Національний технічний університет України

«Київський політехнічний інститут імені Ігоря Сікорського»

Кафедра теоретичних основ радіотехніки

**ЗВІТ З ЛАБОРАТОРНОЇ РОБОТИ №2**

з дисципліни: «Інформатика »

|  |  |
| --- | --- |
|  | Виконав: Бондарчук Назар Олегович  Група: РЕ-11  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.

**Інтеграл**: 

**Ключові моменти**:

1)На початку програми я підключаю бібліотеку

#include <stdio.h>

#include <stdlib.h>

#include <conio.h>

#include <math.h>

2)Задаю функцію

y = 1/(4 + x\*x);

3)Для обрахунку кожного з методів я зробив окремі функції:

double simpsonn () – Метод Сімпсона

double left\_triangle () – Метод Лівого прямокутника

double right\_triangle () – Метод Правого прямокутника

double trapez () – Метод трапецій

4)Для виводу інформації на консоль викликаємо функції дані функції:

print\_result(a ,b ,n ,I2);

printf("\n\nN=%u, I1(N)=%.8lf\n", N, I1);

**Код:**

#include <stdio.h>

#include <stdlib.h>

#include <conio.h>

#include <math.h>

//#define eps 0.0001

unsigned int method\_select( void );

double f( double x );

void print\_result(double a, double b, unsigned int n, double I );

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

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

double trapez( );

double simpsonn();

int main()

{

unsigned int num\_of\_method; // 1 - left, 2 - right, 3 - trap, 4 - Simpson

const double a = 0;

const double b = 1;

unsigned int n = 200;

double I;

double I1, I2;

unsigned int N;

double eps = 0.00001;

num\_of\_method = method\_select();

if (num\_of\_method == 1 ){

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

print\_result(a, b, n, I);

eps=I2-I1;

N = 0;

do{

N = N + 2;

I1 = left\_rectangle(a, b, n);

I2 = left\_rectangle(a, b, n+2);

}while( fabs(I2-I1) > eps );

printf("\n\nN=%u, I1(N)=%.8lf\n", N, I1);

}

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

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

print\_result(a, b, n, I);

N = 0;

do{

N = N + 2;

I1 = right\_rectangle(a, b, n);

I2 = right\_rectangle(a, b, n+2);

}while( fabs(I2-I1) > eps );

printf("\n\nN=%u, I1(N)=%.8lf\n", N, I1);

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

do {

I1 = trapez(a ,b ,n);

N = N + 2;

I2 = trapez(a ,b ,n);

}

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

print\_result(a ,b ,n ,I2);

printf("\n\nN=%u, I1(N)=%.8lf\n", N, I1);

}

else{

do {

double simpsonn();

I1 = simpsonn(a ,b ,n);

N = N + 2;

I2 = simpsonn(a ,b ,n);

}

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

print\_result(a ,b ,n ,I2);

printf("\n\nN=%u, I1(N)=%.8lf\n", N, I1);

}

return 0;

}

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

unsigned int method\_select(void)

{

unsigned int temp;

printf("\n\n Choose your method:\n1 - left \n2 - right \n3 - trap \n4 - Simpson\n>");

scanf("%u", &temp);

while( temp < 1 || temp > 4 ){

printf("\n!!!!Invalid data. Method: 1, 2, 3 or 4: ");

scanf("%u", &temp);

}

switch(temp){

case 1:

printf("\nYou chose method of LEFT RECTANGLE");

break;

case 2:

printf("\nYou chose RIGHT RECTANGLE method");

break;

case 3:

printf("\nYou chose TRAPEZOID method");

break;

case 4:

printf("\nYou chose SIMPSON method");

break;

}

printf("\nPress any button to continue");

getch();

return temp;

}

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

double f( double x )

{

double y;

y = 1/(4 + x\*x);

return y;

}

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

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

{

double h;

unsigned int k;

double x;

double sum = 0;

h = ( b - a ) / n;

x = a;

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

sum = sum + f(x);

x = x + h;

}

return sum \* h;

}

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

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

{

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;

}

return sum \* h;

}

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

double trapez (double a, double b, unsigned int n) {

double h = 0, x = 0, sum = 0;

unsigned int k = 0;

h = (b - a)/n;

x = a;

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

sum = sum + (f(x) + f(x + h))/2;

x = x + h;

}

return h\*sum;

}

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

double simpsonn (double a, double b, unsigned int n) {

double h = 0, x = 0, sum = 0;

unsigned int k = 0;

double parna\_suma = 0 , neparna\_suma = 0;

h = (b - a)/n;

x = a + h;

for (k = 1; k <= n;k = k+2) {

neparna\_suma = neparna\_suma + f(x);

x = x + 2\*h;

}

x = a + 2\*h;

for(k = 2; k <= n-2;k = k+2) {

parna\_suma = parna\_suma + f(x);

x = x + 2\*h;

}

x = a;

sum = f(x) + 4\*neparna\_suma + 2\*parna\_suma + f(x + n\*h);

return sum\*h/3;

}

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

void print\_result(double a, double b, unsigned int n, double I)

{

system("cls");

printf("\n\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");

printf("\n\* Results \*");

printf("\n\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n");

printf("a=%.2lf b=%.2lf n=%u I = %.8lf", a, b, n, I);

}



