Національний технічний університет України

«Київський політехнічний інститут імені Ігоря Сікорського»

Кафедра теоретичних основ радіотехніки

**ЗВІТ З ЛАБОРАТОРНОЇ РОБОТИ №2**

**ОБЧИСЛЕННЯ ВИЗНАЧЕНОГО ІНТЕГРАЛУ**

з дисципліни: «Інформатика 1»

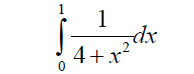
|  |  |
| --- | --- |
|  | Виконав : Берднік Станіслав  Група: РЕ-12  Викладачі: доцент Катін П.Ю.  Оцінка: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Підпис: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |

Київ – 2021

**Мета роботи**:

Вивчити методи чисельного інтегрування; скласти програму обчислення визначеного інтегралу чисельними методами; дослідити залежність точності розрахунку інтегралу від кількості проміжків розбиття інтервалу інтегрування.

**Варіант**:1

**Обраний інтеграл**: 

**Чисельне значення обрахованого интегралу = 0.231824**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Метод | n=10 | n=100 | n=1000 | n=10000 | N |
| Лів. прямокут | 0.234257 | 0.232073 | 0.231849 | 0.231826 | 70 |
| Прав.прямокут | 0.234257 | 0.232073 | 0.231849 | 0.231826 | 70 |
| Трапеції | 0.258467 | 0.234565 | 0.232099 | 0.231851 | 234 |
| Сімсона | 0.261824 | 0.234824 | 0.232124 | 0.231854 | 244 |

**Таблиця:**

Код:

#include <stdio.h>

#include <stdlib.h>

#include <conio.h>

#include <math.h>

#include <locale.h>

unsigned int metod(void);

double left\_p ( double a, double b, unsigned int n );

double right\_p( double a, double b, unsigned int n );

double trap ( double a, double b, unsigned int n );

double simpson( double a, double b, unsigned int n );

double f(double x);

void resultat(double a, double b, unsigned int n, double I);

int main()

{

setlocale(LC\_ALL, "Rus");

unsigned int num\_of\_method;

unsigned int n=100;

const double a = 0;

const double b = 1;

double I, I1, I2;

double maxpoh = 0.00001;

unsigned int N=0;

num\_of\_method = metod();

printf("\n\nВедiть кiлькiсть промiжкiв:");

scanf("%u",&n);

printf("\n\nНатиснiть будь яку клавiшу щоб продовжити\n");

getch();

//----------------------------------------------------------//

if (num\_of\_method == 1 )

{

I = left\_p(a, b, n);

N = 0;

do

{

N = N + 2;

I1 = left\_p(a, b, N);

I2 = left\_p(a, b, N+2);

}

while(fabs(I2-I1) > maxpoh);

resultat(a, b, n, I );

printf("\n\nкiлькiсть розбитiв для допустимоi похибки:\n\nN=%u, Integral(N)=%.8lf\n\n\n\n", N, I1);

}//----------------------------------------------------------//

else if( num\_of\_method == 2 )

{

I=right\_p(a, b, n);

do

{

N = N + 2;

I1 = right\_p(a, b, N);

I2 = right\_p(a, b, N+2);

}

while( fabs(I2-I1) > maxpoh );

resultat(a, b, n, I);

printf("\n\nкiлькiсть розбитiв для допустимоi похибки:\n\nN=%u, Integral(N)=%.8lf\n\n\n\n", N, I1);

}//--------------------------------------------------------------//

else if( num\_of\_method == 3 )

{

I=trap(a, b, n);

do{

N = N + 2;

I1 = trap(a, b, N);

I2 = trap(a, b, N+2);

}

while( fabs(I2-I1) > maxpoh);

resultat(a, b, n, I);

printf("\n\nкiлькiсть розбитiв для допустимоi похибки:\n\nN=%u, Integral(N)=%.8lf\n\n\n\n", N, I1);

}//------------------------------------------------------------//

else if( num\_of\_method == 4 )

{

I=simpson(a, b, n);

do

{

N = N + 2;

I1 = simpson(a, b, N);

I2 = simpson(a, b, N+2);

}

while( fabs(I2-I1) > maxpoh);

resultat(a, b, n, I);

printf("\n\nкiлькiсть розбитiв для допустимоi похибки:\n\nN=%u, Integral(N)=%.6lf\n\n\n\n", N, I1);

}

return 0;

}

//---------------------------------------------------------------------

unsigned int metod(void)

{

unsigned int m;

printf("\n\n Оберiть метод:\n1 - лiвих прямокутникiв \n2 - правих прямокутникiв \n3 - трапецii \n4 - сiмпсона\n>");

scanf("%u", &m);

while( m < 1 || m > 4 )

{

printf("\n!!!Ведене не вiрне число!!!(1,2,3,4?) ");

scanf("%u", &m);

}

switch(m){

case 1:

printf("\nВи обрали метод --лiвих прямокутникiв--");

break;

case 2:

printf("\nВи обрали метод --правих прямокутникiв--");

break;

case 3:

printf("\nВи обрали метод --трапецiй--");

break;

case 4:

printf("\nВи обрали метод --Сiмпсона--");

break;

}

return m;

}

//-------------------------------------------------------------------

double f( double x )

{

double funk;

funk = (1/(4+pow(x,2)));

return funk;

}

//---------------------------------------------------------------------

double left\_p(double a, double b, unsigned int n)

{

double integral;

double h;

double x;

double sum = 0;

unsigned int k;

h = ( b - a ) / n;

x = a;

for (k = 0; k <= n-1; k++ )

{

sum = sum + f(x);

x = x + h;

}

integral = sum \* h;

return integral;

}//------------------------------------------------------

double right\_p(double a, double b, unsigned int n)

{

double integral;

double h;

unsigned int k;

double x;

double sum = 0;

h = ( b - a ) / n;

x = a;

for (k = 1; k <= n; k++ )

{

sum = sum + f(x);

x = x + h;

}

integral = sum \* h;

return integral;

}//--------------------------------------------------------

double trap(double a, double b, unsigned int n)

{

double integral;

double h;

unsigned int k;

double x;

double sum = 0;

sum = f(a)+f(b);

h = ( b - a )/n;

x = a;

for (k = 1; k <= n-1; k++ ){

sum = (sum + f(x));

x = x + h;

}

integral = sum \* h;

return integral;

}//-------------------------------------------------------

double simpson(double a, double b, unsigned int n)

{

double integral;

double h;

unsigned int k;

double x;

double sum1 = 0;

double sum2 = 0;

h=(b - a)/n;

x = a;

for (k = 0; k <= n; k++ ){

if(k%2 != 0)

{

sum1=sum1+f(a+h\*k);

}

if(k%2 == 0)

{

sum2=sum2+f(a+h\*k);

}

}

integral = ((b-a)/(n\*3))\*(f(a)+4\*sum1+2\*sum2+f(b));

return integral;

}

//-------------------------------------------------------------------

void resultat(double a, double b, unsigned int n, double I)

{

system("cls");

printf("\n\na=%.2lf b=%.2lf n=%u Integral = %.6lf \n ", a, b, n, I);

}

