HP Forums / **HP Calculators (and very old HP Computers)** / **General Forum** ▼ / **Comments and discussion on Valentin's 4th "Then and Now" - Area**

🔧 **NEW REPLY**

Comments and discussion on Valentin's 4th "Then and Now" - Area	Threaded Mode | Linear Mode

11th January, 2023, 10:10 (This post was last modified: 11th January, 2023 16:01 by EdS2.)   **Post: #1**

EdS2 👤
Senior Member

Posts: 525
Joined: Apr 2014

Comments and discussion on Valentin's 4th "Then and Now" - Area

This thread is an offering to anyone and everyone who would like a place to discuss or digress on Valentin's latest challenge thread:
[VA] SRC #012d - Then and Now: Area

As Valentin does set some splendid challenges but strongly prefers a clean thread for later publication, here's a place where we can discuss our approaches, show any mathematical analysis, use any tools, calculators or computers which might help us to enjoy the challenge.

My aim is only to collect those posts which are not within Valentin's rules - please continue to post in the original thread, especially with your solutions as requested over there.

I do this in a spirit of friendly collaboration, hoping that one strict thread and one loose thread will be a good combination which can keep everyone happy. As such, please no argumentation about the way Valentin runs his challenges - let's stick to the calculations, the mathematics, the analysis.

Links to tangentially related papers, discussions, videos, are all very welcome.

📧 EMAIL 📝 PM 🔍 FIND   ↩ QUOTE ⚠ REPORT

12th January, 2023, 18:27   **Post: #2**

Albert Chan 👤
Senior Member

Posts: 2,148
Joined: Jul 2018

RE: Comments and discussion on Valentin's 4th "Then and Now" - Area

Albert Chan Wrote:	(12th January, 2023 17:06)

If integrand behave like a polynomial, IBOUND likely over-estimates true error.
IBOUND was based from relative error of current result vs previous, not true result vs current.

For polynomial-like integrand, we can get by with bigger eps (= less function evaluations)

If integrand is *not* polynomial-like, even with u-transform, anything goes.

Plug to my old thread, HP71B IBOUND fooled

📧 EMAIL 📝 PM 🔍 FIND   ↩ QUOTE ⚠ REPORT

19th January, 2023, 20:57   **Post: #3**

Albert Chan 👤
Senior Member

Posts: 2,148
Joined: Jul 2018

RE: Comments and discussion on Valentin's 4th "Then and Now" - Area

Valentin just post the solution, which did include the area of the tiny speck.
Although I knew of the speck, I thought it were not OP were asking ... I guessed wrong.

Anyway, I tried this with HP Prime emulator.
integrate() use Adaptive Gaussian Quadrature, switching to Romberg if that failed.

For the main area, it work wonderfully, even with singularity inside interval.
Adpative part just zeroed in to it automatically, as needed.

```
CAS> s := sqrt(-ln(y*y/30.07+exp(-sin(y)))) :;
CAS> f := surd(y+s,3) - surd(y-s,3) :;
CAS> int(f, y, 0, 2.82740261413)
```

2.07662636775

Assuming speck is an ellipse, we already have 5+ digits accuracy.

```
CAS> c := -4.06717950603 // center of speck interval
CAS> a, b := 0.017967241585, f(y=c)/2 // major, minor axis
CAS> pi*a*b // ellipse area
```

7.19760850546e−5

The shape is simple, but Adaptive Gaussian Quadrature failed, switched to Romberg.
Romberg result (no u-substituion) is worse than above ellipse estimate.

```
CAS> int(f, y, c-a, c+a)
```

[7.19759235247e−5, 7.19761930451e−5]

The reason it failed is because the gap (2a), relative to center (c), is just too small.
Internally, integrate limit is from -1 to 1, with gaussian quadrature weights and abscissa.
When it get mapped to this tight domain, there is just not enough room. (think pigeonhole principle)

The inaccurate mapped abscissa generated fuzziness to f value, which mess up the area.
All we need is to "push out" f fuzziness. (1e-4 > area of speck)

```
CAS> int(f + 1e-4/(2a), y, c-a, c+a) - 1e-4
```

7.1976193051e−5

Post: #4

19th January, 2023, 22:43

Albert Chan 👤
Senior Member

Posts: 2,148
Joined: Jul 2018

RE: Comments and discussion on Valentin's 4th "Then and Now" - Area

> **Albert Chan Wrote:**   (19th January, 2023 20:57)
>
> The inaccurate mapped abscissa generated fuzziness to f value, which mess up the area.
> All we need is to "push out" f fuzziness. (1e-4 > area of speck)
>
> ```
> CAS> int(f + 1e-4/(2a), y, c-a, c+a) - 1e-4
> ```
>
> 7.1976193051e−5

Another way is to set epsilon higher (XCas work the same way too).

```
CAS> epsilon := 1e-11 // default was 1e-12
CAS> int(f, y, c-a, c+a)
```

7.1976193053e−5

Post: #5

19th January, 2023, 23:23 (This post was last modified: 19th January, 2023 23:26 by J-F Garnier.)

J-F Garnier 👤
Senior Member

Posts: 820
Joined: Dec 2013

RE: Comments and discussion on Valentin's 4th "Then and Now" - Area

First of all, thanks to Valentin for this new problem. It kept me busy for long hours.
Also the kind request to postpone our results for a few days was an interesting exercise.

For now, I will just make a couple of comments purely focusing on the programming aspects:

```
1   DESTROY ALL @ DIM U @ H=1/3 @ D=FNROOT(1,1,SQR(FNG(FVAR))-FVAR)
2   DISP "Area:";FNI(0,D)+FNI(D,FNY(2.83))+FNI(FNY(-4.08),FNY(-4.05))

3   DEF FNG(Y)=-LN(Y*Y/30.07+EXP(-SIN(Y))) @ DEF FNY(Y)=FNROOT(Y,Y,FNG(FVAR))
4   DEF FNI(A,B)=INTEGRAL(A,B,1/10^10,FNW(IVAR)) @ DEF FNR(X)=SGN(X)*ABS(X)^H
5   DEF FNW(Y) @ U=SQR(FNG(Y)) @ FNW=FNR(Y+U)-FNR(Y-U)
```

1. You may wonder why Valentin explicitly declared the variable U ("DIM U"), and not the others.
Don't remove this DIM statement! Its purpose is related to a HP-71B bug, still present in the latest 2CDCC version and documented here, that I can summarize as:
don't create new variables in the functions called by INTEGRAL or FNROOT, but create/declare them before.

