МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ

«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ

ІМЕНІ ІГОРЯ СІКОРСЬКОГО»

Кафедра мікроелектроніки

**Лабораторна робота №18**

**Варіант 8**

Виконав:

Студент 2-го курсу

ФЕЛ

Групи ДП-81

Рибін З.Ю.

Перевірив:

Доц. Домбругов М. Р.

Київ – 2020

**Чисельне інтегрування.**

**Формули прямокутників, трапеціи**̆**, Сімпсона**

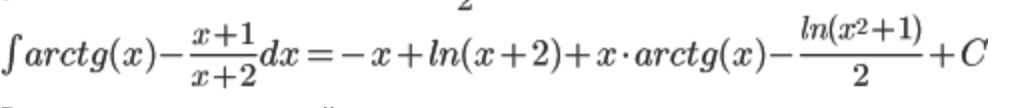
**Мета роботи:** вивчення алгоритмів Ньютона-Котеса чисельного інтегрування функції однієї змінної: квадратурних формул прямокутників, трапецій, Сімпсона та дослідження

поведінки їх похибок.

**Що зробити:** обчислити інтеграл аналітично і за допомогою складеної квадратурної формули при різних кількостях підінтервалів n. Впевнитися у взаємоузгодженості отриманих результатів. Порівняти розбіжності між аналітичним і наближеними результатами при різних n і визначити

порядок точності квадратурної формули.

Варіант 8 : f (x) = arctg x – (x+1)/(x+2)

**Невизначений інтеграл:**

#include <iostream>

#include <fstream>

#include <math.h>

#include <thread>

#include <iomanip>

template <class T>

T f(T x){

return atan(x)-(x+1.0)/(x+2.0);

}

template <class T>

T integral(T x){

return x\*atan(x)-x+log(x+2)-log(x\*x+1)/2;

}

template <class T>

T integral\_ab(T a, T b){

return integral(b)-integral(a);

}

template <class T>

T integral\_Simpson(T a, T b, int n){

T res = 0;

for(int i=0; i<=n-2; i+=2){

T b1 = a + ((b - a)/n)\*i;

T b2 = a + ((b - a)/n)\*(i+1);

T b3 = a + ((b - a)/n)\*(i+2);

T h = b2 - b1;

res = res + h\*(f(b1)+f(b2)\*4.0+f(b3))/3.0;

}

return res;

}

template <class T>

void lab18(T a, T b, const std::string& file\_name){

T I = integral\_ab(a, b);

std::cout << std::setprecision (15) << I << std::endl;

std::ofstream of(file\_name);

for(int n = 2;n<40000; n+=2){

T Is = integral\_Simpson(a, b, n);

T e = fabs(I - Is);

of << n << "\t" <<std::setprecision (15)<< e << std::endl;

}

}

int main(){

std::thread t1([](){

lab18<float>(0.0f, 3.0f, "lab18\_float.txt");

});

std::thread t2([](){

lab18<double>(0.0d, 3.0d, "lab18\_double.txt");

