Heaviside](http://en.wikipedia.org/wiki/Oliver_Heaviside)

Best regards,

-- KS

*Edited: 9 Feb 2008, 1:28 a.m.*

## Re: 35s Programming (need help)

*Message #5 Posted by Gerson W. Barbosa on 5 Feb 2008, 6:24 a.m.,*
*in response to message #2 by Karl Schneider*

> Quote:
>
> ```
> Answer: 2.305
> ```

Hello Karl,

This quite matches what we get when solving 2*X*SQRT(4*SQ(X)+1)-LN(SQRT(4*SQ(X)+1)-2*X)-24 on the HP-33s, that is, 2.30609738256. However the expression was obtained with help of the HP-50G ('sqrt(4*T^2+1)', 'T', RISCH; then the proper substitutions). It's a pity the HP-33s cannot handle INTEG within SOLVE. I was not aware of this. Thanks for the information and explanations!

Regards,

Gerson.

*Edited: 5 Feb 2008, 6:28 a.m.*

## Re: 35s Programming (need help)

*Message #6 Posted by Karl Schneider on 6 Feb 2008, 12:08 a.m.,*
*in response to message #5 by Gerson W. Barbosa*

Hi, Gerson --

> Quote:

*(2.305)* quite matches what we get when solving 2*X*SQRT(4*SQ(X)+1)-LN(SQRT(4*SQ(X)+1)-2*X)-24 on the HP-33s, that is, 2.30609738256.

The somewhat-coarse "FIX 3" diminished the accuracy of the calculation of integrals. I chose it in order to to limit the execution time to a reasonable amount on the HP-34C, which is the slowest of the four RPN models I mentioned. The HP-15C will run the same program without modification, but slightly faster.

Using "FIX 4", I got 2.306097460 on an HP-41CV with Advantage Pac, whose "SOLVE" and "INTEG" microcoded programs use the predecessor algorithms. Only slight changes to the program are needed -- alphanumeric external labels for the routines passed to SOLVE and INTEG.

I'll post an HP-42S solution when I figure out exactly how to do it.

Regards,

-- KS

*Edited: 6 Feb 2008, 2:29 a.m.*

## HP-71B version

*Message #7 Posted by Valentin Albillo on 6 Feb 2008, 2:30 p.m.,*
*in response to message #6 by Karl Schneider*

Hi, Karl & Chuck:

Karl posted:

> *"'ll post an HP-42S solution when I figure out exactly how to do it."*

Ah, yes, I can't fail to provide the **HP-71B** solution as well, which is the following *statement* (*not program*) executed directly from the command prompt, no programming whatsoever necessary:

```
>  FNROOT(2,3,INTEGRAL(0,FVAR,1E-3,SQR(1+4*IVAR^2))-6)

       2.3054199049
```

(result obtained in negligible time, or you can use **1E-12** instead of **1E-3** above to get full accuracy: 2.30609738256)

which is as easy as it gets, as you don't need to write any program code, not even for the function to be integrated or the equation to be solved.

Also, you don't need to *"figure out exactly how to do it"* as is frequently the case with the relatively complicated setup the HP42S needs to solve or integrate your programs.

By the way, I'd like to see an RPL version of this. Taking into account that RPL supposedly is "the acme of power and flexibility", I expect the RPL version to be even simpler, shorter, and more immediate to concoct, enter, and execute than even my humble HP-71B solution above, right ? :-)

Best regards from V.

## HP 49g / 50g version

*Message #8 Posted by Gene Wright on 6 Feb 2008, 3:03 p.m.,*
*in response to message #7 by Valentin Albillo*

Actually, it was pretty easy as well. Here's what I did.

I entered the equation writer environment and typed in:

(integral from 0 to X of ( 1 + 4 x T^2 ) dT ) - 6 = 0

(Some of those parentheses are to help we humans parse the equation.

I stored that into 'EQ' I entered the Num.Slv menu. I pushed down arrow to highlight the X row. I pressed Solve.

Returned 2.30609738256

fairly quickly. Not bad and required no programming.

## Re: HP 49g / 50g version

*Message #9 Posted by Valentin Albillo on 6 Feb 2008, 3:33 p.m.,*
*in response to message #8 by Gene Wright*

Hi, Gene:

Gene posted:

*"I entered the equation writer environment and typed in:*

*(integral from 0 to X of ( 1 + 4 x T^2 ) dT ) - 6 = 0*

*[...] I pressed Solve. Returned 2.30609738256"*

> Amazing !! Most especially since you didn't even enter the necessary square root operation, yet the calc guessed your intent and computed the correct result nevertheless !!! Wow, I want one of those !! ;-)
>
> (and don't even think about *editing and correcting* your post above and thus rendering this reply incomprehensible or I'll call you chicken for the rest of your life :-) )

Best regards from V.

---

# Re: HP 49g / 50g version

*Message #10 Posted by Gene Wright on 6 Feb 2008, 3:57 p.m.,*
*in response to message #9 by Valentin Albillo*

Yes, that amazing new "GuessUserIntent.lib" file that I installed recently is very helpful. Makes those of us who can't type get correct results more often. :-)

FWIW, I don't know how long the solution would take on a real 49g+/50g. I ran it in a second or so on Power48 on my Palm E2.

---

# Re: HP 49g / 50g version

*Message #11 Posted by Damir on 7 Feb 2008, 7:07 a.m.,*
*in response to message #10 by Gene Wright*

With standard number format on a real 50g solution would take about 65 seconds and 5 seconds with Fix 3 number format.

