[6.19] Sub-divide integration range to improve accuracy

You can improve the accuracy of numerical integration by taking advantage of the fact that the 89/92+ numerical integrator is more accurate over small intervals. For example, consider this function:

$$Q(x) = \frac{1}{\sqrt{2\pi}} \int_x^\infty e^{-\frac{t^2}{2}} dt$$

This is actually the integral for the the complement of the cumulative normal probability distribution function. If you just integrate this function as shown, for x = 1,

```
1/(√(2·π))·∫(e^(-t^2/2),t,1,∞) = Ø.1586 5525 3456 96
```

which has an error of about 4.745E-10. To get a more accurate result, find integrals over smaller ranges that cover the entire integration limits range, and sum these to get the complete integral. The table below shows the individual integrals for some ranges I chose.

Integration limits	Integral
t = 1 to t = 2	0.1359 0512 1983 28
t = 2 to t = 3	0.0214 0023 3916 549
t = 3 to t = 4	0.0013 1822 6789 7969
t = 4 to t = 5	3.1384 5902 6124 E-5
t = 5 to t = 6	3.8566 4984 2341 4 E-7
t = 6 to t = 7	9.8530 7832 4938 2 E-10
t = 7 to t = 8	1.2791 9044 7828 2 E-12
t = 8 to t = 9	6.2198 3198 5865 6 E-16

Summing these individual integrals gives a result of 0.1586 5525 3931 46, which has an error of zero, to 14 digits.

Note that I stopped taking integrals when the individual integral is less than 1E-14, since further, smaller integrals will not contribute to the sum.

This method can be used with the ∫ operator, or with the *nInt()* function.

The function *nintx(),* shown below, automates this process.

```
nintx(ffx,xv,xx1,xx2,nx)
func
©Numerical integrator with summed subintervals
©dburkett@infinet.com 6 nov 99
©ffx: function to integrate
©xv: variable of integration
©xx1,xx2: lower and upper integration limits
©nx: number of subintervals

local dx

(xx2-xx1)/nx→dx

sum(seq(nint(ffx,xv,xx1+i*dx,xx1+(i+1)*dx),i,Ø,nx-1))

endfunc
```

For example, the call nintx(tan(x),x,0,1.5707,5) integrates tan(x) from x=0 to x=1.5707, with 5 subintervals.

The amount of improvement depends on the function. The built-in *nint()* function has an error of -29.7E-12 for the tan(x) example above. *nintx()* with 10 intervals has an error of about 13E-12.