Matter of fact, my own solution is buggy in that respect, since the X, X1 and X2 variables are created in the FNF function called by INTEGRAL:

J-F Garnier Wrote: (12th January, 2023 11:34)

```
280 DEF FNF(Y)
290 X=SQR(-LOG(FNR(Y)))
300 X1=ABS(Y-X)^(1/3) @ IF Y<X THEN X1=-X1
310 X2=ABS(Y+X)^(1/3) @ IF Y<-X THEN X2=-X2
320 FNF=X2-X1
330 END DEF
```

During my program development I manually called FNF surely several times before running the INTEGRAL function on it, so it was harmless, still to be safe X, X1 and X2 should be created before, for instance with:
`55 REAL X,X1,X2`

2. The "1/10^10" expression may look inefficient and be better replaced by 1E-10.
The problem is that the HP-71 decompiles this value in standard format as .0000000001 that is difficult to re-read. This is one of the few minor annoyances of the HP-71 BASIC.
I used the same 10^n trick for readability in my own solution.

J-F

[EMAIL] [PM] [WWW] [FIND]   [QUOTE] [REPORT]

20th January, 2023, 02:07 (This post was last modified: 20th January, 2023 14:39 by Albert Chan.)   **Post: #6**

Albert Chan

Senior Member   Posts: 2,148
Joined: Jul 2018

RE: Comments and discussion on Valentin's 4th "Then and Now" - Area

Albert Chan Wrote: (19th January, 2023 20:57)

Although I knew of the speck, I thought it were not OP were asking ... I guessed wrong.

I had already spent my 1 post in VA thread. This update included the speck too.

```
10 DESTROY ALL @ M=30.07 @ A=.831971149978 @ B=2.82740261413-A @ P=10^(-8)
20 T=1/3 @ DEF FND(Y,S)=SGN(Y+S)*ABS(Y+S)^T-SGN(Y-S)*ABS(Y-S)^T
30 DEF FNF(Y)=FND(Y,SQR(-LN(Y*Y/M+EXP(-SIN(Y)))))
40 B=B*2 @ DEF FNG(Z)=(FNF(A*Z)+FNF(A-A*Z))*A+FNF(A+B*Z)*B
50 SETTIME 0 @ I=INTEGRAL(0,1,P,FNG(.5*IVAR^3)*IVAR^2)*1.5 @ DISP I;"+";
60 DISP INTEGRAL(-4.08514674762,-4.04921226445,P,FNF(IVAR));"=";I+RES,TIME
```

```
>run
2.07662636775 + 7.19761930307E-5 = 2.07669834394     1.14
```

HP17B estimated timing = 1.14*200 = 228 s (≈ 4 minutes)

```
FNF(y) calls:    main : speck = 381 : 63 ≈ 6.05 : 1
Time spent :     main : speck = .98 : .16 ≈ 6.13 : 1
```

Update Jan 20,2023:

Above hard coded constants are obtained from another section of code.
I normally code it this way, to keep core part clean.

```
>list 999,9999
999 END
1000 DEF FNE(Y)=Y*Y/M+EXP(-SIN(Y))-1
```

```
1010 DISP "main: 0",FNROOT(.5,1,FNE(FVAR)-EXPM1(-FVAR^2)),FNROOT(2,3,FNE(FVAR))
1020 DISP "speck:",FNROOT(-4.1,-4.05,FNE(FVAR)),FNROOT(-4.05,-4,FNE(FVAR))

>run 1000 ! integral limits
main: 0      .831971149978      2.82740261413
speck:      -4.08514674762      -4.04921226445
```

Valentin Albillo
Senior Member

Posts: 970
Joined: Feb 2015
Warning Level: 0%

RE: Comments and discussion on Valentin's 4th "Then and Now" - Area

.

Hi, **J-F**,

> **J-F Garnier Wrote:**   (19th January, 2023 23:23)
>
> First of all, thanks to Valentin for this new problem. It kept me busy for long hours.

You're welcome. Glad you liked it.

> **Quote:**
>
> Also the kind request to postpone our results for a few days was an interesting exercise.

Not everyone agreed. Someone posted his solution before the stated time limit, which I'm sure you didn't like at all, though I immediately posted a message to let the person know and he kindly deleted the post.

Also, some other person essentially complained that he much preferred to see tentative post after tentative post, refining the solution all the time, and learning from the refining process, which with the delay and the *suggestion* (**not** mandatory rule) that people would do best refining their initial solutions *before* posting them, to reduce the clutter in the thread, essentially prevented that learning process for him.

This being so, I'm not sure if the benefits outdo the disadvantages, so perhaps I will reinstate the 2-3 day delay for future problems, or perhaps I won't. What do you think ?

> **Quote:**
>
> 1. You may wonder why Valentin explicitly declared the variable U ("DIM U"), and not the others. Don't remove this DIM statement! Its purpose is related to a HP-71B bug, still present in the latest 2CDCC version and documented here, that I can summarize as: don't create new variables in the functions called by INTEGRAL or FNROOT, but create/declare them before.

Correct. I mentioned it in my original solution's post, I quote:

> *"Line 1: [...] **DIM U** is necessary to create variable **U** here and not inside **DEF FNW**, where we would fall prey to a well-known system bug."*

> **Quote:**
>
> The "1/10^10" expression may look inefficient and be better replaced by 1E-10.
> The problem is that the HP-71 decompiles this value in standard format as .0000000001 that is difficult to re-read.