Damir

*Edited: 7 Feb 2008, 7:21 a.m.*

---

# Re: HP 49g / 50g version

*Message #12 Posted by George Bailey (Bedford Falls) on 7 Feb 2008, 7:16 a.m.,*
*in response to message #10 by Gene Wright*

Quote:

FWIW, I don't know how long the solution would take on a real 49g+/50g.

Measured with the second hand of my watch

49G+ 93s (really, a tad faster)

50G 94s

*Edited: 7 Feb 2008, 7:17 a.m.*

---

# Re: HP 49g / 50g version

*Message #13 Posted by Chuck on 8 Feb 2008, 1:22 a.m.,*
*in response to message #12 by George Bailey (Bedford Falls)*

You guys are all great: fun discussion. Since my programming skills aren't what they used to be, and no where near to most here), and since no one has mentioned the ubiquitous HP 200LX, I ran my equation on my 200LX running Derive, and it solved it in 10.1 seconds. Keystrokes: didn't count.

And, George, *my* slide rule out in the garage is 7-feet long. That should be enough to solve this problem "on a slide rule." :)

Thanks all for this great discussion, and for everyones insightful programming expertise. Looks like a good weekend coming up.

CHUCK

---

# Re: HP 49g / 50g version

*Message #14 Posted by George Bailey (Bedford Falls) on 8 Feb 2008, 2:20 a.m.,*
*in response to message #13 by Chuck*

Quote:

And, George, *my* slide rule out in the garage is 7-feet long. That should be enough to solve this problem "on a slide rule." :)

Any photos? ;-)

## Re: HP 49g / 50g version

*Message #15 Posted by Chuck on 8 Feb 2008, 12:32 p.m.,*
*in response to message #14 by George Bailey (Bedford Falls)*

Here's the one in the garage: Dietzgen

Here's the 7-footer in my office: Pickett

CHUCK

## Re: HP 49g / 50g version

*Message #16 Posted by George Bailey (Bedford Falls) on 8 Feb 2008, 2:15 p.m.,*
*in response to message #15 by Chuck*

Chuck, this is really great! Never saw slide rules that BIG! Thank you very much for these pictures. (I love this thread!)

And now I go hide where you won't find me. I'm scared! Your's are sooo much bigger than mine ;-)

But speaking of Dietzgen, I might take this one which can easily be attached to the belt "for a quick draw"...

*Edited: 8 Feb 2008, 2:19 p.m.*

## Re: HP 49g / 50g version

*Message #17 Posted by Karl Schneider on 7 Feb 2008, 3:02 a.m.,*
*in response to message #8 by Gene Wright*

Hi, Gene --

Valentin stated,

> Quote:
>
> By the way, I'd like to see an RPL version of this. Taking into account that RPL supposedly is "the acme of power and flexibility",
> I expect the RPL version to be even simpler, shorter, and more immediate to concoct, enter, and execute than even my humble
> HP-71B solution above, right ? :-)

Yours was a very sensible and straightforward approach, but my objection would be that it's not really <u>RPL</u>, which is what Valentin asked for.

Egan Ford subsequently provided his solutions using algebraic-equation and pure-RPL approaches. The ability of the particular model of RPL-based calculator to symbolically integrate the equation will have a bearing on computational speed.

It's also good that you titled your solution "HP 49G/50g version", because the same input might not work at all, or as well, on different RPL-based models.

-- KS

## Re: HP-71B version

*Message #18 Posted by Egan Ford on 6 Feb 2008, 3:31 p.m.,*
*in response to message #7 by Valentin Albillo*

From the stack:

```
50/50 ALG/RPL


'INTEGRAL(0,X,SQRT(1+4*X^2),X)-6' 'X' 0 ROOT (39 keystrokes)


100% RPL:


'X' PURGE (5 keystrokes)
0 X X 2 ^ 4 * 1 + SQRT X INTEGRAL 6 - X 0 ROOT (29 keystrokes)
```

100% RPL version is much faster.

*Edited: 6 Feb 2008, 3:53 p.m.*

## Re: (RPL) version

*Message #19 Posted by Karl Schneider on 6 Feb 2008, 10:33 p.m.,*
*in response to message #18 by Egan Ford*

Good work, Egan!

> Quote:

```
50/50 ALG/RPL


'INTEGRAL(0,X,SQRT(1+4*X^2),X)-6' 'X' 0 ROOT (39 keystrokes)


100% RPL:


'X' PURGE (5 keystrokes)
0 X X 2 ^ 4 * 1 + SQRT X INTEGRAL 6 - X 0 ROOT (29 keystrokes)
```

100% RPL version is much faster.

---

Well, of course it is! On the HP-49G at least, the pure-RPL version computes the integral symbolically upon entry of the INTEGRAL symbol, then numerically solves only the antiderivative. The other solutions (including your "50/50 ALG/RPL" method) numerically estimate the integral as part of the function to be root-solved. The difference in speed would be much less for an integrand having no determinable closed-form integral. (This applies to the original HP-48, whose less-capable CAS tries but fails to perform the symbolic integration of this problem.)

RPL novices (such as myself) should also note that " 'X' PURGE " is necessary if X exists; else, X will be *evaluated upon entry* instead of entered into the equation.