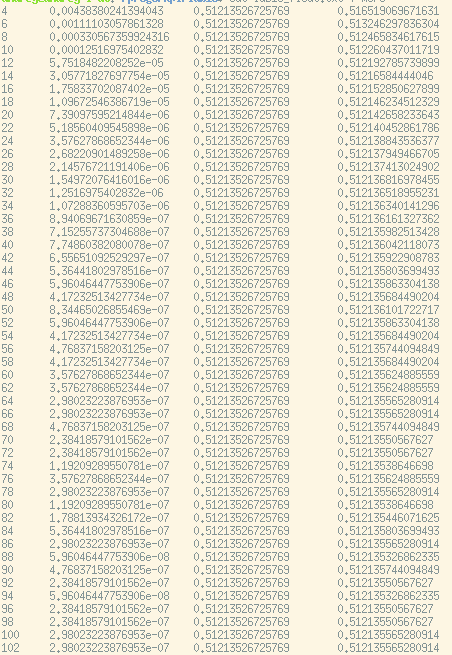
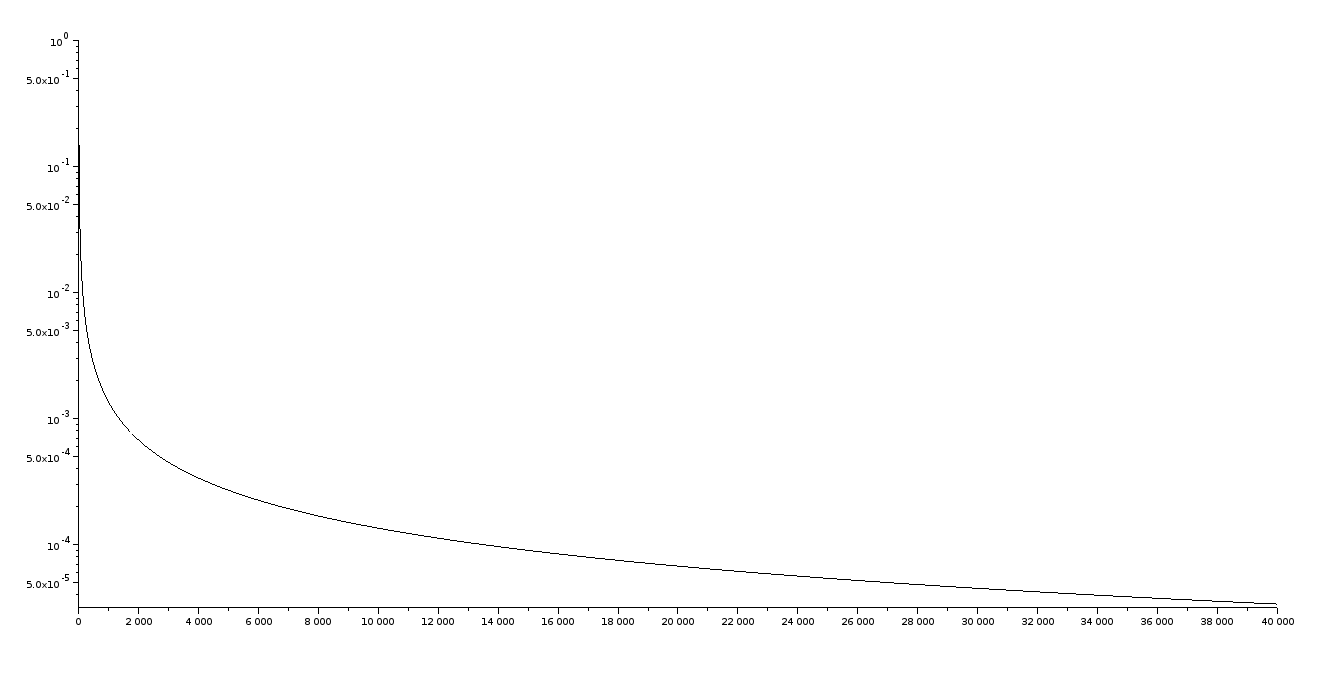
});