Correct. On the positive side, though entering **1E-10** is decompiled as **.0000000001** (which certainly looks *ungainly* and may cause the line to *exceed* maximum length,) it has the *benefit* that it takes **4** *less bytes* of RAM (my program would get shortened from *244 bytes* to just *240 bytes*,) and also executes somewhat *faster*.

> **Albert Chan Wrote:**
>
> **I had already spent my 1 post** in VA thread.

Let's see. First of all, you posted **3** times to the thread, not **1**, which is perfectly Ok by me, and in any case I posted the following:

> *"Wait instead until next Wednesday 9:00 pm GMT+1 so that other people will have a chance. In the meantime you can mull it over so that you eventually post the one (1) message featuring your fully refined solution instead of a*

myriad posts refining it little by little."

but it's just a *suggestion*, *not* a mandatory rule, so you hadn't already spent anything, you could have posted again, no problem there. On the other hand, you're one of the *"usual suspects"* that tend to post the aforequoted *"myriad"* posts with marginal improvements (which at times, *aren't*) instead of thinking it over and suitably refining your approach *before* posting it. I can cite many threads where you've done exactly that, for no good reason that I can discern other that you seem to be *itching* to do it.

Don't take my advice if you don't want to and please *don't take offence*, but I think that your efforts and time would be *best* spent creting a very good, *refined* post, properly *formatted*, for the benefit of the readers if nothing else.

Also, in your code above, namely:

*10 DESTROY ALL @ M=30.07 @ **A=.831971149978** @ **B=2.82740261413-A** @ P=10^(-8)*
20 T=1/3 @ DEF FND(Y,S)=SGN(Y+S)*ABS(Y+S)^T-SGN(Y-S)*ABS(Y-S)^T
[...]
*60 DISP INTEGRAL(**-4.08514674762,-4.04921226445**.P,FNF(IVAR));"=";I+RES,TIME"*

you are using **four full-accuracy values that come out of nowhere**, the program does not compute them in any way, not even *refining d.dd approximations as suggested in my OP*. You just stuck the numbers there, and any user looking at your code later wouldn't have the slightest idea of the origin of those values. Your program gets them *out of the blue*.

Last, in one of your posts in my main thread you *begin* your post with this, I quote:

"exp(-((x-d)^3-y)^2) > (R = y^2/M + exp(-sin(y)))   // Given: M=30.07, d=1.596

Let s = sqrt(-log(R)) ≥ 0

((x-d)^3-y)^2 < s^2
y-s < (x-d)^3 < y+s

x is real if s is real --> R ≤ 1 --> 0 ≤ y ≤ 2.82740261413

Height, $f(y) = x_2 - x_1 = \sqrt[3]{y+s} - \sqrt[3]{y-s}$

f slope is infinite when y = s --> R = exp(-y^2) --> y = 0 or 0.831971149978

*Let a = 0.831971149978, a+**b** = a+**B**/2= 2.82740261413"*

where you use variables **b** and **B**/2, but up to that point you haven't *defined* them nor *assigned* any value to them since the beginning of the post, so once again they come *out of the blue*. IMHO, you shouldn't do that, i.e., using variables before you define them and/or before assigning values to them, it's bad practice and likely to confuse the reader.

My honest, well-meaning advice would be: Do not be so eager to spend your time posting message after message, instead do try to refine your solution *before* posting anything, also don't include constants out of the blue, and last but certainly not least, spend a decent amount of time *properly formatting* your explanations and improving the *understandability* of your expressions.

Remember, the quality of your posts ultimately depends not only on the quality of your *math* but equally important, also depends on the quality of the *presentation* of your results.

Hope it helps. Thanks for your interest in my challenges *("Problems")* and for your valuable contributions.

Regards.
V.

J-F Garnier 🔒
Senior Member

RE: Comments and discussion on Valentin's 4th "Then and Now" - Area

> **Valentin Albillo Wrote:** (20th January, 2023 03:56)
>
> > **Quote:**
> > Also the kind request to postpone our results for a few days was an interesting exercise.
>
> ... some other person essentially complained that he much preferred to see tentative post after tentative post, refining the solution all the time, and learning from the refining process...
> This being so, I'm not sure if the benefits outdo the disadvantages, so perhaps I will reinstate the 2-3 day delay for future problems, or perhaps I won't. What do you think ?

There are disadvantages for both. Respecting his delay avoid to prematurely spoil the game.
But for difficult problems it's sometime nice to see other's attempts to progress toward a solution.
I would say that a 2-3 day delay is fine, it's not excessive, and we can still have the opportunity to share partial results towards a complete solution.
Actually, this is what happened here, Fernando and Werner identified the 'tiny area', Albert and I gave the correct 12-digit result (by different ways) of the main area, and I finally merged everything together.

Also, we should keep in mind that if some of us are can afford to spent a lot of time on a problem each day, it's not necessary the case for everybody.

> **Quote:**
>
> I mentioned it in my original solution's post, I quote:
>
> > *"**Line 1**: [...] **DIM U** is necessary to create variable **U** here and not inside **DEF FNW**, where we would fall prey to a well-known system bug."*

Sorry, I missed that comment, anyway it was not bad to insist on this bug that not every HP-71B user may know or remember.

J-F

Albert Chan
Senior Member

RE: Comments and discussion on Valentin's 4th "Then and Now" - Area

> **J-F Garnier Wrote:**   (20th January, 2023 11:12)
>
> There are disadvantages for both. Respecting his delay avoid to prematurely spoil the game.
> But for difficult problems it's sometime nice to see other's attempts to progress toward a solution.
> I would say that a 2-3 day delay is fine, it's not excessive, and we can still have the opportunity to share partial results towards a complete solution.

I vote for no delay, and less rules, to encourage brainstorming among members.
Many times, OT ideas are more interesting than original puzzle!

The delay does not avoid spoiling the game ... only delay it.
Whoever post solution first, already spoil the game ... but someone has to do it!