-- KS

*Edited: 6 Feb 2008, 11:58 p.m.*

---

## HP-42S solution for INTEG inside SOLVE

*Message #20 Posted by Karl Schneider on 7 Feb 2008, 3:31 a.m.,*
*in response to message #7 by Valentin Albillo*

Here are several examples of an HP-42S solution for Chuck's arc-length calculation:

```
/ X
|
|    sqrt(1 + 4t²) dt = 6;    solve for X
|
/ 0


"More-detailed"            "Concise"
```

```
01 LBL "AL1"            01 LBL "AL2"
02 MVAR "X"             02 MVAR "ULIM"
03 0                    03 PGMINT "IG"
04 STO "LLIM"           04 INTEG "t"
05 RCL "X"              05 6
06 STO "ULIM"           06 -
07 PGMINT "IG"          07 RTN
08 INTEG "t"            08 LBL "IG"
09 6                    09 RCL "t"
10 -                    10 x²
11 RTN                  11 4
12 LBL "IG"             12 *
13 RCL "t"              13 1
14 x²                   14 +
15 4                    15 SQRT
16 *                    16 RTN
17 1
18 +
19 SQRT
20 RTN


Execution:              Execution:


0.000001                0.000001
STO "ACC"               STO "ACC"
2                       0
STO "X"                 STO "LLIM"
3                       2
SOLVER                  STO "ULIM"
(select "AL1")          3
(select "X" twice)      SOLVER
                        (select "AL2")
                        (select "ULIM" twice)
```

**Answer:**

**2.30609738393**

This example shows many of the formalities that must be addressed when performing SOLVE and INTEG on an HP-42S:

    1. Each program must be assigned an alphanumeric external label.

2. Each variable being solved or integrated must be named.

3. For interactive-mode invocation, the SOLVE variable must be included in an "MVAR" statement, or else the function to be solved will not appear in the selection list.

4. Values for the limits of integration, as well as the uncertainty ("accuracy") must be stored by their designated variable names.

5. When invoking integration inside a program, "PGMINT" must specify the name of the routine to be integrated. (Similarly, "PGMSLV" must be used to specify the name of the routine containing the function to be root-solved, from within a program.)

---

And now, the vexing question: So, why can't the HP-32S/32SII/33s/35s run INTEG inside SOLVE and vice-versa? One minor issue is the following:

*Only one "FN=" statement is provided for use by both SOLVE and INTEG -- unlike the pair "PGMSLV" and "PGMINT" in the HP-42S.*

This is not a disqualifying limitation, as the information provided within "FN=" at invocation of SOLVE or INTEG could be stored elsewhere. However, this could cause some confusion to the user.

More likely, the reason is that the programming effort to maintain this functionality was simply not performed, for reasons of budget or product differentiation. It is noted that integration requires 140 bytes, and SOLVE requires 33.5 bytes, on the HP-32S and HP-32SII, which respectively offer only 190 bytes and 184 bytes of allocatable memory. However, SOLVE and INTEG work together on the HP-34C, HP-15C, and HP-41 Advantage Pac, while requiring no more memory than does INTEG alone. Furthermore, the HP-33s and HP-35s have 31 kB of free memory.

-- KS

*Edited: 10 Feb 2008, 10:01 p.m. after one or more responses were posted*

---

### Re: HP-42S solution for INTEG inside SOLVE
*Message #21 Posted by Valentin Albillo on 7 Feb 2008, 6:16 a.m.,*
*in response to message #20 by Karl Schneider*

Hi, Karl:

Thanks for your neat HP42S version and subsequent explanations which will surely prove to be very useful for novice 42S programmers as well as for seasoned ones that just can't remember all the convoluted details when the need arises (every Pope's death or so, as we

say in Spain ["cada muerte de Papa"]). If I'm not wrong you mentioned that you found yourself in that exact case when considering an HP42S solution ... :-)

That said, I still think that *the HP-71B way is by far the easiest, most convenient, most general way of solwing this kind of problem*, because:

- *The syntax is crystal-clear and pretty much intuitive*. The only thing to remember is that IVAR (<u>I</u>ntegration <u>VAR</u>iable) stands for the integration variable and FVAR (<u>F</u>indroot <u>VAR</u>iable) stands for the variable to solve. Their mnemonic names greatly facilitates it.

  This perfectly scales to nested Solves and Integrates, possibly mixed up in any order, up to 5 of each, and is much simpler than the complicated setup which plagues HP42S solving & integrating, which, quite frankly, requires the user to remember very precise details in order to casually compute an integral (and further doesn't allow for nesting more than one solve or more than one integrate at a time).

- *No programming is required whatsoever*, you can simply execute it from the command prompt. Most other solvers and integrators, if not all, require the user to write a program for the function to be solved or integrated.

  This no-programming-needed, command-line solving & integration ability of the HP-71B was a conscious design specification stated by the HP-71B Math ROM's programming team in the corresponding article featured in HP Journal, July 1984, pp30-33 (10.5 Mb PDF document) and it was pretty unique at the time.

- *It doesn't depend on the ability to being able to symbolically compute the integral* but will work for arbitrary integrands even if they can't be symbolically integrated. The RPL versions offered rely specifically on that ability and wouldn't work as listed for non-symbolically integrable functions.

  I'll hazard my guess that a purely numerical RPL version for Chuck's problem wouldn't run so fast nor would it be so simple to set up.

Best regards from V.

---

## Re: HP-42S solution for INTEG inside SOLVE

*Message #22 Posted by George Bailey (Bedford Falls) on 7 Feb 2008, 6:54 a.m., in response to message #21 by Valentin Albillo*

Valentin, sorry for hijacking your layout ;-) But it's soooo neat...

