Computer Programming on Geosciences

Numerical Integration Methods

Gauss CHANG[†]

[†] Department of Physics
NATIONAL TAIWAN UNIVERSITY

September 18, 2024

Outline

Inspiration

Inspiration: An Interesting Question

I saw this interesting question a few month ago:

Question

How do you prove that

$$\pi^3 > 31 \tag{1}$$

Inspiration: An Interesting Question

And there was a brilliant answer:

Proof

Notice that

$$\int_0^1 \left[\frac{8x^5 (1-x)^2 \left(324889 - 120736x^2\right)}{445625 (1+x^2)} \cdot \left(\ln^2 x\right) \right] \mathrm{d}x = \pi^3 - 31 \quad (2)$$

since the integrand is positive over the region, the integral is also positive.

Trivially, $\pi^3 > 31$ is proven.

Addressing The Problem

Addressing The Problem: Construct Function

Here is an example code for the integrand.

```
function integrand(x) result(res)
   implicit none
   real :: x
   real :: numerator, denominator, res
   real, parameter :: C1 = 324889.0
   real, parameter :: C2 = 120736.0
   real, parameter :: C3 = 445625.0
   numerator = 8.0 * x**5 * (1.0 - x)**2 * (C1 - C2 *
       x**2) * (log(x))**2
   denominator = C3 * (1.0 + x**2)
   res = numerator / denominator
end function integrand
```

Addressing The Problem: Plot The Integrand

Try to plot the integrand, just make sure it is a “well-behaved” function.

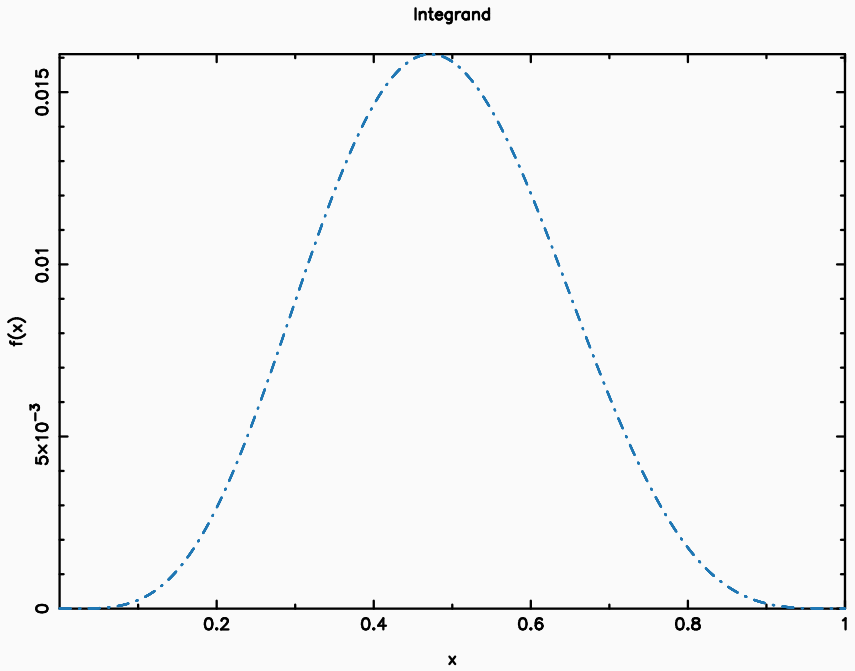

Figure 1: $\frac{8x^5(1-x)^2\left(324889-120736x^2\right)}{445625\left(1+x^2\right)} \cdot \left(\ln^2 x\right)$

Approach One

Approach One: Monte Carlo Method

Here is an example code for the integrand.

```
def integrand(x: float) -> float:
    C1: Final = 324889
    C2: Final = 120736
    C3: Final = 445625

    numerator = 8 * x**5 * (1 - x)**2 * (C1 - C2 * x**2)
        * (np.log(x))**2
    denominator = C3 * (1 + x**2)

    return numerator / denominator
```

Approach One: Monte Carlo Method

Here is another example code for the integrand.

```
double integrand(double x) {
    const double C1 = 324889;
    const double C2 = 120736;
    const double C3 = 445625;

    double numerator = 8 * pow(x, 5) * pow(1 - x, 2) *
        (C1 - C2 * pow(x, 2)) * pow(log(x), 2);
    double denominator = C3 * (1 + pow(x, 2));

    return numerator / denominator;
}
```

Fortran Basic: Fortran is formatted

Fortran 77

- Fixed Format
- 1-72 char

```
Program Test
declare
...
Stop
End
```

Fortran 90

- Free Format
- 1-132 char

```
Program test
declare
...
End program test
```

Fortran Basic: Fixed Format

"Fixed Format" regulates different roles of character positions. Columns 7-72 are used for coding. The first 5 columns can only be blank or contain a numeric label. Column 6 can only be a nonblank, nonzero character to be identified as a continuation line.