Valentin Albillo
Senior Member

RE: Comments and discussion on Valentin's 4th "Then and Now" - Area

.
Hi, **J-F Garnier** and **Albert Chan**,

> **J-F Garnier Wrote:**
>
> [...] for difficult problems it's sometime nice to see other's attempts to progress toward a solution. **I would say that a 2-3 day delay is fine**, it's not excessive, and we can still have the opportunity to share partial results towards a complete solution.
>
> Actually, this is what happened here, Fernando and Werner identified the 'tiny area', Albert and I gave the correct 12-digit result (by different ways) of the main area, and I finally merged everything together.

I see. Thanks for your reasoned opinion, I think that I'll tentatively make do for a *2-day delay* and I will see how it works. If most people get used to it, fine. If most people resent it, there will be no delays and thus the earliest bird will catch the worm.

> **J-F Garnier Wrote:**

Sorry, I missed that comment, anyway **it was not bad to insist on this bug that not every HP-71B user may know** or remember.[/b]

I agree. I said *"well-known"* but on second thought it might not actually be the case. Besides **DIM U**, you can also use **U=0**, say, the important bit being that variable **U** is created outside of the *UDF*.

> **Albert Chan Wrote:**
> I vote for **no delay**, [...]

Of course you do. Your vote is duly noted.

> **Albert Chan Wrote:**
> [...] and **less rules** [...]

Oh really ! And which rules would you want to see eliminated ? Perhaps

> **NO** XCAS, MATHEMATICA, MAPLE, EXCEL, C/C++/C#, PYTHON, LUA, etc.,

or may be

> **NO LENGTHY MATH** SESSIONS

?

> **Albert Chan Wrote:**
> **Many times**, OT ideas are **more interesting than original puzzle!**

Is that so ? My, my ... 🙁 . Well, feel free to *stop* participating in my less interesting puzzles and post instead *your own creations*, surely you'll find them interesting enough.

> **Albert Chan Wrote:**
> Whoever post solution first, already spoil the game ... **but someone has to do it!**

That much is **obvious**. But *silly truisms* aside, the delay is useful for the purpose I intend, which is to give interested but currently busy people a 2-day window of opportunity to try and work on their own solutions with no spoilers on sight.

Anyway, **Albert Chan**, I did notice that you *didn't reply to my post* but to **J-F Garnier**'s, and matter of fact you didn't even deign to reply to *any* of the points and well-meant advice I gave you re your posts, so rest assured that in the future I won't offer again any further comments or advice to you re your productions.

V.

PM | WWW | FIND | EDIT | X | QUOTE | REPORT

21st January, 2023, 03:25 Post: #11

Albert Chan
Senior Member

Posts: 2,148
Joined: Jul 2018

RE: Comments and discussion on Valentin's 4th "Then and Now" - Area

> **Valentin Albillo Wrote:** (21st January, 2023 02:34)
>
> > **Albert Chan Wrote:**
> > **Many times**, OT ideas are **more interesting than original puzzle!**
>
> **Is that so ?** My, my ... 🙁 . Well, feel free to *stop* participating in my less interesting puzzles and post instead *your own creations*, surely you'll find them interesting enough.

Please don't take it the wrong way. I did not meant *your* puzzles. They are always great.

In fact, I joined HP forum just to join in the math discussion.
My very first post here, dated July 20, 2018, for **[VA] SRC#001 - Spiky Integral**

> **Quote:**
> Anyway, **Albert Chan**, I did notice that you *didn't reply to my post* ...

I believe action is better than words.
I had already updated the post regarding the "magic constants" issues.

Thanks for the advice. I'll try to improve presentation of my posts in the future.

EdS2 🔒
Senior Member

Posts: 525
Joined: Apr 2014

RE: Comments and discussion on Valentin's 4th "Then and Now" - Area

I rather hope that the next Valentin challenge will come with a cross-linked comments-and-discussion thread like this - the two of us can coordinate.

And my hope is very much that we have both good things: a thread for the business of the challenge, and a thread for related business. And then we don't have to worry so much about what we post, only about where we post it.

It remains impossible, I'm sure, to please everyone by choosing a precise time interval between the posting of the challenge and the encouraging of sharing of solutions. Any time interval will be too short for some and too long for others. But still we must choose something, zero or more days, and indeed it's Valentin who gets to choose.

What I would recommend is to try to accommodate people who don't visit too often, and particularly I'd recommend posting the challenge in one week, allowing a weekend to pass, and allowing the free discussion the following week. This could be as short an interval as Friday evening through to Monday morning, in some timezone.

I think it would help if the posted challenge includes the date and time for the thread to open for submissions and contributions.

J-F Garnier 🔒
Senior Member

Posts: 820
Joined: Dec 2013

RE: Comments and discussion on Valentin's 4th "Then and Now" - Area

I want to discuss here one aspect of the challenge, computing the integral:

> **Valentin Albillo Wrote:**   (19th January, 2023 19:08)
>
> The integral for the big island, however, has **two** *singularities*, one at **y = 0** (which doesn't cause any problems as this is the **ymin** extreme and the *Width* function being integrated is never evaluated there by **INTEGRAL**,) and another *inside* the range [**ymin**, **ymax**], which causes **INTEGRAL** to be at least *400% slower* and even so it can only produce a value accurate to at most *7-8 digits*, not *12*.
>
> [...] we can then *split* the single problematic integral into two parts, the singularities being now at the extremes of the ranges of integration where they do no harm, like this:
>
> $$Area = \int_{y\min}^{ysng} Width(y).dy + \int_{ysng}^{y\max} Width(y).dy$$

My observations showed me that the singularities *do harm* actually, both in *execution time and accuracy*.