I think that *the HP 49+/50G way is by far the easiest, most convenient, most general way of solwing this kind of problem*, because:

- ***The syntax is crystal-clear and pretty much intuitive***. The Equation Writer lets you enter the integral in textbook style. [...]

- ***No programming is required whatsoever***.

- ***It doesn't depend on the ability to being able to symbolically compute the integral*** but will work for arbitrary integrands even if they can't be symbolically integrated. You just use the NUM.SLV numerical solver.

*Edited: 7 Feb 2008, 6:54 a.m.*

## Re: HP-42S solution for INTEG inside SOLVE

*Message #23 Posted by [Valentin Albillo](#) on 7 Feb 2008, 9:06 a.m.,*
*in response to message #22 by George Bailey (Bedford Falls)*

Hi, George:

George posted:

> *"Valentin, sorry for hijacking your layout ;-) But it's soooo neat..."*

> Thanks, glad you like it. As for the hijacking itself, be my guest.

> *"I think that the HP 49+/50G way is by far the easiest, most convenient, most general way of solwing this kind of problem"*

> Couple' comments:

> 1. Equation Writers, Equation Writers, ... stop cheating and fight like a man ! Real programmers are command-line guys ! ... this is what happens with so much Windows, GUIs, and the rest of hedonistical trends in computing.

> 2. You can brag all you want about that wondeful gadget of yours but we both know that if we confronted each other, both of us armed with a zip case with the master-cleared calculator inside, I'd have the result on my HP-71B's display before you even finished navigating to the required menus, with or without previously consulting some manual for the proper procedure ... :-)

Best regards from V.

---

# 71B command line vs. 49g+ / 50g apps

*Message #24 Posted by [Gene Wright](#) on 7 Feb 2008, 9:58 a.m.,*
*in response to message #23 by Valentin Albillo*

I'm pretty sure that Valentin's 71B command line approach would be somewhat / marginally faster than the EquationWriter / Num.Slv approach on the 49g+ / 50g.

However, ISTM that there is a bit more syntax knowledge required to use the command line 71B approach than I did with the 49g+ / 50g application approach.

The EquationWriter entry of the integral was fairly painless. I really didn't have to know what order to type in arguments - I just keyed in what appeared to be needed to "view" the integral.

Num.Slv was a piece of cake too.

Don't get me wrong, both the 71B and 49g+ / 50g approaches are MUCH faster/better than probably most other machines could muster.

It is my opinion though that the 71B approach would require more knowledge of the command line syntax than any specialized knowledge needed on the 49g+ / 50g.

And, yes, there is a :-) scattered throughout.

Personally, I wouldn't challenge Valentin to ANY programming contest...command line or NOT!

---

# Re: 71B command line vs. 49g+ / 50g apps

*Message #25 Posted by [George Bailey (Bedford Falls)](#) on 7 Feb 2008, 10:18 a.m.,*
*in response to message #24 by Gene Wright*

> Quote:
>
> Personally, I wouldn't challenge Valentin to ANY programming contest...command line or NOT!

Yup, I've just hidden behind a tree, taken off my Stetson so he won't find me...

## Re: 71B command line vs. 49g+ / 50g apps

*Message #26 Posted by Valentin Albillo on 7 Feb 2008, 10:32 a.m.,*
*in response to message #25 by George Bailey (Bedford Falls)*

...

   I can see your slide ruuuuuule ... :-)

Best regards from V.

## Re: 71B command line vs. 49g+ / 50g apps

*Message #27 Posted by George Bailey (Bedford Falls) on 7 Feb 2008, 10:39 a.m.,*
*in response to message #26 by Valentin Albillo*

 Quote:

 ...

   I can see your slide ruuuuuule ... :-)

Shit, why did I bring it? Could've left it in the corral with the others...

## Re: HP-42S solution for INTEG inside SOLVE

*Message #28 Posted by George Bailey (Bedford Falls) on 7 Feb 2008, 10:05 a.m.,*
*in response to message #23 by Valentin Albillo*

Valentin,

;-)

1: real programmers are punch card guys. Well, maybe even spoked metal wheel dials guys ;-)

2: my 50G essentially is master-cleared. And as for this specific problem, I can key in the whole integral in its equation writer with fewer keystrokes as you can type the word "INTEGRAL" on the 71B (actually I need 23, I was just bragging! 32 in total for the

result as opposed to your 50 something key strokes. And I don't need an extra Math ROM...).

And before we both ride off into the sunset: your 71B is longer than my 50G (just compared it). But my 71B is as long as yours!!! AND my slide rule is even longer!!!!!!!

*Edited: 7 Feb 2008, 10:16 a.m.*

---

# Re: HP-42S solution for INTEG inside SOLVE
*Message #29 Posted by Valentin Albillo on 7 Feb 2008, 10:28 a.m.,*
*in response to message #28 by George Bailey (Bedford Falls)*

Hi again, George:

George posted:

> *"real programmers are punch card guys."*

> I've been one of those, but I didn't specially like it, at all. I happened to buy my very first HP calculator at the time, the wondrous HP-25, and I just couldn't believe how much easier was it to key in and run a root finder program (say) on it as compared to punching the cards for the computer, etc. Neither could my teachers, colleagues, ...

> *"I can key in the whole integral in its equation writer with fewer keystrokes as you can type the word "INTEGRAL" on the 71B (actually I need 23, I was just bragging!)."*

> Very good. Starting from the master-cleared, zipped state I can get the result in 54 keystrokes, including [ON] and [ENDLINE].