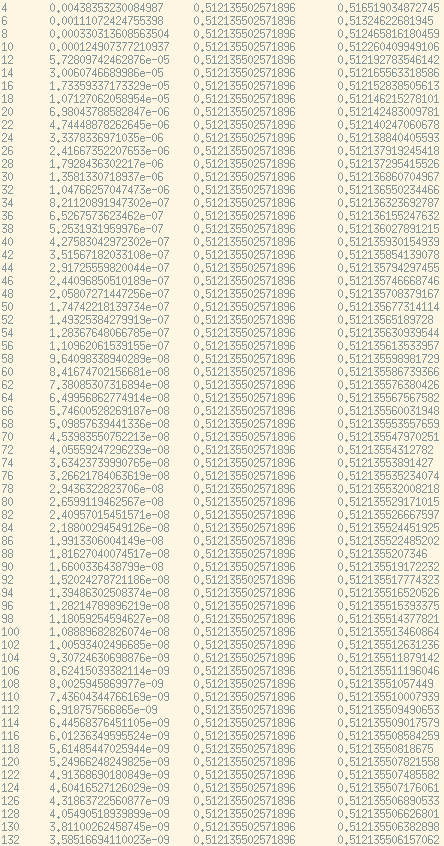
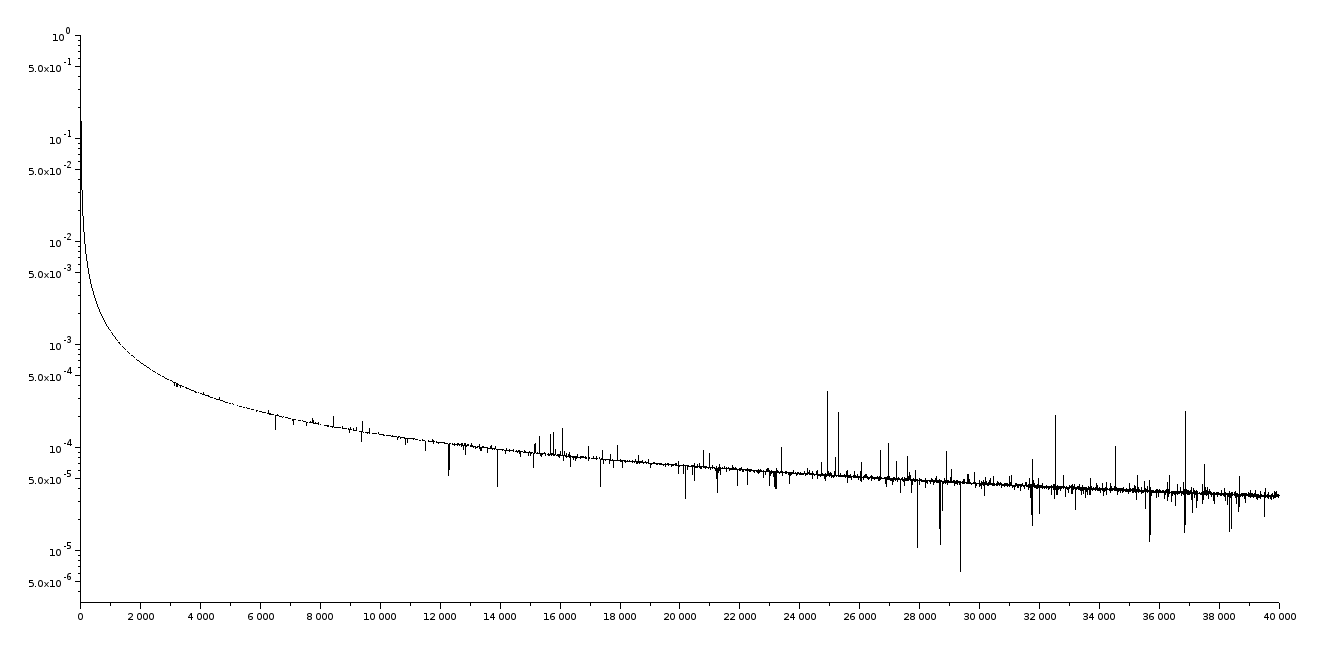
t1.join();

t2.join();

return 0;