To illustrate the effect of the singularity at y=0, let's consider this simpler integral:
I=INTEGRAL(0,1,E,IVAR^(1/6))
that reproduces the behaviour of Valentin's integral near y=0.
The exact value of this integral is 6/7 = 0.857142857143

Here is what happens when trying to evaluate this integral with the 71B with increasing target accuracies:

```
>FOR N=6 to 12 @ I=INTEGRAL(0,1,10^(-N),IVAR^(1/6)) @ DISP N;I;IBOUND;ABS(I-6/7) @ NEXT N
 N   INTEGRAL        IBOUND            ERROR
 6 .857143021890 8.57147612555E-7  .000000164747
 7 .857142864063 8.57143170419E-8  .000000006920
 8 .857142858515 8.57142936872E-9  .000000001372
 9 .857142857195 8.57142862245E-10 .000000000052
10 .857142857151 8.57142858410E-11 .000000000008
11 .857142857127 8.57142857257E-12 .000000000016
12 .857142857127 -8.57142857257E-13 .000000000016
```

The accuracy is not improved beyond 1E-10, even if IBOUND seems to indicate better accuracy. At 1E-12, the negative IBOUND indicates that convergence is not detected within the limit of 32768 samples.

The same occurs with Free42 for this example, where the effective accuracy is limited to about 1E-14, far from the 34-digit limit of the arithmetic.

I solved the issue by splitting the main integral into 4 pieces in my solution.
In that way, it is possible to concentrate the samples in the regions that need them, and avoid wasting time with useless samples in the other regions.

J-F

21st January, 2023, 17:53   **Post: #14**

Albert Chan
Senior Member   Posts: 2,148
                                                          Joined: Jul 2018

RE: Comments and discussion on Valentin's 4th "Then and Now" - Area

> **J-F Garnier Wrote:** (21st January, 2023 12:15)
>
> I solved the issue by splitting the main integral into 4 pieces in my solution.
> In that way, it is possible to concentrate the samples in the regions that need them, and avoid wasting time with useless samples in the other regions.

This is what HP Prime integrate does, by zeroed-in to the region that need them.
However, this adaptive routine does not work well with fuzzy numbers. (see here)

> **Albert Chan Wrote:** (12th January, 2023 00:17)
>
> Let a = 0.831971149978, a+b = a+B/2 = 2.82740261413
> Infinite slopes at y = 0 and a, moved to z = 0
>
> $$\text{Area} = \int_0^{a+b} f(y)\, dy = \int_0^{1/2} \Big[\big(f(a\,z) + f(a - a\,z) \big) \cdot a \;+\; f(a + B\,z) \cdot B \Big] dz$$
>
> Let g(z) = RHS integrand, and substitute z = x³/2, to make z=0 infinite slope, down to 0.
> INTEGRAL built-in u-transform should turn curve to bell-shaped, easy to integrate.   (*)
>
> $$\text{Area} = \int_0^{1/2} g(z)\, dz = \int_0^1 g\!\left(\frac{x^3}{2}\right) \cdot \left(\frac{3}{2} x^2 \, dx \right)$$

My approach was by folding back to 1 integrand, with really bad region to the left.
It has the advantage of flattening the curve, and possibly cancelled out singularities a bit.

Except the ends, folded curve look like a straight line, with dowward slope: ⌐ ⌐⌐ = ⌐ + ⌐ + ⌐ ≈ ＼
We can even estimate size of main area by 1 g(z) evaluation

>FNG(0.25) * 0.5
2.07508490846

BTW, a+b = a+B/2 = 2.82740261413 meant b = 2.82740261413-a, B = 2*b
Better: (b,B) such that the relation is true ... sometimes it is hard to isolate variable.

Thanks to JFG analysis, we have: $f(\varepsilon) \approx 2 * \sqrt[6]{\varepsilon}$
Based from plots, I have the other: $f(a \pm \varepsilon) \approx f(a) - 7/8 * \sqrt[3]{\pm\varepsilon}$

→ $g(\varepsilon) \approx f(a)*(a+B) + (2a) \sqrt[6]{\varepsilon} - (B-a)*(7/8) \sqrt[3]{\varepsilon} \approx 5.715 + 1.664 \sqrt[6]{\varepsilon} - 2.764 \sqrt[3]{\varepsilon}$

Note that sign of ε terms are opposite, cancelling each other somewhat.
$g(\varepsilon)$ started curving downward when $\varepsilon > 0.0000207$

21st January, 2023, 18:16   **Post: #15**

J-F Garnier
Senior Member   Posts: 820
                                                          Joined: Dec 2013

RE: Comments and discussion on Valentin's 4th "Then and Now" - Area

I had the idea that it may be possible to write an adaptive integration code on the 71B by first analysing the integrand behaviour, actually I'm surprised not to find anything like that in the forum or various 71B code archives (I didn't do a extensive search though). Maybe is it just too complex, or just with no real usage.

I read this part of your posts, but honestly didn't understand how you devised the RHS expression from f(y).
I guess this is classic math, but after all Valentin's challenges have always been the right place to learn something, so if you could explain it a bit, it would be appreciated.

J-F

21st January, 2023, 18:41 **Post: #16**

Werner
Senior Member

Posts: 777
Joined: Dec 2013

RE: Comments and discussion on Valentin's 4th "Then and Now" - Area

Thanks J-F for making me feel a bit less stupid, as I couldn't figure that one out either!
Cheers, Werner

21st January, 2023, 19:07 **Post: #17**

Albert Chan
Senior Member

Posts: 2,148
Joined: Jul 2018

RE: Comments and discussion on Valentin's 4th "Then and Now" - Area

First we write as 3 mini-integrals, with bad region move to the left (lower limit)

$$\int_0^{a+b} = \int_0^{a/2} - \int_a^{a/2} + \int_a^{a+b}$$

Second, let y = c1+c2*z, dy = c2 dz, so that integrand limits matched.
We could use any limits. I picked 0 .. 0.5, thus needed B = 2b.

$$\int_0^{a/2} f(y)\,dy = \int_0^{1/2} f(a\,z)\,(a\,dz)$$

$$-\int_a^{a/2} f(y)\,dy = \int_0^{1/2} -f(a - a\,z)\,(-a\,dz)$$

$$\int_a^{a+\frac{B}{2}} f(y)\, dy = \int_0^{1/2} f(a + B\,z)\,(B\,dz)$$

Add it all up, we have:

$$\int_0^{a+b} f(y)\, dy = \int_0^{1/2} \left[\big(f(a\,z) + f(a - a\,z)\big) \cdot a \ + \ f(a + B\,z) \cdot B \right] dz$$

21st January, 2023, 22:06   **Post: #18**

Albert Chan

Senior Member

Posts: 2,148
Joined: Jul 2018

RE: Comments and discussion on Valentin's 4th "Then and Now" - Area

> **Werner Wrote:** (21st January, 2023 18:41)
>
> > **J-F Garnier Wrote:** (21st January, 2023 18:16)
> >
> > I read this part of your posts, but honestly didn't understand how you devised the RHS expression from f(y).
>
> Thanks J-F for making me feel a bit less stupid, as I couldn't figure that one out either!