> The natural state of my HP-71B isn't master-cleared, of course, and I actually have a "keys" file in RAM which allows the INTEGRAL and FNROOT statements to be entered (together with their commas and parentheses) at the touch of a few keys in user mode.

> This also includes FVAR, IVAR, and many other such, so in actual practice I don't really need anywhere near 54 keystrokes to enter that expression, I had to actually count them one by one to get to know just how many were indeed required in a master-clear startup situation.

*"AND my slide rule is even longer!!!!!!!"*

Now solving Chuck's problem with a slide rule would be an interesting challenge. I know how to do it using an old HP-35 (not "HP35s") and it's just about workable, if laborious.

Thanks for indulging my good-humored rant and ...

Best regards from V.

---

## Re: HP-42S solution for INTEG inside SOLVE

*Message #30 Posted by George Bailey (Bedford Falls) on 7 Feb 2008, 10:34 a.m.,*
*in response to message #29 by Valentin Albillo*

Quote:

Thanks for indulging my good-humored rant and ...

Why, thank YOU for your 35S solution of Chuck's problem. It made me dive into my 35S more than I had before.

---

## Re: HP-42S solution for INTEG inside SOLVE

*Message #31 Posted by Garth Wilson on 7 Feb 2008, 2:44 p.m.,*
*in response to message #29 by Valentin Albillo*

If you really want to get picky, the 71 has a 2/3-size QWERTY keyboard. When I typed a lot on mine, I was able to do 30+wpm, which is just over half of what I can do on a full-sized keyboard. I quit typing memos and meeting notes on it when the 71 was discontinued and it was kind of sobering to think that I could wear out the keyboard and not have any HP support. By coincidence, I just had this discussion with a friend over the 50g and 71B a few days ago.

---

## Re: HP-42S solution for INTEG inside SOLVE

*Message #32 Posted by Valentin Albillo on 7 Feb 2008, 3:03 p.m.,*

*in response to message #31 by Garth Wilson*

Hi, Garth:

Garth posted:

> *"By coincidence, I just had this discussion with a friend over the 50g and 71B a few days ago. "*
>
> Who won ?
>
> As for typing on the **HP-71B**, I remember you could attach an external full-size keyboard and use the **KEYBOARD IS** keyword (available in several media, the **FORTH/Assembler ROM** among them) to direct the mainframe to accept keystrokes from it instead.

Best regards from V.

---

## Re: HP-42S solution for INTEG inside SOLVE

*Message #33 Posted by Garth Wilson on 7 Feb 2008, 4:22 p.m.,*
*in response to message #32 by Valentin Albillo*

> Quote:
> ___
> Who won ?
> ___

There was no conflict about any of it, and he immediately agreed with me concerning the keyboard. He is very pleased with his new 50g though. It's his first HP, and it has gotten him interested in its ancestry-- especially the 41 and 71 which he is also interested in acquiring. But first he wants to expand the use of the serial port on the 50g to interface to lab instrumentation, an area of work we share.

---

## Re: HP-42S solution for INTEG inside SOLVE

*Message #34 Posted by Thomas Klemm on 7 Feb 2008, 3:03 p.m.,*
*in response to message #28 by George Bailey (Bedford Falls)*

> Quote:
> ___
> real programmers are punch card guys
> ___

Real programmers set the universal constants at the start such that the universe evolves to contain the disk with the data they want.

## Re: HP-42S solution for INTEG inside SOLVE

*Message #35 Posted by George Bailey (Bedford Falls) on 7 Feb 2008, 3:17 p.m.,*
*in response to message #34 by Thomas Klemm*

Quote:

Real programmers set the universal constants at the start such that the universe evolves to contain the disk with the data they want.

;-)

And for that there's a set of 71B commands as well

DESTROY ALL @ DIM UNIVERSE[INF] @ EVOLVE NEW CONST @ PRONTO

## Re: HP-42S solution for INTEG inside SOLVE

*Message #36 Posted by Karl Schneider on 8 Feb 2008, 2:38 a.m.,*
*in response to message #21 by Valentin Albillo*

Hi, Valentin --

Quote:

Thanks for your neat HP42S version and subsequent explanations which will surely prove to be very useful for novice 42S programmers as well as for seasoned ones that just can't remember all the convoluted details when the need arises... If I'm not wrong you mentioned that you found yourself in that exact case when considering an HP42S solution.

Thank you; it was an exercise I'd never completed, so the structure is more clear to me now.

Really, the process for solving this problem and related ones on the HP-42S is fundamentally similar to that for the HP-15C, HP-34C, and HP-41/Advantage. It's just that more *formalities* must be addressed, due to the greater sophistication of the HP-42S' implementation of

SOLVE/INTEG with programming. If one doesn't like RPL and doesn't have an HP-71B <u>with Math ROM</u>, then the HP-42S is the only Saturn-processor tool for the problem posed by "Chuck".

> Quote:
>
> The RPL versions offered rely specifically on *(the ability to symbolically compute the integral)* and wouldn't work as listed for non-symbolically integrable functions...

Not true for Egan Ford's pure-RPL solution, at least. The HP-48G finds the root, but takes a while because it cannot symbolically integrate the function, and must then use numerical integration. The HP-49G's more-capable CAS <u>can</u> do the symbolic integration, and finds the root quickly as a result.

> Quote:
>
> That said, I still think that the HP-71B way is by far the easiest, most convenient, most general way of solving this kind of problem, because:

I've got a well-equipped HP-71B, acquired piecemeal a few years ago through eBay and one benevolent Forum reader. It's got HP-IL, Math ROM, Surveying ROM, HP-41/71 Translator, (32 + 4 =) 36 kB of extra RAM. I have all paper manuals save for one, and all the manuals in electronic form.

