Національний технічний університет України

«Київський політехнічний інститут імені Ігоря Сікорського»

**ЗВІТ З ЛАБОРАТОРНОЇ РОБОТИ №2**

**«ОБЧИСЛЕННЯ ВИЗНАЧЕНОГО ІНТЕГРАЛУ»**

з дисципліни: «Інформатика1. Основи програмування та алгоритми»

Виконав: Чуйко А.Є.

Група: РЕ–12

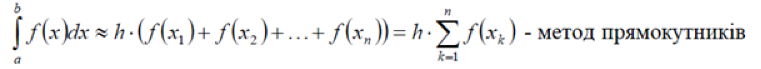
Викладач: Катін П. Ю.

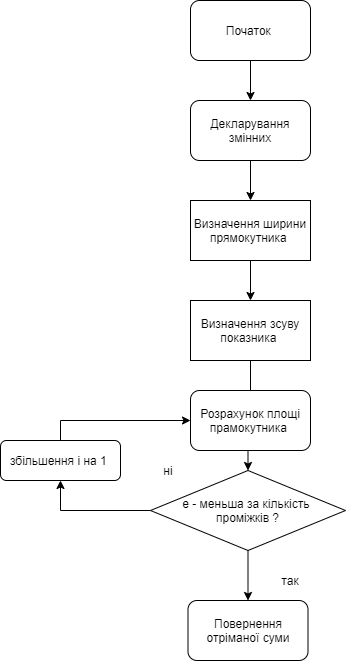
Київ – 2021

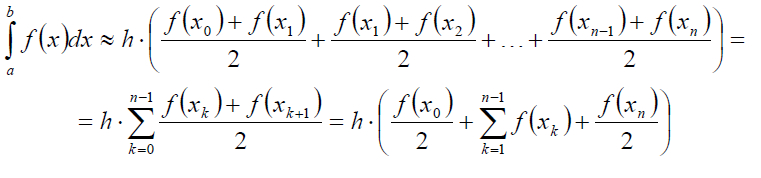
Мета роботи: навчитися використовувати цикли для вирішення інтегралів , виводити значення інтегралу на екран і змінювати значення інтеграла на потрібну похибку.

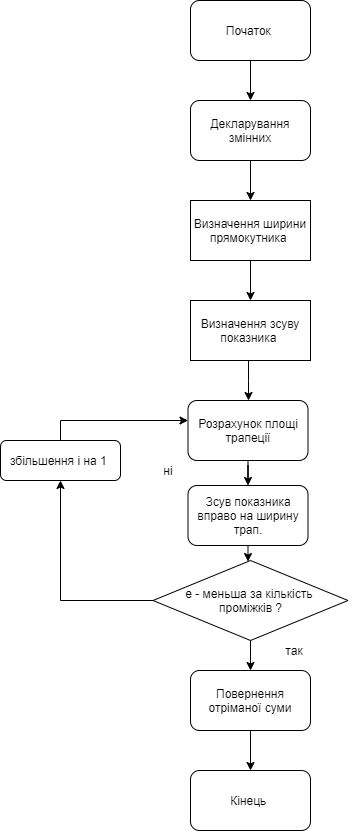
Блок-схеми кожного методу:

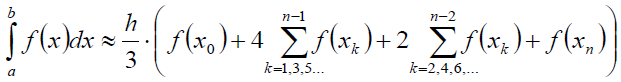
1. Метод прямокутника:

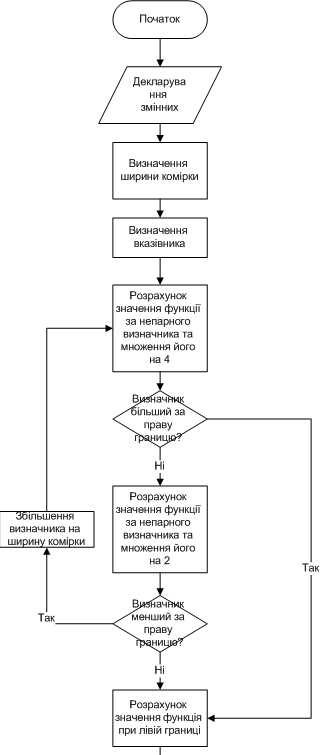


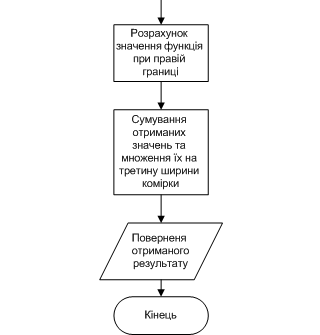


1. Метод трапецій: 