Sorry about this.

My original draft spelled out the steps, but I was not sure about breaking LENGTHY MATH SESSION rule.
(You may have noticed my recent math posts are getting very short ... sometimes too short)

I had also considered breaking up main area into more natural 2 pieces, and apply u-transformation.

$$\int_a^{a+b} = \int_0^a + \int_a^{a+b}$$

Code is more readable, because this does not involve "back folding".
But, I may need math to explain how u-substitution work, and why I need it here.
INTEGRAL built-in u-sub is not enough. (that's why JFG sum main area from 4 pieces)

To keep post short, I gave up on that idea.

The code for getting integral limits were also in the draft. (*)
I trimmed that too. Wrong move 🙁

(*) the code is in this thread

22nd January, 2023, 01:16   **Post: #19**

Albert Chan

Senior Member

Posts: 2,148
Joined: Jul 2018

RE: Comments and discussion on Valentin's 4th "Then and Now" - Area

> **J-F Garnier Wrote:** (21st January, 2023 18:16)
>
> I had the idea that it may be possible to write an adaptive integration code on the 71B by first analysing the integrand behaviour, actually I'm surprised not to find anything like that in the forum or various 71B code archives (I didn't do a extensive search though). Maybe is it just too complex, or just with no real usage.

I did Adaptive Simpson HP71B code a few months ago.
With that loaded in memory, we update function, limits, and eps.

```
>10 X1=0 @ X3=2.82740261413 @ E=10^(-8)
>50 SUB F(X,Y) @ S=SQR(-LN(X*X/30.07+EXP(-SIN(X))))
>60 Y=SGN(X+S)*ABS(X+S)^(1/3)-SGN(X-S)*ABS(X-S)^(1/3) @ STOP
>run
 2.07662636771     8.96 ! @200x, about 30 minutes for HP71B
```

SIMPSON can't see user defined function, so I coded with slower CALL F(X,Y)

Based on timings alone, I guessed F(X,Y) called about 3000 times.
I don't know how to count F(X,Y) calls. Variables inside F turned local. Any ideas?

22nd January, 2023, 08:54 (This post was last modified: 22nd January, 2023 09:01 by C.Ret.)   **Post: #20**

C.Ret
Member

Posts: 230
Joined: Dec 2013

RE: Comments and discussion on Valentin's 4th "Then and Now" - Area

Hello Albert Chan,

Simply add the counter variable C to the list of call indices: SUB SIMPSON(X1,X3,X5,F1,F3,F5,E,S,C) and increment C in Simpson's procedure each time it calls the function F.

Alternatively, add the counter in the definition of F and its calls CALL F(X,Y,C) but I'm afraid that will slow down the process further.

Have a nice sunday.

22nd January, 2023, 11:02   **Post: #21**

EdS2
Senior Member

Posts: 525
Joined: Apr 2014

RE: Comments and discussion on Valentin's 4th "Then and Now" - Area

(In these Comments and Discussion threads, of which this is the first, LENGTHY MATHS SESSIONS are welcome - provided of course the intent is to spread knowledge and have an enjoyable exchange of ideas. It's especially useful, I think, to adopt the convention of showing long decimal results with the correct and incorrect digits emphasised in some way: part of the interest with calculators, I think, is how much accuracy you really have, even if you always see lots of digits.)

22nd January, 2023, 13:27   **Post: #22**

Albert Chan
Senior Member

Posts: 2,148
Joined: Jul 2018

RE: Comments and discussion on Valentin's 4th "Then and Now" - Area

> **C.Ret Wrote:**   (22nd January, 2023 08:54)
>
> Simply add the counter variable C to the list of call indices: SUB SIMPSON(X1,X3,X5,F1,F3,F5,E,S,C) and increment C in Simpson's procedure each time it calls the function F.

Thanks! I was hoping for something similar to Python "global c", but this tip work!
Here is full listing, integrating main area.

10 X1=0 @ X3=2.82740261413 @ E=10^(-8)
20 T=TIME @ X2=(X3+X1)/2 @ CALL F(X1,F1) @ CALL F(X2,F2) @ CALL F(X3,F3)
30 S=(F1+4*F2+F3)*(X3-X1)/6 @ C=3 ! F calls counter
40 CALL SIMPSON(X1,X2,X3,F1,F2,F3,32*E,S,C) @ DISP S,C,TIME-T @ END

50 SUB F(X,Y) @ S=SQR(-LN(X*X/30.07+EXP(-SIN(X))))
60 Y=SGN(X+S)*ABS(X+S)^(1/3)-SGN(X-S)*ABS(X-S)^(1/3)

100 SUB SIMPSON(X1,X3,X5,F1,F3,F5,E,S,C)
110 H=(X5-X1)/4 @ X2=X1+H @ X4=X5-H
120 CALL F(X2,F2) @ CALL F(X4,F4) @ C=C+2
130 H=H/3 @ S1=(F1+4*F2+F3)*H @ S2=(F3+4*F4+F5)*H
140 S3=S1+S2 @ D=S3-S @ H=E/2
150 IF H<1.E-15 OR ABS(D)<H THEN S=S3+D/15 @ STOP
160 CALL SIMPSON(X1,X2,X3,F1,F2,F3,H,S1,C)
170 CALL SIMPSON(X3,X4,X5,F3,F4,F5,H,S2,C)
180 S=S1+S2

>run
 2.07662636771   2601   9.13 ! @200x, HP71B about 30 minutes

Overhead of adding counter = 9.13/8.96 - 1 ≈ 2%, not too bad
SIMPSON calls = ((F calls) - 3)/2 = 1299

Above accuracy is a fluke. Without extended precision, best we could hope for is about 10 digits.
E (max absolute error) should setup to match this. Smaller E may make the integral estimate worse.

