Національний технічний університет України

«Київський політехнічний інститут імені Ігоря Сікорського»

# Кафедра радіотехнічних систем

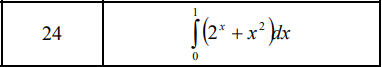
**ЗВІТ З ЛАБОРАТОРНОЇ РОБОТИ №2**

з дисципліни: «Інформатика1. Основи програмування та алгоритми»

|  |  |
| --- | --- |
|  | Виконав: Черкас Максим Віталійович  Група: РС-21  Викладач: Турєєва О.В. |

Київ – 2022

**Мета роботи**: вдосконалення знань, умінь та навичок з технології розроблення програмного забезпечення (ПЗ) з використанням мови С у процедурній парадигмі. Також лабораторна робота дає основні навички обчислення визначеного інтегралу чисельними методами.

**Обрана функція**: 

**Ключові моменти**:

1. На початку програми задаємо змінні типу unsigned int та double.
2. Далі вводимо дані, які потрібні будуть для обчислень.
3. Реалізовуємо вибір варіанту для обрахунків через method\_select().
4. Обраховуємо функцію за формулою pow(2,x)+pow(x,3).

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Метод | n=10 | n=100 | n=500 | t | N |
| Ліві | 1.678273 | 1.766051 | 1.774029 | 5.524 s | 142 |
| Праві | 2.213627 | 1.816391 | 1.784043 | 5.822 s | 284 |
| Трапеція | 1.991861 | 1.796186 | 1.780035 | 5.744 s | 200 |
| Парабола | 1.602112 | 1.756298 | 1.772039 | 5.625 s | 198 |

1. Виводимо результат через print\_result та дані які вказують на кількість проміжків, коли буде найбільш точний розрахунок.

**Код:**

#include <stdio.h>

#include <stdlib.h>

#include <conio.h>

#include <math.h>

#define eps 0.0001

unsigned int method\_select(void);

double fun(double x);//функція

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

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

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

double trapezoid(double a, double b, unsigned int n);//трапеції

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

int main()

{

unsigned int num\_of\_method;

double a, b, I, I1, I2;

unsigned int n, N=0;

printf("\n\n Введіть нижню границю:" );

scanf("%lf", &a);

printf("\n\n Введіть верхню границю:" );

scanf("%lf", &b);

printf("\n\n Введіть кількість проміжків:" );

scanf("%d", &n);

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

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

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

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

do{

N=N+2;

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

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

}

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

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

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

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

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

do{

N=N+2;

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

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

}

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

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

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

I=trapezoid(a, b, n);

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

do{

N=N+2;

I1=trapezoid(a, b, N);

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

}

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

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

}

else{

I=simpson(a,b,n);

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

do{

N=N+2;

I1 = simpson(a, b, N);

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

}

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

printf("\nN= %u I(N)= %lf", N, I1);

}

return 0;

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

unsigned int method\_select(void)

{

unsigned int temp;

printf("\n\n Введіть ваш метод:\n1 - ліві \n2 - праві \n3 - трапеція \n4 - сімпсона\n");

scanf("%u", &temp);

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

printf("\n Помилка. Введіть коректне значення:");

scanf("%u", &temp);

}

return temp;

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

double fun(double x){//обрахунок функції

double y;

y=pow(2,x)+pow(x,3);

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 + fun(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+h;

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

sum = sum + fun(x);

x = x + h;

}

return sum \* h;

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

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

double sum = 0, x = 0, h;

unsigned int k;

h=(b-a)/n;

x=a+h;

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

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

x+=h;

}

return sum \* h;

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

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

double sum1 = 0, sum2 = 0, h=0;

unsigned int k;

h=(b-a)/n;

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

if(k%2==0){

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

}else {

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

}

}

return (fun(a)+fun(b) + 4\*sum1 + 2\*sum2)\*h/3;

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

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

system("cls");

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

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

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

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

}

***Висновки:***

На цій лабораторній роботі ми розробили алгоритм обчислення інтегралу функції декількома способами, закріпивши навички використання вказівників.

