Національний технічний університет України

«Київський політехнічний інститут імені Ігоря Сікорського»

**ЗВІТ З ЛАБОРАТОРНОЇ РОБОТИ №2**

**«ОБЧИСЛЕННЯ ВИЗНАЧЕНОГО ІНТЕГРАЛУ»**

з дисципліни: «Інформатика. Основи програмування та алгоритми»

Виконав: Мосіюк Є. О.

Група: РЕ–12

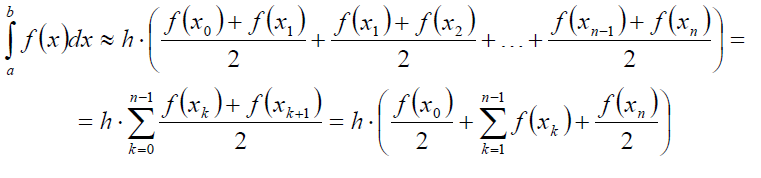
Викладач: доц. Катін П. Ю.

Київ – 2021

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

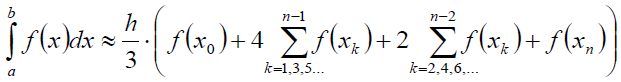
Блок-схеми:

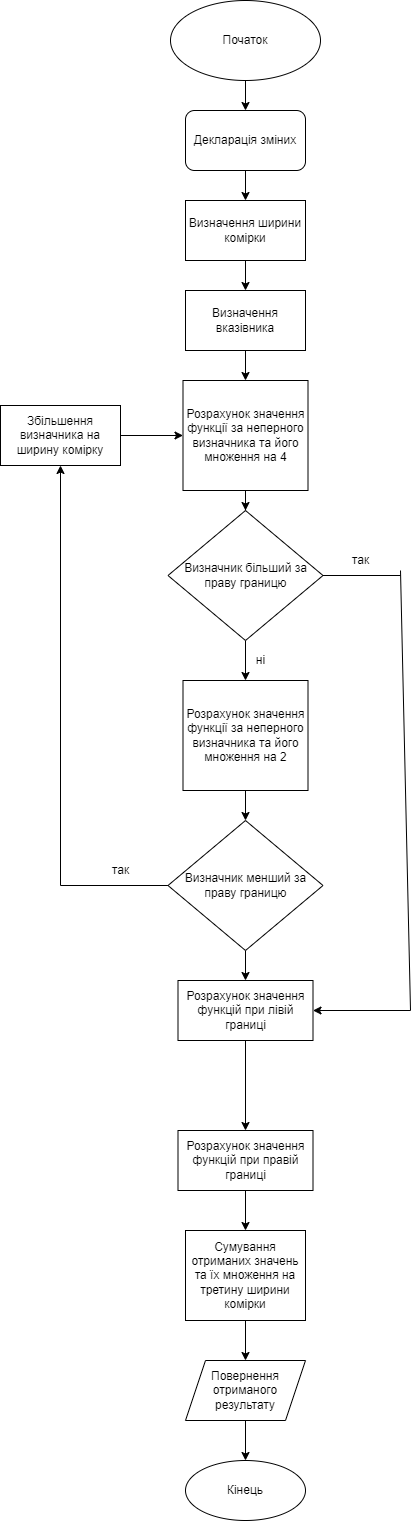
1.Метод Трапеції



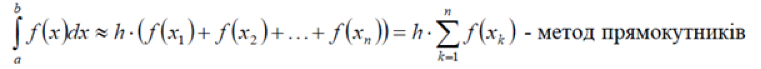


Метод сімпсона:





Метод прямокутника:





Основна блок-схема:



Обрана функція: x\*x\*exp(2\*x)

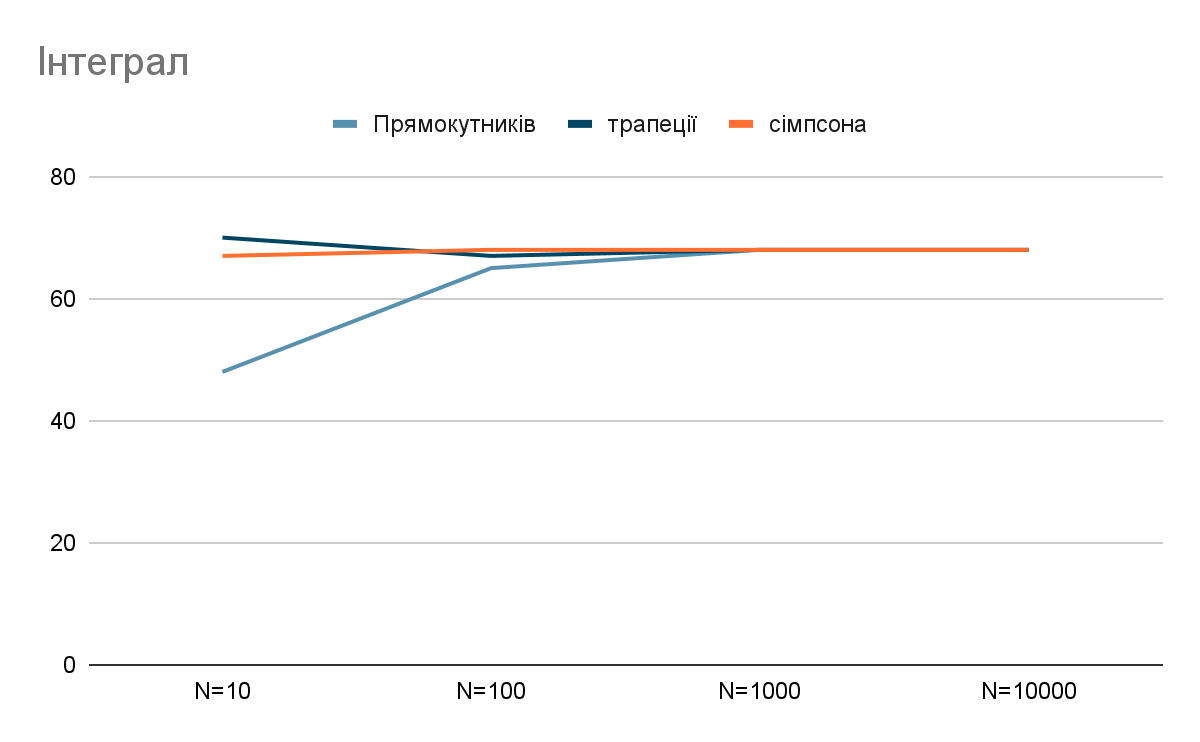
Обрахований аналітично інтеграл в символьному виді:(exp(2)-1)/4