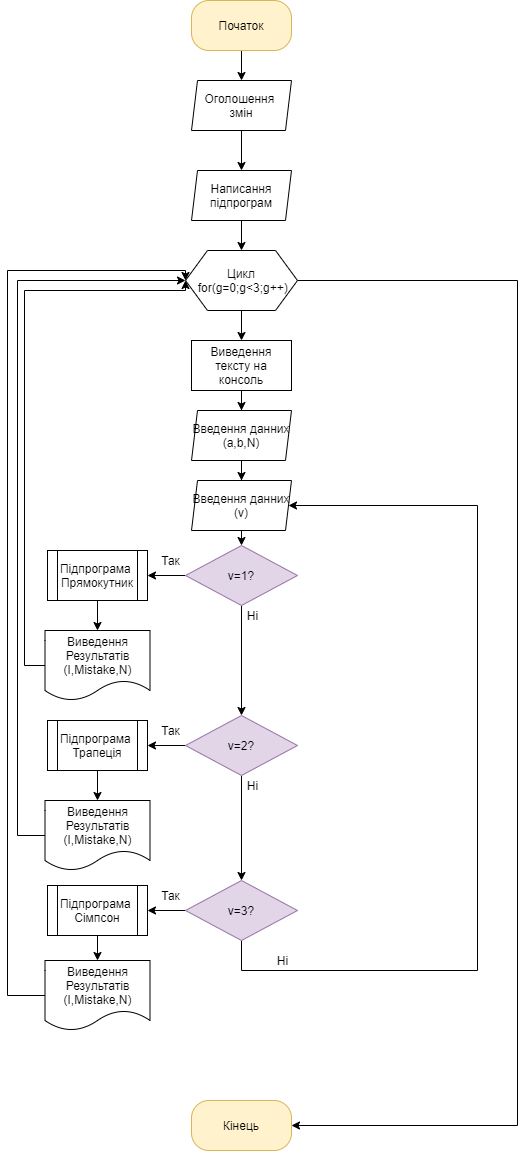


1. Метод Сімпсона (метод парабол): 





Основна блок-схема :



Обрана функція:

Чисельне значення аналітично обрахованого інтеграла: 0

Таблиця з результатами обрахунків заданого інтеграла всіма методами при 4 кількостях проміжків:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Метод | N=10 | N=100 | N=1000 | N=10000 |
| Прямокутник | 0,193158 | 0,200764 | 0,196099 | 0,196375 |
| Трапеція | 0,202573 | 0,203698 | 0,196599 | 0,196425 |
| Сімпсон | 0,209171 | 0,199683 | 0,198674 | 0,196587 |

Графік залежності значення інтеграла від кількості проміжків:

Код:

#include <stdio.h>

#include <stdlib.h>

#include <math.h>

#include <locale.h>

void info(double I,double Pomilka,int n )

{

printf("I = %lf\n",I);

printf("Pomilka= %lf\n",Pomilka);

printf("Kilkist shagiv = %d\n",n);

printf("Good job\n\n");

}

double Calc(double x)

{

return(pow(x,3)/((pow(x,8)+1)));

}

double Rectangles(int n,double A, double B)

{

double h,I,x;

h=(B-A)/n;

for(x = A; x <= B ;x = x + h)

{

I=I+Calc(x);

}

I=I\*h;

return I;

}

double Trapezium (int n,double A, double B)

{

double h,I=0,x;

h=(B-A)/n;

I=Calc(A)+Calc(B);

for(x = A; x <= B ;x = x + h)

{

I=I+Calc(x);

}

I=I\*h;

return I;

}

double Simpson(int n,double A, double B)

{

double h,I=0,x,II=0;

int i=0;

h=((B-A)/n);

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

{

if(i%2 != 0)

{

I=I+Calc(A+h\*i);

}

if(i%2 == 0)

{

II=II+Calc(A+h\*i);

}

}

I=((B-A)/(3\*n))\*(Calc(A)+4\*I+2\*II+Calc(B));

return I;

}

int main()

{

setlocale(LC\_ALL, "Rus");

int v,i,j=1,g,n;

double h,d,x,I=0.0,II=0.0,A,B;

for(g=0;g<3;g++)

{

printf("Hello, RE-12\n\n");

printf("1 - Squares\n2 - Trapezia\n3 - Simpson(parabols)\n");

printf("Choose your method: ");

scanf("%d",&v);

switch("%d",v)

{

case 1:

printf("Kilkist shagiv:");

scanf("%d",&n);

printf("\nlow border: ");

scanf("%lf",&A);

printf("\nhigh border: ");

scanf("%lf",&B);

for (j;j<=2;j++)

{

if(j==1)

{

I=Rectangles(n,A,B);

n=n+2;

}

II=Rectangles(n,A,B);

if(fabs(I-II)>0.00001 && fabs(I-II)<0.001)

{

II=fabs(I-II);

info(I,II,n);

break;

}

else

{

j=0;

}

}

break;

case 2:

printf("Kilkist shagiv:");

scanf("%d",&n);

printf("\nlow border: ");

scanf("%lf",&A);

printf("\nhigh border: ");

scanf("%lf",&B);

for(j; j<=2;j++)

{

if(j==1)

{

I = Trapezium(n,A,B);

n=n+2;

}

II = Trapezium(n,A,B);

if(fabs(I-II)>0.00001 && fabs(I-II)<0.001)

{

II = fabs(I-II);

info(I,II,n);

break;

}

else

{

j=0;

}

}

break;

case 3:

do

{

printf("Numbers must be only парные \n");

printf("Kilkist shagiv:");

scanf("%d",&n);

}

while(n%2!=0);

printf("\nlow border:");

scanf("%lf",&A);

printf("\nhigh border:");

scanf("%lf",&B);

for(j; j<=2;j++)

{

if(j==1)

{

I = Simpson(n,A,B);

n=n+2;

}

II = Simpson(n,A,B);

if(fabs(I-II)>0.00001 && fabs(I-II)<0.001)

{

II = fabs(I-II);

info(I,II,n);

break;

}

else

{

j=0;

}

}

break;

default:

printf("Not correct\n");

break;

}

}

return 0;

}

**Висновок:** ми навчилися використовувати цикли для вирішення визначених інтегралів, виводити їх значення на екран і змінювати його значення на потрібну похибку.