}

Графік для данних виведених з точністью **double**

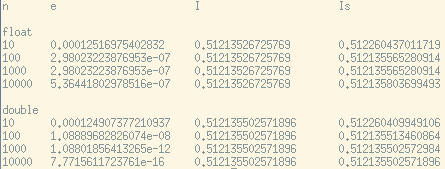
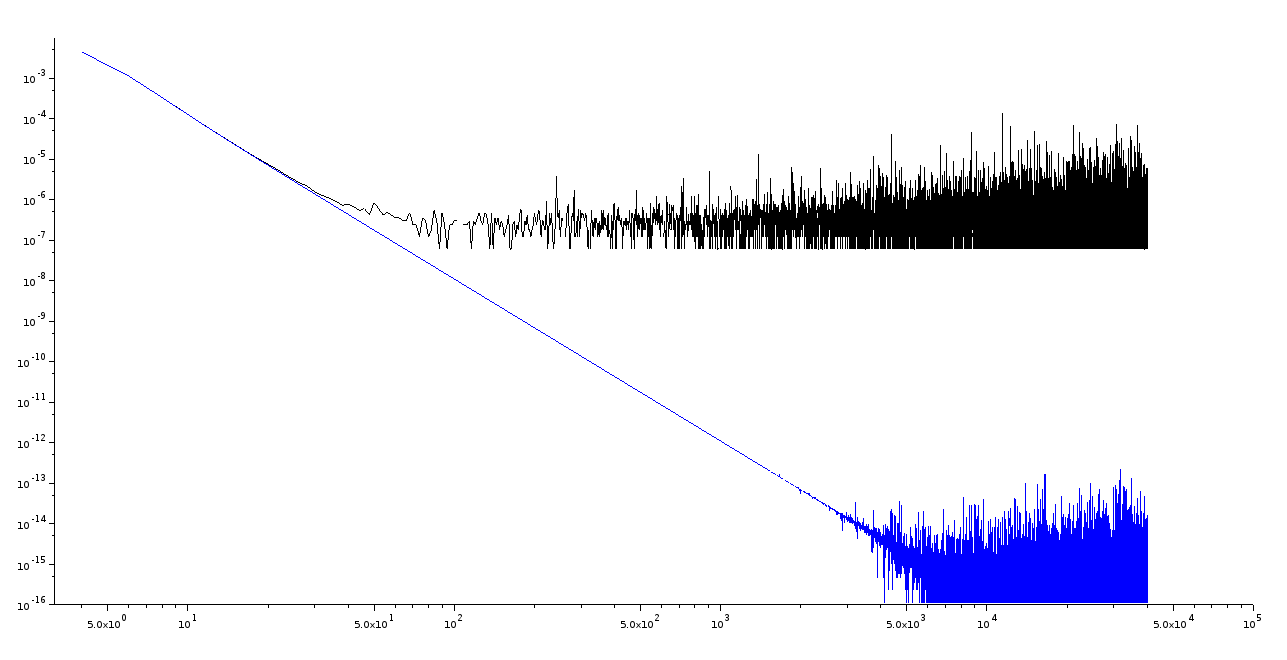
Графік для данних виведених з точністью **float**

**Висновок:** у цій лабораторній роботі я вивчив алгоритм Ньютона-Котеса чисельного інтегрування функції однієї змінної методом Сімпсона. Також вивів два графікі для Двох точностей, а саме для **Double та Float** і помітив, що приблизно після відміти в 3000 графік для значень з точністью **Float** починае стрибати то вниз то вгору(на графіку зточністью **Double** це не відно откільки ми не досягли бажаної точності), а це означае те, що машинна похибка стала більшою ніж та що виникае при інтегруванні, тому подальше інтегрування не має сенсу.

**Все це дуже цікаво, але де ж величина самого інтеграла?**

**Аналітичного значення та результатів чисельних розрахунків при *n* = 10, 100 та 1000 було б достатньо.**

**Дуже добре, що на графіках вісь похибки Ви здогадалися зробити в логарифмічному масштабі, але вісь *n* також було б краще робити логарифмічною. Ви б побачили багато цікавого.**

**Виправляйте.**

**По-перше, не завадило б зазначити межі інтегрування. Підозрюю, що це від 0 до 3, але я не впевнений.**

**В самій программі написано що межі інтегрування є від 0 до 3 (ви праві) я просто це не зазначив:**

**а=0**

**b=3**

**По-друге, щось не клеїться. У Вашій верхній таблиці Ви зазначаєте, що похибка при n=10 дорівнює 0.124, а в нижній – 0.000124, тобто в 1000 раз менше.**

**При n=100 похибка зверху 0.013, а знизу – 3**´**10-7, це близько до машинного епсілон, тобто в 100 000 разів менше.**

**Захаре, Вас наказати за зухвалу брехню? (вібачте та то не брехня, просто е в кінці кудись зникло тому значення вийшли не ті, а так усе ж правильно) )**

**Ви ж підписували кодес наукової доброчесності!**

**Мені соромно за Вас і огидно.**