I've run programs you've posted and have made available, but I'm still not very skilled at the HP-71B. It will take much more time and effort, RTFM and "doing", in order to gain proficiency. While the design is well-engineered as the July 1984 HP Journal articles illustrate, it's still a small computer utilizing a high-level language having its own syntactical requirements.

-- KS

## Re: 35s Programming (need help)

*Message #37 Posted by [George Bailey (Bedford Falls)](#) on 7 Feb 2008, 7:22 a.m.,*
*in response to message #5 by Gerson W. Barbosa*

> Quote:
>
> It's a pity the HP-33s cannot handle INTEG within SOLVE. I was not aware of this.

Me neither in respect to the 35s. But I now found out that the manual tells us so in its 'messages' section.

## Re: 35s Programming (an HP-34C/15C/41 solution)

*Message #38 Posted by Valentin Albillo on 8 Feb 2008, 10:59 a.m.,*
*in response to message #2 by Karl Schneider*

Hi, Karl:

Karl posted:

"*Here's a program and result returned by my 1982-made LED HP-34C [...]*

```
LBL B
 *
 4
 *
 1
 +
SQRT
RTN
```

*execution: [...]*

May I suggest that here (and in LBL "AB" as well) you can use instead:

```
LBL B
 +
 1
->P
RTN
```

which saves 3 full steps (50%).

Looks smart as well, doesn't it ? :-)

Best regards from V.

## Re: 35s Programming (an HP-34C/15C/41 solution)

*Message #39 Posted by Karl Schneider on 8 Feb 2008, 11:55 a.m.,*
*in response to message #38 by Valentin Albillo*

Hi, Valentin --

Hmm, I dunno. Your alternative is a clever optimization for fewest instructions, but it kind of "defeats the purpose" two ways:

1. The expression being calculated is not as immediately clear. (In fact, I ought to have used "$x^2$" instead of the first "*" -- it's just as fast.)

2. The Rect->Pol conversion also computes an unnecessary and time-consuming arctangent, which will bog down the numerical integration that is performed multiple times.

Best regards,

-- KS

*Edited: 8 Feb 2008, 3:28 p.m.*

## Re: 35s Programming (need help)

*Message #40 Posted by Valentin Albillo on 5 Feb 2008, 8:04 a.m.,*
*in response to message #1 by Chuck*

Hi, Chuck:

Chuck posted:

> *"So, is there an elegant way to "solve" an integral equation on the 35s?"*

If by *"elegant"* you mean some way to make the built-in Solve functionality work with the built-in Integrate functionality then sorry, that's impossible in the current firmware.

What to do ? Well, you'll have to replace one or the other with a user-created program, as simple and efficient as possible. As integration programs tend to be much more complicated and slower than root-finding programs, we'd do well to make use of the built-in Integrate and provide instead our own Solver.

To that effect, I suggest you use something like the solver I created and proposed decades ago for its inclusion in the PPC ROM, which you can find in the following thread: Valentin Albillo's Root-finder

If you visit that thread (which you should do, for the documentation if nothing else) you'll notice that I wrote the program for the HP-41C which was the rage at the time. For you to be able to use it in your HP35s, I'm providing the following converted (and slightly optimized) version instead

```
A001  LBL A
A002  CF 0              A016  X=0?
A003  STO A             A017  E^X
A004  50                A018   /
A005  STO B             A019  Go
A006  Go                A020  *
A007  RCL+ A            A021  STO- A
A008  XEQ F001          A022  RND
A009  STO C             A023  X=0?
A010  RCL A             A024  GTO A028
A011  XEQ F001          A025  DSE B
A012  X=0?              A026  GTO A006
A013  GTO A028          A027  SF 0
A014  STO- C            A028  RCL A
A015  RCL C             A029  RTN
```

where the **Go** at lines A006 and A019 is obtained from the **CONST**ants menu, defined as the "Conductance quantum", which evaluates to 7.748E-5 approximately.

Once you've entered this simple, general, and quite fast root-finding procedure o'mine, you just need to define the equation to solve, which in your case is essentially **Integral(0,X) = 6**, which gets programmed in the HP35s as:

```
F001  LBL F              I001  LBL I
F002  FN= I              I002  4
F003  0                  I003  RCLx T
F004  X<>Y               I004  RCLx T
F005  INTEG T            I005  1
F006  6                  I006  +
F007  -                  I007  SQRT
F008  RTN                I008  RTN
```

where **LBL F** is the equation being solved, which uses the built-in Integrate, and **LBL I** is the function being integrated, which is defined above as an **RPN** program segment. You can also define it as an **ALG**ebraic equation, like this:

```
I001  LBL I
I003  EQN SQRT(1+4*SQ(T))
I004  RTN
```

which is shorter and possibly clearer, but the RPN version above is significantly faster to evaluate.

Once you've entered the root-finder program (**LBL A**), the equation to solve (**LBL F**), and the function to integrate (**LBL I**), you can then proceed to specify the precision wanted (**FIX 3**) and supply a suitable initial guess (**3**) to get:

FIX 3, 3, XEQ A  -> **2.305**

in either 17 seconds for the **RPN** version of the function or 23 seconds for the **ALG**ebraic version, whichever you chose.

Hope this is more or less what you wanted. See the thread linked above for documentation on my root finder such as what happens if the computed derivative is zero or if the equation being solved has no real roots or convergence is extremely slow.

