Numerikus integrálás C++-ban

Krucsay András

2025. 03. 01.

Alkalmazott függvények

A következő integrálokat használtam a következő kimenetekkel:

$$\int_{-1}^{3} \cos(x) \cdot \exp(-x^2) \mathrm{d}x$$

n = 10, Az integrál eredménye: 1.3369576385529844
n = 100, Az integrál eredménye: 1.3462937716508321
n = 1000, Az integrál eredménye: 1.3463870149680062
n = 10000, Az integrál eredménye: 1.3463879473851070
n = 100000, Az integrál eredménye: 1.3463879567092620
WolframAlpha: 1.3463879568034504

$$\int_{-1}^{0} \exp\left(\exp\left(\exp\left(x\right)\right)\right) \mathrm{d}x$$

n = 10, az integrál eredménye: 7.2764682979272095
n = 100, az integrál eredménye: 7.2445031929960821
n = 1000, az integrál eredménye: 7.2441819549714355
n = 10000, az integrál eredménye: 7.2441787424301340
n = 100000, az integrál eredménye: 7.2441787103046975
WolframAlpha: 7.2441787099802110

$$\int_{0}^{\pi} \sin\left(\cos\left(\exp\left(-\tanh\left(x^x\right)\right)\right)\right)$$

n = 10, az integrál eredménye: 2.4774198963372136
n = 100, az integrál eredménye: 2.4758435679798834
n = 1000, az integrál eredménye: 2.4758187873419883
n = 10000, az integrál eredménye: 2.4758184386504758
n = 100000, az integrál eredménye: 2.4758184341330622
WolframAlpha: 2.4758184334819195

Maga a kód

```
#include <iostream>
#include <iomanip>
#include <cmath>

using namespace std;

double f(double x) {
    return cos(x) * exp(-x * x);
}

double integrate(int n, double x0, double x1) {
    double h = (x1 - x0) / n;
    double sum = 0.5 * (f(x0) + f(x1));

    for (int i = 1; i < n; ++i) {
        double x = x0 + i * h;
        sum += f(x);
    }

    return sum * h;
}

int main() {
    double x0 = -1.0;
    double x1 = 3.0;

    cout << fixed << setprecision(16);

    int n_values[] = {10, 100, 1000, 10000, 100000};

    for (int n : n_values) {
        double result = integrate(n, x0, x1);
        cout << "n = " << n << ", az integrál eredménye: " << result << endl;
    }

    return 0;
}
```

A kód magyarázata

`f(double x)`

Ez a függvény egy konkrét integrálandó függvényt definiál.

`integrate(int n, double x0, double x1)`

Ez a függvény végzi el az integrálás numerikus kiszámítását a trapézszabály segítségével.

A h változó a lépésközt határozza meg: $h = \frac{x_1 - x_0}{n}$. Ugyebár minél kisebb a h, annál pontosabb lesz az eredmény, ezért $\lim_{n \to \infty}$ esetén a pontos eredmény egyre jobban történő közelítését kapjuk vissza, avagy a függvény konvergál a valódi értékhez.

A sum változó tárolja az integrál közelítő értékét, kezdetben a végpontok felével. A for ciklus hozzáadja a belső pontok függvényértékeit, majd az összeg szorzása a lépésközzel adja a közelítést.

`main()`

A main függvény beállítja az integrálási tartományt és a kiírás pontosságát, majd egy ciklusban kiszámítja és kiírja az eredményeket különböző n értékekre.

Egyéb megjegyzések

Lustaság gyanánt `using namespace std;`-t használtam.

Először a program nagyon lassan futott le, ezért optimalizálás gyanánt a h lépésközt előre kiszámíttattam és az x értékét a ciklusban minden iterációban kiszámítva minimalizálni próbáltam a függvénymeghívások számát.