Metody numeryczne - N14

DAMIAN PORADYŁO

N14 *Zadanie numeryczne*

Obliczyć całkę

$$\int_{-1}^{1} \frac{\exp(x^2)\,dx}{\sqrt{1-x^2}}$$

z dokładnością do 10^{-6} za pomocą złożonej metody trapezów, Simpsona i reguły 3/8, stosując iteracyjne zagęszczanie podprzedziałów.

Uwaga: za liczenie dwukrotnie (lub więcej) wartości funkcji w obrębie jednej metody przy zagęszczaniu będą odejmowane punkty.

Wskazówka: zmienić zmienne tak aby pozbyć się nieskończoności na brzegach przedziału.

1 OPIS METODY

Całkowanie numeryczne jest zbiorem metod pozwalających oszacować wartość zadanej całki oznaczonej. Mając daną funkcję $f(x)$, szukaną wartością jest:

$$\int_a^b f(x)\,dx$$

gdzie a i b to przedziały całkowania.

W naszym przypadku na pierwszy rzut oka możemy zobaczyć, że z tą całką jest coś nie tak:

$$\int_{-1}^{1} \frac{\exp(x^2)\,dx}{\sqrt{1-x^2}}$$

Problem mianowicie jest taki, że na końcach jej przedziałów funkcja zmierza do nieskończoności.

Całkę należy przekształcić do poniższej postaci:

$$\int_0^{\pi} \exp\left(cos^2(t)\right) dt$$

gdzie, $x = \cos(t)$

Funkcja ta przyjmuje wartości -1 i 1 w miejscach $[0;\,\pi]$. I w takich tez granicach obliczamy nową całkę.

We wszystkich metodach została użyta zmienna $\mathrm{h} = \frac{\mathrm{b-a}}{\mathrm{N}}$ oznaczająca długość najmniejszego podprzedziału.

1.1 METODA TRAPEZÓW

Metoda trapezów interpoluje funkcję podcałkową za pomocą wielomianu pierwszego stopnia wykorzystując punkty będące granicą całkowania:

$$\int_a^b f(x)dx \sim (b-a)\frac{f(a)+f(b)}{2}$$

Nazwa metody pochodzi od obliczanego pola, którym jest trapez. Przedział całkowania musimy podzielić na n części i dla każdej z tych części obliczyć całkę ze wzoru powyżej, zaprowadzi nas to do poniższej zależności:

$$\int_a^b f(x)dx \sim \frac{b-a}{n}\sum_{i=0}^{n-1}\left(f\left(a+i\frac{b-a}{n}\right)+f\left(a+(i+1)\frac{b-a}{n}\right)\right)$$

Algorytm ten wymaga wyznaczania przy każdej iteracji dwa razy wartości funkcji podcałkowej. Możemy go uprościć, biorąc $x_i = a + i\frac{b-a}{n}$. Otrzymujemy wówczas:

$$\int_a^b f(x)dx \sim \frac{b-a}{n}\left(\frac{f(a)}{2}+\frac{f(b)}{2}+\sum_{i=1}^{n-1} f\left(a+i\frac{b-a}{n}\right)\right)$$

1.2 METODA SIMPSONA

W metodzie Simpsona funkcję podcałkową f(x) przybliża się wielomianem interpolacyjnym Lagrange'a, gdzie węzłami są punkty reprezentujące początek, środek i koniec przedziału całkowania.

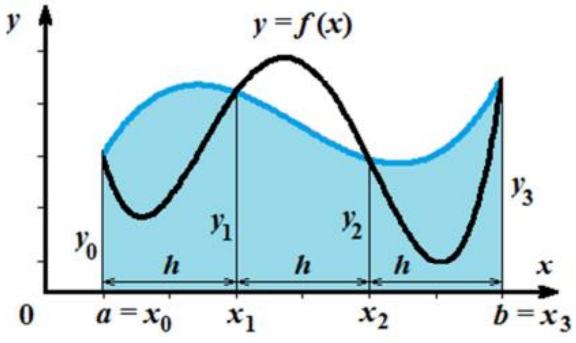
Przyjmując, że $x_0 = a, x_2 = b$, oraz $h = \frac{b-a}{2}$ otrzymuje się zależność szacującą całkę funkcji f(x) w przedziale [a; b]:

$$\int_a^b f(x)dx \sim \frac{b-a}{6}\left(f(a) + f(b) + 4f\left(\frac{a+b}{2}\right)\right)$$

