FI – 3-point Gaussian Integration

Abstract

FI is a program written in 1980 for the HP-41C programmable calculator to evaluate the definite integral between given limits of an arbitrary user-supplied function f(x) using the 3-point Gauss-Legendre quadrature formula applied over a number of subintervals. Three worked examples are included.

Keywords: *definite integration, Gauss-Legendre quadrature, programmable calculator, RPN, HP-41C, HP-41CV, HP-41CX, HP42S*

1. Introduction

FI is a short *(56 steps)* RPN program that I wrote in 1980 for the *HP-41C* programmable calculator (will also run *as-is* in the *HP-41CV/CX* and the *HP42S)*, which can evaluate the definite integral between specified limits of an arbitrary user-supplied function *f(x)* using the fast *3-point Gauss-Legendre* quadrature formula applied over a given number of subintervals. The method is as follows: we want to compute

$$I = \int_a^b f(x)\,.\,dx$$

but first of all the change of variable $x = \frac{1}{2}(b+a) + \frac{1}{2}(b-a)t$, $dx = \frac{1}{2}(b-a).dt$ transforms the interval *(a, b)* into the interval *(-1, 1).* The *3-point Gauss-Legendre* quadrature formula then gives:

$$\int_{-1}^{1} f(x)\,.\,dx = \frac{8}{9}f(0) + \frac{5}{9}\ (f\left(\sqrt{\frac{3}{5}}\right) + f\left(-\sqrt{\frac{3}{5}}\right))$$

which is *exact* for polynomial *f(x)* up to the 5^{th} degree, and a 5^{th}-order approximation otherwise. See **Notes**.

2. Program Listing

```
01 ♦LBL "FI"
02 "NAME?"
03 AON
04 PROMPT
05 AOFF
06 ASTO 00
07 "N?"
08 PROMPT
09 STO 01
10 "a↑b?"
11 PROMPT
12 ♦LBL "FIP"
13 STO 04
14 -
15 RCL 01
16 /
17 STO 05
18 2
19 /
20 ST+ 04
21 .6
22 SQRT
23 *
24 STO 03
25 CLX
26 STO 02
27 ♦LBL 00
28 RCL 04
29 RCL 03
30 +
31 XEQ IND 00
32 5
33 *
34 ST- 02
35 RCL 04
36 RCL 03
37 -
38 XEQ IND 00
39 5
40 *
41 ST- 02
42 RCL 04
43 XEQ IND 00
44 8
45 *
46 ST- 02
47 RCL 05
48 ST+ 04
49 DSE 01
50 GTO 00 ►
51 RCL 02
52 *
53 18
54 /
55 END
```

- *55 steps*
- *requires at least SIZE 006*
- *uses the Alpha register*
- *uses no flags and doesn't alter angular mode or display settings*
- *to get* ***** *press the* [x] *key*
- *to get* / *press the* [÷] *key*
- *the symbols ♦ and ► are purely cosmetic, to indicate branching*

3. Usage Instructions

The program can be used both interactively and programmatically, as follows:

1) ***Interactively:*** in **RUN** mode, call **"FI"** *(**F**unction **I**ntegrator)*. The program will prompt for the *name* of the program which defines *f(x)*, the required number of subintervals (***n***) and the integration limits (***a***, ***b***). Once provided, the program will proceed to compute the integral and return its value to the display.

2) ***Programmatically:*** your program must call **"FIP"** *(**F**unction **I**ntegrator **P**rogrammable)*, which assumes that the name of the program which defines *f(x)* is stored in register R_{00} , the number of subintervals (***n***, which should be an integer ≥ 1) is stored in register R_{01} , the lower limit (***a***) is in stack register *Y* and the upper limit (***b***) is in stack register *X*.

   Upon returning control to your program, the computed value of the integral will be in stack register *X*. The name of your *f(x)* will remain unaltered in register R_{00} but the number of subintervals will be lost. No subroutine levels are used apart from calls to your *f(x)*.

In both cases you need to write a program to define *f(x)*, the equation to solve. It must be an independent program under its own *global label*, must assume that the argument *x* is in stack register *X* upon being called, and must return the computed value of *f(x)* in stack register *X* as well.

4. Examples

The following examples can be useful to check that the program is correctly entered and to understand its usage.

