Національний технічний університет України

«Київський політехнічний інститут імені Ігоря Сікорського»

Кафедра теоретичних основ радіотехніки

**ЗВІТ З ЛАБОРАТОРНОЇ РОБОТИ №2 Частина№2**

з дисципліни: «Інформатика1. Основи програмування та алгоритми»

|  |  |
| --- | --- |
|  | Виконав: Скепський Данило Віталійович  Група: РЕ-22  Викладач: доцент Катін П.Ю. |

Київ – 2022

**Мета роботи**: Навчитися використовувати умовний оператор switch, створювати власні функції та обчислювати інтеграл різними методами.

**Ключові моменти:**

1. Додаємо бібліотеки
2. Створюємо прототипи функцій обчислення інтегралу
3. Робимо запит та зчитуємо значення лімітів(a,b) та кількості кроків(N)
4. Робимо запит та зчитуємо значення варіанту(variant), що відповідає одному з чотирьох методів.
5. Враховуючи метод, що був вибран вище використовуючи оператор switch(variant) виводимо відповідь визиваючи відповідну функцію обчислення
6. Створюємо функції обчислення інтегралу.

**Код:**

#include <stdlib.h>

#include <stdio.h>

#include <math.h>

#include <conio.h>

#define eps 0.00001 //похибка

double left\_rectangles(double a, double b, unsigned int N); //прототип функції методос лівих прямокутників

double right\_rectangles(double a, double b, unsigned int N); //прототип функції методом правих прямокутників

double trapezium (double a, double b, unsigned int N); //прототип функції методом трапеції

double simpson (double a, double b, unsigned int N); //прототип функції методом Сімпсона

double integral(double); //прототип функції знаходження інтегралу

#include <stdlib.h>

#include <stdio.h>

#include <math.h>

#include <conio.h>

#define eps 0.00001 //похибка

double left\_rectangles(double a, double b, unsigned int N); //прототип функції методос лівих прямокутників

double right\_rectangles(double a, double b, unsigned int N); //прототип функції методом правих прямокутників

double trapezium (double a, double b, unsigned int N); //прототип функції методом трапеції

double simpson (double a, double b, unsigned int N); //прототип функції методом Сімпсона

double integral(double); //прототип функції знаходження інтегралу

int main() // основна программа

{

double a, b;

int N, variant;

double integral\_sqare;

do // Запит на введення лімітів інтегрування

{

printf("\nEnter left limit:");

scanf("%lf", &a);

printf("\nEnter right limit:");

scanf("%lf", &b);

}

while(a==b);

do // Запит на введення кількості кроків

{

printf("\nEnter N(>0):");

scanf("%u", &N);

}

while(N<=0);

system("cls");

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

do { // Вибір методу

printf("\nChoose method:\n");

printf("\n1 = Left Rectangles method:");

printf("\n2 = Right Rectangles method:");

printf("\n3 = Trapezium method:");

printf("\n4 = Simpson method:");

printf("\nVariant:");

scanf("%u", &variant);

}

while (variant!=1 &&variant!=2 &&variant!=3 &&variant!=4 );

system("cls");

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

switch(variant) // Вивід відповіді залежно від вибранного варіанту

{

case 1:

{

integral\_sqare = left\_rectangles(a, b, N);

printf("\n\_\_\_\_\_\_\_\_\_Left Rectangles method\_\_\_\_\_\_\_\_\_");

printf("\na = %.2lf \nb = %.2lf \nIntegral = %.5lf \nN = %d",a, b, integral\_sqare, N);

}

break;

case 2:

{

integral\_sqare = right\_rectangles(a, b, N);

printf("\n\_\_\_\_\_\_\_\_\_Right Rectangles method\_\_\_\_\_\_\_\_\_");

printf("\na = %.2lf \nb = %.2lf \nIntegral = %.5lf \nN = %d",a, b, integral\_sqare, N);

}

break;

case 3:

{

integral\_sqare = trapezium(a, b, N);

printf("\n\_\_\_\_\_\_\_\_\_Trapezium method\_\_\_\_\_\_\_\_\_");

printf("\na = %.2lf \nb = %.2lf \nIntegral = %.5lf \nN = %d",a, b, integral\_sqare, N);

}

break;

case 4:

{

integral\_sqare = simpson(a, b, N);

printf("\n\_\_\_\_\_\_\_\_\_Simpson method\_\_\_\_\_\_\_\_\_");

printf("\na = %.2lf \nb = %.2lf \nIntegral = %.5lf \nN = %d",a, b, integral\_sqare, N);

}

break;

}

return 0;

}

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

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

double integral( double x ) // функція обчислення інтегралу

{

return (0.25\*pow(x,5)+4\*pow((x+20),4)-2\*x);

}

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

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

double left\_rectangles(double a, double b, unsigned int N) //функція методу лівих прямокутників

{

double integral\_sqare, x, h;

unsigned int i;

h = (b - a) / N;

x = a;

for (i = 0; i < N; i++)

{ x += h; integral\_sqare += integral(x);}

return integral\_sqare\*h;

}

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

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

double right\_rectangles(double a, double b, unsigned int N) //функція методу правих прямокутників

{

double integral\_sqare, x, h;

unsigned int i;

h = (b - a) / N;

x = a + h;

for (i = 0; i <= N; i++)

{ x += h; integral\_sqare += integral(x);}

return integral\_sqare \* h;

}

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

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

double trapezium (double a, double b, unsigned int N) //функція методу трапецій

{

double integral\_sqare, x , h;

unsigned int i;

h = (b - a) / N;

x = a + h;

for (i = 0; i <= (N - 1); i++)

{x += h; integral\_sqare += (integral(x) + integral(x + h)) / 2;}

return integral\_sqare \* h;

}

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

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

double simpson (double a, double b, unsigned int N) // фунція методу сімпсона

{

double sum1 , sum2 , h ;

h = (b - a) / N;

for (int i = 1; i <= (N - 1); i++)

{

if (i%2!=0)

sum1 += integral(a + h \* i);

else

sum2 += integral(a+ h \* i);

}

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

} }**Висновки:**

Ми навчилися використовувати використовувати умовний оператор switch(), створювати власні функції та викликати їх, а також ми навчилися знаходити інтеграл 4ма різними методами.