Aby zwiększyć dokładność szacowanej całki, należy przedział [a;b] podzielić na n części. Dla $h = \frac{b-a}{n}$ otrzymujemy poniższy wzór, z którego korzystamy podczas numerycznego obliczania:

$$\int_a^b f(x)dx \sim \frac{h}{6}\left(f(a) + 4\sum_{i=1}^{n} f\left(a + ih + \frac{h}{2}\right) + 2\sum_{i=1}^{n-1} f(a + ih) + f(b)\right)$$

1.3 REGUŁA 3/8



W tej metodzie parabole są używane do przybliżenia każdej części krzywej. Zasada 3/8 jest również nazywana drugą zasadą Simpsona. Jest ona następująca:

$$\int_a^b f(x)\,dx \approx \frac{3h}{8}\left[f(a) + 3f\left(\frac{2a+b}{3}\right) + 3f\left(\frac{a+2b}{3}\right) + f(b)\right]$$

Gdzie $b - a = 3h$

Dzieląc przedział [a,b] na n podprzedziałów o długości $h = \frac{b-a}{n}$ i wprowadzając $x_i = a + ih$ otrzymujemy poniższy wzór:

$$\int_a^b \mathrm{f(x)\,dx} = \frac{3h}{8}\left[f(a) + 3\sum_{i \neq 3k}^{n-1} f(a+ih) + 2\sum_{j=1}^{\frac{n}{3}-1} f(a+ih) + f(b)\right]$$

2 WYNIKI

Wyniki uzyskane z dokładnością 10^{-6}

Nazwa metody	Liczba iteracji	Wynik
Metoda trapezów	8	5.5084297778
Metoda Simpsona	8	5.5084297726
Reguła 3/8	21	5.5084297581

3 KOD PROGRAMU

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

#define E 1e-6

double f(double x) {
    return exp(cos(x) * cos(x));
}
double method_Trapez(double a, double b, double (*f)(double)) {
    double sum = f(a) + f(b);
    double prevIntegral, integral = 0.0;
    int intervals = 1;
    do  {
        intervals++;
        prevIntegral = integral;
        double sum2 = 0.0;
        double h = (b-a) / intervals;

        for(int i=1; i<intervals; i++) sum2 += f(a+i*h);

        integral = h*((0.5 * sum) + sum2);
    } while (fabs(integral - prevIntegral) > E);
    std::cout << "Metoda trapezow - liczba iteracji: " << intervals << " - \t";
    return integral;
}
double method_Simpson(double a, double b, double (*f)(double)) {
    double sum = f(a) + f(b);
    double prevIntegral, integral = 0.0;
    int intervals = 0;
    do {
        intervals++;
        prevIntegral = integral;
        double sum2 = 0.0;
        double sum4 = 0.0;
        double h = (b-a)/intervals;
        double h2 = h/2.0;
        for(int i=1; i<=intervals; i++) sum4 += f(a+i*h-h2);
        for(int i=1; i<intervals; i++) sum2 += f(a+i*h);
        integral = h/6.0 * (sum + 4.0 * sum4 + 2.0 * sum2 );
    } while (fabs(integral - prevIntegral) > E);
    std::cout << "Metoda Simpsona - liczba iteracji: " << intervals << " - \t";
    return integral;
}
double method_threeAndEight(double a, double b, double (*f)(double)) {
    double sum = f(a) + f(b);
    double sum_copy = sum;
    double prevIntegral, integral = 0.0;
    int intervals = 0;
    do {
        intervals+=3;
        sum = sum_copy;
        prevIntegral = integral;
        double h = (b-a) / intervals;
        for (int i=1; i<intervals; i++) {
            if(i%3 == 0) sum += 2*f(a+i*h);
            else sum += 3*f(a+i*h);
        }
        integral = (3.0*h / 8.0) * sum;

    } while (fabs(integral - prevIntegral) > E);
    std::cout << "Metoda 3/8 - liczba iteracji: " << intervals << " - \t";
    return integral;
}

int main() {
    std::cout << std::fixed << std::setprecision(10) << method_Trapez(0, M_PI, f) << std::endl;
    std::cout << std::fixed << std::setprecision(10) << method_Simpson(0, M_PI, f) << std::endl;
    std::cout << std::fixed << std::setprecision(10) << method_threeAndEight(0, M_PI, f) << std::endl;
}
```

4 URUCHOMIENIE

$>>$ **make run**

W skład zestawu wchodzą:

- Plik programu
- Plik zawierający otrzymane wyniki
- Opracowanie
- Makefile