Albert Chan 👤   Posts: 2,148
Senior Member   Joined: Jul 2018

RE: Comments and discussion on Valentin's 4th "Then and Now" - Area

> **Albert Chan Wrote:**   (21st January, 2023 22:06)
>
> I had also considered breaking up main area into more natural 2 pieces, and apply u-transformation.
>
> $$\int_a^{a+b} = \int_0^a + \int_a^{a+b}$$
>
> Code is more readable, because this does not involve "back folding".
> But, I may need math to explain how u-substitution work, and why I need it here.
> INTEGRAL built-in u-sub is not enough. (that's why JFG sum main area from 4 pieces)
>
> To keep post short, I gave up on that idea.

This was my 2-splits main area solution, as opposed to 3-splits: $\int_0^{a+b} = \int_0^{a/2} - \int_a^{a/2} + \int_a^{b}$

10 M=30.07 @ A=.831971149978 @ B=2.82740261413-A @ P=10^(-9)
20 T=1/3 @ DEF FND(Y,S)=SGN(Y+S)*ABS(Y+S)^T-SGN(Y-S)*ABS(Y-S)^T
30 DEF FNF(Y)=FND(Y,SQR(-LN(Y*Y/M+EXP(-SIN(Y)))))
40 DEF FNG(Z)=FNF(A*Z)*A+FNF(A+B*Z)*B
45 DEF FNU(U,U2)=FNG(U2*U*(4-3*U))*U2*(1-U)*12
50 SETTIME 0 @ DISP INTEGRAL(0,1,P,FNU(IVAR,IVAR*IVAR)),TIME

>run
 2.07662636771   .71   ! @200x, HP71B = 142 seconds
>p=1e-10 @ run 50
 2.07662636775   1.42   ! @200x, HP71B = 284 seconds

Both 3-splits and 2-splits are equally good.

3-splits use 3*127 = 381 FNF() calls to get 12 accurate digits.
2-splits use 2*127 = 254 calls to get 11+ digits, 2*255 = 510 to get full 12.

We wanted u-substitution to produce polynomial-like integrand, easy for INTEGRAL to handle.
Let x = U, where U is a function of u

$$\int_0^1 g(x)\, dx = \int_0^1 g(U)\,(U'\, du)$$

Based from plots, I had guessed g(x) both ends curve like cube root, left side more extreme.
To transform to polynomial like integrand, I cube left edge, and square the other.
We start from derivatives: (u^3)' = 3**u^2**, ((1-u)^2)' = -2**(1-u)**

$$\to U' = K\, u^2\,(1-u)$$

For u = 0 .. 1, U' has sign of K, a non-zero constant --> x and u are one-to-one.

$$\to U = \int U' du = K\left(\frac{u^3}{3} - \frac{u^4}{4}\right) + C$$

Matching integral limits, x=0 → u=0, x=1 → u=1, we have K = 12, C = 0

$$\to U = u^3\,(4 - 3u)\quad,\quad U' = 12u^2\,(1-u)$$

NOTE: U does not have to be a polynomial of u. see Kahan's Q(x) example

Gjermund Skailand ♟
Member

RE: Comments and discussion on Valentin's 4th "Then and Now" - Area

I found that double exponential integration is also well suited to evaluate the integrals.
With pre-calculated boundaries, and the DM42 running on batteries I can evaluate the integrals in about 3 seconds to 12 digits accuracy using only 169 function evaluations.
Reported relative accuracy of 6.8E-13 compared to actual relative accuracy of 3.7E-13 is also good.

However, when I increase the required accuracy to e.g 1E-20, then in 7.2 seconds, using only 403 function evaluations, I get a reported relative accuracy of 2.7E-26, but comparing with
2.07669834394 7601111175686517522841 (double exponential)
2.07669834394 76059651.. (Valentin Free42, standard integration?)

the result only agree up to 14-15 digits.

Pushing the limits to e.g 1E-30 does results in a reported accuracy of 4E-32, but the results does not really change.
2.07669834394 7601111175686517522 91 (16.7 seconds, 941 function evaluations)

So it is possible that the transformations cause a lot of cancellation but we got 32 digits to work with, and the requirement was 12 digits, which is quickly calculated in a few seconds without excessive evaluations.
Use is similar to using standard integrator, the DE-integration routine takes all inputs from standard 4-level stack
T: lower limit
Z: upper limit
Y: "function"
X: "X", variable to integrate
and returns
Y: error estimate
X: result
I have posted the DE program in .raw format with some description in the "General software library"
https://www.hpmuseum.org/forum/thread-19547.html
Thanks to Didier for the zip tip.
best regards
Gjermund

`EMAIL`   `PM`   `FIND`   `QUOTE`   `REPORT`

Albert Chan ♟
Senior Member

RE: Comments and discussion on Valentin's 4th "Then and Now" - Area

> **Gjermund Skailand Wrote:**   (12th February, 2023 19:00)
>
> I found that double exponential integration is also well suited to evaluate the integrals.
> ...
> Pushing the limits to e.g 1E-30 does results in a reported accuracy of 4E-32, but the results does not really change.
> 2.07669834394 7601111175686517522 91 (16.7 seconds, 941 function evaluations)

Thanks for DE integrate code!

A weakness with DE algorithm is we cannot have fuzzy integrand.
Above calculated area only had about half-precision accuracy.