4.1 Example 1

Evaluate the integral $I = \int_{3.59}^{20.19} x^5 \,.dx$

In **PRGM** Mode, enter the following *4-step* program to define the function *f(x)* to be integrated:

```
01 ♦LBL "FX1"     02 5      03 Y↑X      04 END
```

In **RUN** Mode, proceed as follows to evaluate the integral using just one (sub)interval:

`XEQ "FI"` ***NAME?*** `"FX1" [R/S]` ***N?*** `1 [R/S]` ***a↑b?*** `3.59 [ENTER] 20.19 [R/S]` <u>***11288934.08***</u>

The theoretical value is $(20.19^6 - 3.59^6)/6$ = ***11288934.08***... so the computed value has all 10 digits correct. The reason for such accuracy despite the large integration interval and using just one (sub)interval is because the *3-point Gauss-Legendre* quadrature formula is *exact* for polynomial functions up to the 5th degree, which x^5 is.

4.2 Example 2

Evaluate the integral $I = \int_{0}^{1} \frac{\text{Sin}(x)}{x} \,.dx$

In **PRGM** Mode, enter the following *5-step* program to define the function *f(x)* to be integrated:

```
01 ♦LBL "FX2"     02 SIN     03 LASTX     04 /     05 END
```

In **RUN** Mode (*and* **RAD**, **FIX 9**), proceed as follows to evaluate the integral using *1*, *2* and *4* subinterval(s):

```
XEQ "FI"   NAME?  "FX2"  [R/S]  N?  1  [R/S]  a↑b?  0 [ENTER] 1 [R/S]   0.946083134  (339)
XEQ "FI"   NAME?  "FX2"  [R/S]  N?  2  [R/S]  a↑b?  0 [ENTER] 1 [R/S]   0.946083072  (717)
XEQ "FI"   NAME?  "FX2"  [R/S]  N?  4  [R/S]  a↑b?  0 [ENTER] 1 [R/S]   0.946083071  (706)
```

The exact value rounds to ***0.9460830704*** so the computed values have *6*, *8* and ~*10* correct digits, respectively.

4.3 Example 3

Write a program to compute the volume of the solid of revolution generated by *any* user-specified curve and in particular use it to find the volume of the solid of revolution obtained by the turn of the catenary

$$y = \frac{3e^{x/3} + 3e^{-x/3}}{2}$$

around the x axis between $x = 0$ and $x = 1.2$ (your program will call **"FIP"** ***(P**rogrammable)* to do the hard work.)

In **PRGM** Mode, enter the following *20-step* generic program to compute said volume for any curve *y=f(x):*

```
01 ♦LBL "VOL"     05 AOFF       09  2          13 XEQ "FIP"    17 ♦LBL "VAUX"
02 "NAME?"        06 ASTO 06    10 STO 01      14  PI          18 XEQ IND 06
03 AON            07 "VAUX"     11 "X1↑X2?"    15  *           19 X↑2
04 PROMPT         08 ASTO 00    12 PROMPT      16 RTN          20 END
```

Now, still in **PRGM** Mode, enter the following *10-step* program to define the catenary's formula:

```
01 ♦LBL "CATEN"   03  /         05 ENTER↑      07  +           09  *
02  3             04 E↑X        06 1/X         08 1.5          10 END
```

In **RUN** Mode, proceed as follows to evaluate the volume asked for:

```
XEQ "VOL"  NAME?  "CATEN"  [R/S]  X1↑X2?   0 [ENTER] 1.2 [R/S]   35.79755410   (correct to ~9 places)
```

Notes

1. Despite its very simple coefficients, this *3-point Gauss-Legendre* quadrature formula gives 5th-order accuracy for arbitrary *f(x)* using just 3 evaluations per subinterval. This is much better than *Simpson's Rule*, which only gives 3rd-order accuracy.

2. Integrals which have *singularities* at one or both endpoints can also be computed, as *f(x)* is not evaluated there.

3. If you can use *synthetic instructions*, you may replace registers R_{03}, R_{04} and R_{05} by registers *M*, *N*, and *O* respectively (i.e.: `STO 03` becomes `STO M` and so on), and insert step `55 CLA` just before `END` to clear the Alpha register before the program ends. After this, program length will be 56 steps and min. *SIZE 003*, thus saving 3 numbered registers for other uses.

4. This program (**FI**, **FIP**) was duly submitted for inclusion in the **PPC ROM** but it wasn't accepted.

References

Francis Scheid (1988). *Schaum's Outline of Theory and Problems of Numerical Analysis, 2nd Edition*.

Copyrights