For solving equations with *complex roots* or *complex parameters* using your HP35s, have a look at my ***"Boldly Going - Going Back to the Roots"*** Datafile article published in Datafile September/October 2007 issue (V26N6P28).

Best regards from V.

---

## Re: 35s Programming (need help)

*Message #41 Posted by [Giancarlo (Italy)](#) on 5 Feb 2008, 9:04 a.m.,*
*in response to message #40 by Valentin Albillo*

Hey, Antonio - how about "V" == "K" in your equation above...? :-)
Ciao.
Giancarlo

---

## Great job, Valentin!

*Message #42 Posted by [Gene Wright](#) on 5 Feb 2008, 9:57 a.m.,*
*in response to message #40 by Valentin Albillo*

This is exactly the type of application I wish we'd see more of.

Personal note: It appears I will be passing through your country (well, at least the Madrid airport) sometime in May. No, I don't expect we'll see each other :-) but I will think about you when my son and I are spending 4 hours waiting for the next flight.

Cheers!

---

## Re: Great job, Valentin!

*Message #43 Posted by [Valentin Albillo](#) on 5 Feb 2008, 11:22 a.m.,*
*in response to message #42 by Gene Wright*

Hi, Gene:

    Long time no read !

Gene posted:

> *"This is exactly the type of application I wish we'd see more of."*

Thanks a lot for your kind words. Coming from the man who painstakingly created those amazing training modules for the HP35s it's quite an honor, much appreciated.

> *"I will be passing through your country (well, at least the Madrid airport) sometime in May [...] but I will think about you when my son and I are spending 4 hours waiting for the next flight."*

Sorry for the extreme delay (4 hours !) but believe me, you won't lose a thing for not seeing me, despite my other charms our Lord punished me with a heavy hand as far as beauty is concerned. Kinda compensation, I think ...

At least I hope you'll enjoy a good flight and typical Spanish fair weather, and thanks for thinking about me. In all fairness, we having met would have ruined your mental image of me beyond repair. :-)

Best regards from V.

---

## Re: 35s Programming (need help)

*Message #44 Posted by Chuck on 5 Feb 2008, 10:26 a.m.,*
*in response to message #40 by Valentin Albillo*

Valentin and Karl. Thank you so much for the outstanding replies. I will give the programming problem a go during my break today. My attempt was going to involve Newton's method to find the zeros (along with the Fundamential Theorem of Calculus Part 2), but I have a feeling the program length will be quite a bit longer. I would say your method IS as elegant as I was hoping. It looks like I'll have to pull out my 42s from the collection box and try something on that.

Much thanks again. You guys are amazing here. :)

CHUCK

---

## Re: 35s Programming (need help)

*Message #45 Posted by Valentin Albillo on 5 Feb 2008, 11:05 a.m.,*
*in response to message #44 by Chuck*

Hi again, Chuck:

Chuck posted:

> "Thank you so much for the outstanding replies"

You're welcome.

> "My attempt was going to involve Newton's method [...] but I have a feeling the program length will be quite a bit longer. I would say your method IS as elegant as I was hoping"

Thanks but my method *is* an implementation of Newton's method:

$$x_{i+1} = x_i - f(x_i) / f'(x_i)$$

so that's exactly how long it gets on the HP35s ! :-)

Best regards from V.

---

## Re: 35s Programming (need help)

*Message #46 Posted by Marcus von Cube, Germany on 7 Feb 2008, 7:11 p.m.,*
*in response to message #45 by Valentin Albillo*

When the function to solve is an integral, can't Newtons method be used "natively" by using the exact derivative instead of an approximation?

$$x_{n+1} = x_n - f(x_n) / f'(x_n)$$

In our case:
f(x) = Integ 0 to x g(t)dt - constant
f'(x) = g(x)

Am I right?

---

## Particularized version

*Message #47 Posted by Valentin Albillo on 8 Feb 2008, 10:41 a.m.,*
*in response to message #46 by Marcus von Cube, Germany*

Hi, Marcus:

Marcus posted:

*"When the function to solve is an integral, can't Newtons method be used "natively" by using the exact derivative instead of an approximation?"*

Yes, absolutely. The version I posted for the HP35 was my general-purpose Newton's-based solver, which deals with arbitrary equations not restricted to Chuck's integral-related one, and thus must compute an approximation to the derivative on the fly, as fast as possible.

For the particular case of solving integral equations such as Chuck's, you can indeed make use of the fact that the exact derivative is available and further it's already needed and thus stored in program memory for the purpose of computing its integral, so it can be called at no extra cost. The resulting particularized program for the HP35s looks as this:

```
A001  LBL A       A014  X=0?
A002  CF 0        A015  E^X
A003  STO A       A016  STO/ C
A004  50          A017  RCL C
A005  STO B       A018  ST0- A
A006  RCL A       A019  RND
A007  XEQ F001    A020  X=0?
A008  X=0?        A021  GTO A025
A009  GTO A025    A022  DSE B
A010  STO C       A023  GTO A006
A011  RCL A       A024  SF 0
A012  STO T       A025  RCL A
A013  XEQ I001    A026  RTN
```

which is three steps shorter, and *much faster*, taking just **9 seconds** to compute the **2.305** root at **FIX 3**. This is almost twice as fast, but the price you pay for it is, of course, that this particularized version will only work for equations involving integrals in a similar way as the original Chuck's problem.