Let's compare the same calculations, fuzzy vs "sharp" integrand:

```
>>> from mpmath import *
>>> mp.pretty = 1
>>> mp.dps = 34
>>> fnc = lambda x: sign(x)*abs(x)**(1/3)
>>> fnd = lambda y,s: fnc(y+s) - fnc(y-s)
>>> fne = lambda y: y*y*100/3007 + expm1(-sin(y))
>>> fnf = lambda y: fnd(y, sqrt(-log1p(fne(y))))
>>> main = [0, findroot(lambda y: fne(y)-expm1(-y*y),[0.5,1]), findroot(fne,[2.5,3])]
>>> speck = [findroot(fne, [-4.09,-4.08]), findroot(fne, [-4.05,-4.04])]

>>> main
[0, 0.8319711499790767979993060165734035, 2.827402614129560091868484044583741]
>>> speck
[-4.085146747638309747433733595149126, -4.049212264396950476554495084637107]
```

Note: mpmath quad use DE algorithm by default. Also, main had 2 parts, treated as 2 integrals.

>>> quad(fnf, main) + quad(fnf, speck)
2.0766983439476010079521669308070185

The fuzziness comes from catastrophic cancellation of fnd
We redefine fnd, using identity: a^3 - b^3 = (a-b) * (a^2 + a*b + b^2)

>>> fnd = lambda y,s: 2*s/(fnc((y+s)**2) + fnc((y+s)*(y-s)) + fnc((y-s)**2))

>>> quad(fnf, main) + quad(fnf, speck)
2.0766983439476011169386367570011992

13th February, 2023, 09:59   **Post: #26**

Gjermund Skailand
Member

Posts: 53
Joined: Dec 2013

RE: Comments and discussion on Valentin's 4th "Then and Now" - Area

Thanks Albert,

I was hoping for insight on the problem, and wishing for someone to calculate a result to high precision.
PS I did split the large area into two, so I actually did calculate 3 integrals to get the results.
br Gjermund

14th February, 2023, 15:29   **Post: #27**

Albert Chan
Senior Member

Posts: 2,148
Joined: Jul 2018

RE: Comments and discussion on Valentin's 4th "Then and Now" - Area

Hi, Gjermund Skailand

My numbers were wrong!

I was using sign(x) * abs(x) ** (1/3) for cube root, but (1/3) evaluated as Python float.
I should have used mpf(1)/3 exponent, or simply use mpmath cbrt(x) = exp(log(x)/3)

For total area, fnd cancellation errors does not matter.

>>> fnc = lambda x: sign(x)***cbrt**(abs(x))
>>> fnd = lambda y,s: fnc(y+s) - fnc(y-s)
>>> quad(fnf, main) + quad(fnf, speck)
2.0766983439476011111175686517522914

Now, it match your numbers, almost exactly. Sorry for the noise.

14th February, 2023, 22:52 (This post was last modified: 14th February, 2023 22:52 by pier4r.)   **Post: #28**

pier4r
Senior Member

Posts: 2,230
Joined: Nov 2014

RE: Comments and discussion on Valentin's 4th "Then and Now" - Area

> **Valentin Albillo Wrote:**   (20th January, 2023 03:56)
>
> which with the delay and the *suggestion* (**not** mandatory rule) that people would do best refining their initial solutions *before* posting them, to reduce the clutter in the thread, essentially prevented that learning process for him.
>
> This being so, I'm not sure if the benefits outdo the disadvantages, so perhaps I will reinstate the 2-3 day delay for future problems, or perhaps I won't. What do you think ?

Since the forum is flexible and EdS2 (or anyone else) can create a sibling thread for "things that do not fit the rules", they could post the iterative refinement in such sibling topics, and only the final solution in the original one.
I agree that reading solutions that grow, like learning the historical process of some scientific/mathematical discoveries , may be easier to understand things.

Though one can have both, the process with the scaffolding, in one thread, and "final solution only, without the

scaffolding" in the other (quoting "no self-respecting architect leaves the scaffolding in place after completing the building" of Gauss).

But that's my suggestion.

PS: I disagree with Gauss for the motives mentioned, from the scaffolding one can learn, because not everyone can make large comprehension steps at once, so seeing more steps could be beneficial.

14th February, 2023, 22:54 (This post was last modified: 14th February, 2023 22:56 by pier4r.)   **Post: #29**

pier4r
Senior Member

Posts: 2,230
Joined: Nov 2014

RE: Comments and discussion on Valentin's 4th "Then and Now" - Area

> **Albert Chan Wrote:**   (20th January, 2023 14:30)
>
> I vote for no delay, and less rules, to encourage brainstorming among members.
> Many times, OT ideas are more interesting than original puzzle!
>
> The delay does not avoid spoiling the game ... only delay it.
> Whoever post solution first, already spoil the game ... but someone has to do it!

But for that there are the sibling threads, it is perfectly fine (and I'd say even more ordered). Opening a new thread costs like, dunno, 5 minutes? After all the effort that Valentin (or whoever would do the same, but so far mostly Valentin does it) puts in the posts, I find opening a sibling thread very easy in comparison.

15th February, 2023, 09:54   **Post: #30**

EdS2
Senior Member

Posts: 525
Joined: Apr 2014

RE: Comments and discussion on Valentin's 4th "Then and Now" - Area

It is true, and fortunate, that both Valentin and Albert are (by their responses) happy with the sibling threads idea.

It's also true that in the latest challenge offering, Valentin has chosen not to request a delay.

So, I think, and hope, there is no particular need to re-open the question. Perhaps I could respectfully suggest another thread, a new one, for discussing the merits and optimisation of how to corral or constrain the contributions to challenge threads. I'd say it won't be of interest to everyone who is interested in discussing the challenges themselves - it's a speciality subject, on the design of challenges.

« Next Oldest | Next Newest »   Enter Keywords   Search Thread

NEW REPLY

View a Printable Version

Send this Thread to a Friend

Subscribe to this thread

User(s) browsing this thread: Valentin Albillo*

Contact Us | The Museum of HP Calculators | Return to Top | Return to Content | Lite (Archive) Mode | RSS Syndication   English (American)   Go

Forum software: MyBB, © 2002-2023 MyBB Group.