чисельне значення аналітично обрахованого інтеграла:1.59726

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

| Метод | N=10 | N=100 | N=1000 | N=10000 |
| --- | --- | --- | --- | --- |
| Прямокутника | 48.331287 | 65.835600 | 67.779513 | 67.975850 |
| Трапеції | 70.170547 | 68.019526 | 67.997906 | 67.997690 |
| Сімпсона | 68.040866 | 67.997692 | 67.997688 | 67.997688 |

Графік залежності значення інтеграла від кількості проміжків:



Код:#include <stdio.h>

#include <stdlib.h>

#include <conio.h>

double func(double x){

return x\*x\*exp(2\*x);

}

double definitive(double x, double y)

{

return x/y;

}

int main()

{

const double analit\_integral=(exp(2)-1)/4;//=1.59726;

unsigned int N;

double x, f, delta, a, b;

printf("vedit pochatok promizka : ");

scanf("%lf", &a);

printf("vedit kinec promizka : ");

scanf("%lf", &b);

printf("vedit klk tochok rozbutya : ");

scanf("%u", &N);

double integral1=0;

delta=(b-a)/N;

x=a;

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

{

integral1+=func(x)\*delta;

x+=delta;

}

printf("integral(livuh pryamoknykiv)= %lf\n",integral1);

printf("pohubka integrala(livuh pryamoknykiv)= %lf\n",fabs(integral1-analit\_integral));

double integral2=0;

x=a+delta;

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

{

integral2+=func(x)\*delta;

x+=delta;

}

printf("integral(pravuh pryamoknykiv)=%lf\n",integral2);

printf("pohubka integrala(pravuh pryamoknykiv)= %lf\n",fabs(integral2-analit\_integral));

double integral3=func(a)/2+func(b)/2;

x=a+delta;

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

{

integral3+=func(x);

x+=delta;

}

integral3\*=delta;

printf("integral(trapeciy)=%lf\n",integral3);

printf("pohubka integrala(trapeciy)= %lf\n",fabs(integral3-analit\_integral));

double integral4=func(a)+func(b);

x=a+delta;

for(int i=1; i<N; i+=2)

{

integral4+=4\*func(x);

x+=2\*delta;

}

x=a+2\*delta;

for(int i=2; i<N; i+=2)

{

integral4+=2\*func(x);

x+=2\*delta;

}

integral4\*=delta/3;

printf("integral(Simpsona)=%lf\n",integral4);

printf("pohubka integrala(Simpsona)= %lf\n",fabs(integral4-analit\_integral));

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

double epsilon=0, I1=0,I2=0;

while(epsilon<0.00001 || epsilon>0.001){

printf("vedit epsilon 0.00001<eps<0.001 :");

scanf("%lf", & epsilon);

}

N=1;

do{

N++;

I1=0;

delta=(b-a)/N;

x=a;

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

{

I1+=func(x)\*delta;

x+=delta;

}

N+=2;

I2=0;

delta=(b-a)/N;

x=a;

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

{

I2+=func(x)\*delta;

x+=delta;

}

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

printf("N (LIVUH PRYAMOKYTNYKIV) %u\n",N-2);

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

N=1;

do{

N++;

I1=0;

delta=(b-a)/N;

x=a+delta;

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

{

I1+=func(x)\*delta;

x+=delta;

}

N+=2;

I2=0;

delta=(b-a)/N;

x=a+delta;

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

{

I2+=func(x)\*delta;

x+=delta;

}

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

printf("N (pravuh PRYAMOKYTNYKIV)%u\n",N-2);

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

N=1;

do{

N++;

delta=(b-a)/N;

I1=func(a)/2+func(b)/2;

x=a+delta;

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

{

I1+=func(x);

x+=delta;

}

I1\*=delta;

N+=2;

delta=(b-a)/N;

I2=func(a)/2+func(b)/2;

x=a+delta;

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

{

I2+=func(x);

x+=delta;

}

I2\*=delta;

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

printf("N (TRAPECIY)%u\n",N-2);

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

N=1;

do{

N++;

delta=(b-a)/N;

I1=func(a)+func(b);

x=a+delta;//x1

for(int i=1; i<N; i+=2)

{

I1+=4\*func(x);

x+=2\*delta;

}

x=a+2\*delta;//x2

for(int i=2; i<N; i+=2)

{

I1+=2\*func(x);

x+=2\*delta;

}

I1\*=delta/3;

N+=2;

delta=(b-a)/N;

I2=func(a)+func(b);

x=a+delta;//x1

for(int i=1; i<N; i+=2)

{

I2+=4\*func(x);

x+=2\*delta;

}

x=a+2\*delta;//x2

for(int i=2; i<N; i+=2)

{

I2+=2\*func(x);

x+=2\*delta;

}

I2\*=delta/3;

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

printf("N (SIMPSONA)%u\n",N-2);

return 0;

}