By the way, I find it quite curious that my original message, which was #3 at the time I originally posted it, just after Karl Schneider's #2, is now at #36 and going down all the time as other much more recent posts are inserted before it.

Come to think of it, this seems a good way to effectively get rid of someone else's posts: just post a large number of messages in reply to an earlier post than the one you want to get rid of, and it will simply go down into oblivion as no one will care to scroll down that long.

See what happened to my original listing: it's now so far from the beginning of the thread that anyone casually reading the thread will obviously think my reply was made much, much after the original request for help and after seeing tons of other people's messages and solutions when in fact it's exactly the opposite ! :-)

Best regards from V.

## Re: Particularized version

*Message #48 Posted by Thomas Klemm on 8 Feb 2008, 12:48 p.m.,*
*in response to message #47 by Valentin Albillo*

Hi Valentin,

> Quote:
>
> ---
>
> By the way, I find it quite curious that my original message, which was #3 at the time I originally posted it, just after Karl Schneider's #2, is now at #36 and going down all the time as other much more recent posts are inserted before it.
>
> ---

With 8 posts in this thread you're part of the problem.
I like the way this forum is presented and had never problems to notice any of your posts.

Thomas

*Edited: 8 Feb 2008, 12:49 p.m.*

## Re: 35s Programming (need help)

*Message #49 Posted by Marcus von Cube, Germany on 7 Feb 2008, 7:05 p.m.,*
*in response to message #1 by Chuck*

The TI CAS caluculators manage the task easily:

On the Voyage 200 (same for ti-92) you enter:

solve(Integ(Sqrt(1+4t^2),t,0,x)=6,x) {Integ and Sqrt are symbols}

The result appears after about 95 seconds: x=2.30609738256, accompanied by a warning that more than one solution may exist.

My Nspire CAS comes to the same result in about 4 seconds. The entry is with the equation writer and hence more intuitive than the older command line entry method.

*Edited: 7 Feb 2008, 7:23 p.m.*

## Re: 35s Programming (TI solution)

*Message #50 Posted by Karl Schneider on 8 Feb 2008, 1:15 a.m.,
in response to message #49 by Marcus von Cube, Germany*

Hi, Marcus --

> Quote:
> _____
>
> The TI CAS caluculators manage the task easily:
>
> On the Voyage 200 (same for ti-92) you enter:
>
> solve(Integ(Sqrt(1+4t^2),t,0,x)=6,x) {Integ and Sqrt are symbols}
>
> The result appears after about 95 seconds: x=2.30609738256, accompanied by a warning that more than one solution may exist.
> _____

I have to admit, that's pretty good -- very straightforward. Several questions:

1. Is there a way of directing the solution by establishing, with two values, a range for the answer? Sure, the "warning that more than one solution may exist" is given, but that might not be what the user wants to know.

2. Is there a means of specifying the accuracy of calculation for numerical integration?

   > Quote:
   > _____
   >
   > My Nspire CAS comes to the same result in about 4 seconds. The entry is with the equation writer and hence more intuitive than the older command line entry method.
   > _____

3. Does the Nspire symbolically integrate the expression? If so, that might account for the speed, although I'm sure that the processor is fast and modern.

-- KS

*Edited: 9 Feb 2008, 12:54 p.m. after one or more responses were posted*

## Re: 35s Programming (TI solution)

*Message #51 Posted by Marcus von Cube, Germany on 8 Feb 2008, 2:49 a.m.,*

*in response to message #50 by Karl Schneider*

Karl, it looks as if the CAS is the same in both machines. If I just enter the integral, it is solved symbolically:

ln(sqrt(4x^2+1)+2x)/4 + x*sqrt(4x^2+1)/2

I assume that the solver operates on the solution.

There is an nSolve command with parameters to direct it to a specific solution. I can't see how the accuracy can be influenced. The display setting doesn't make any difference.

I tried the following on my Nspire:

nSolve(nInt(sqrt(4*t^2+1),t,0,x)=6,x)

It arrives at 2.30609738255 in a mattter of seconds, without any warning. If I set the inital guess to 2, the solution is returned within a second. Numerically solving the symbolic integral with an initial guess of 2 is even quicker.

*Edited: 14 Feb 2008, 11:58 a.m. after one or more responses were posted*

## Re: 35s Programming (TI-82 solution)

*Message #52 Posted by Karl Schneider on 14 Feb 2008, 12:34 a.m.,*
*in response to message #51 by Marcus von Cube, Germany*

Marcus --

OK, using your solution template, the TI-82 manual, and some experimentation, I got a solution on my 1993 TI-82. (Good thing it wasn't a "pressure situation"...)

solve(fnInt(sqrt(1+4*T$^2$),T,0,X)-6,X,{2,3})

yields **2.306093783**

For "solve" in this problem, one or two guesses are <u>required</u>, even though the manual suggests that they are optional. (I used 2 and 3.) Also, a pair of guesses cannot be entered in descending order, for no justifiable reason. The value for the "tolerance" parameter for integration -- undefined in the manual -- is omitted in my expression, so the TI-82 uses a default value of 1E-5.

The TI-82's Gauss-Kronrod numerical-integration method is very fast and accurate for routine, well-behaved integrals. I'm also sure that its microprocessor is no slouch, as the calc's 4 x "AAA" cells would suggest. However, the quadrature lacks the real-world sophistication of HP's implementation, such as evaluability at limits for which the integrand is undefined.

-- KS

*Edited: 16 Feb 2008, 1:44 p.m.*

[ Return to Index | Top of Index ]

GTO Go back to the main exhibit hall