«Київський політехнічний інститут імені Ігоря Сікорського»

Факультет електроніки

Кафедра мікроелектроніки

Лабораторна робота №18

Варіант №21

Виконав: студент групи ДП-82

Мнацаканов Антон

Перевірив: Домбругов М.Р.

Київ-2020

**Чисельне інтегрування.Формули прямокутників, трапецій, 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.

Хід роботиСнимок экрана 2020-05-10 в 14.51.34.png

Головний код на С:

#include <stdio.h>

#include <math.h>

#define DOUBLE double

DOUBLE my\_func(DOUBLE x)

{

return 10.0\*x\*x\*exp(-x)-3.0\*x;

}

DOUBLE my\_func\_integral(DOUBLE x){

return 10.0\*(-(x\*x)-2.0\*x-2.0)\*exp(-x)-(3.0\*x\*x)/2.0;

}

DOUBLE my\_func\_integral\_ab(DOUBLE a, DOUBLE b){

return my\_func\_integral(b)-my\_func\_integral(a);

}

DOUBLE near\_integral(DOUBLE a, DOUBLE b, int n){

DOUBLE res = 0;

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

DOUBLE beg = a + ((b - a)/n)\*i;

DOUBLE end = a + ((b - a)/n)\*(i+1);

DOUBLE h = end - beg;

res = res + h/2\*(my\_func(beg)+my\_func(end));

}

return res;

}

int main()

{

DOUBLE a = -1.0;

DOUBLE b = 2.0;

printf("\n\nМоя функция:-----------------------------> 10x^2exp(-x)-3x\n");

DOUBLE I = my\_func\_integral\_ab(a, b);

printf("Результат интегрирования моей функции----> %e\n", I);

int n = 1;

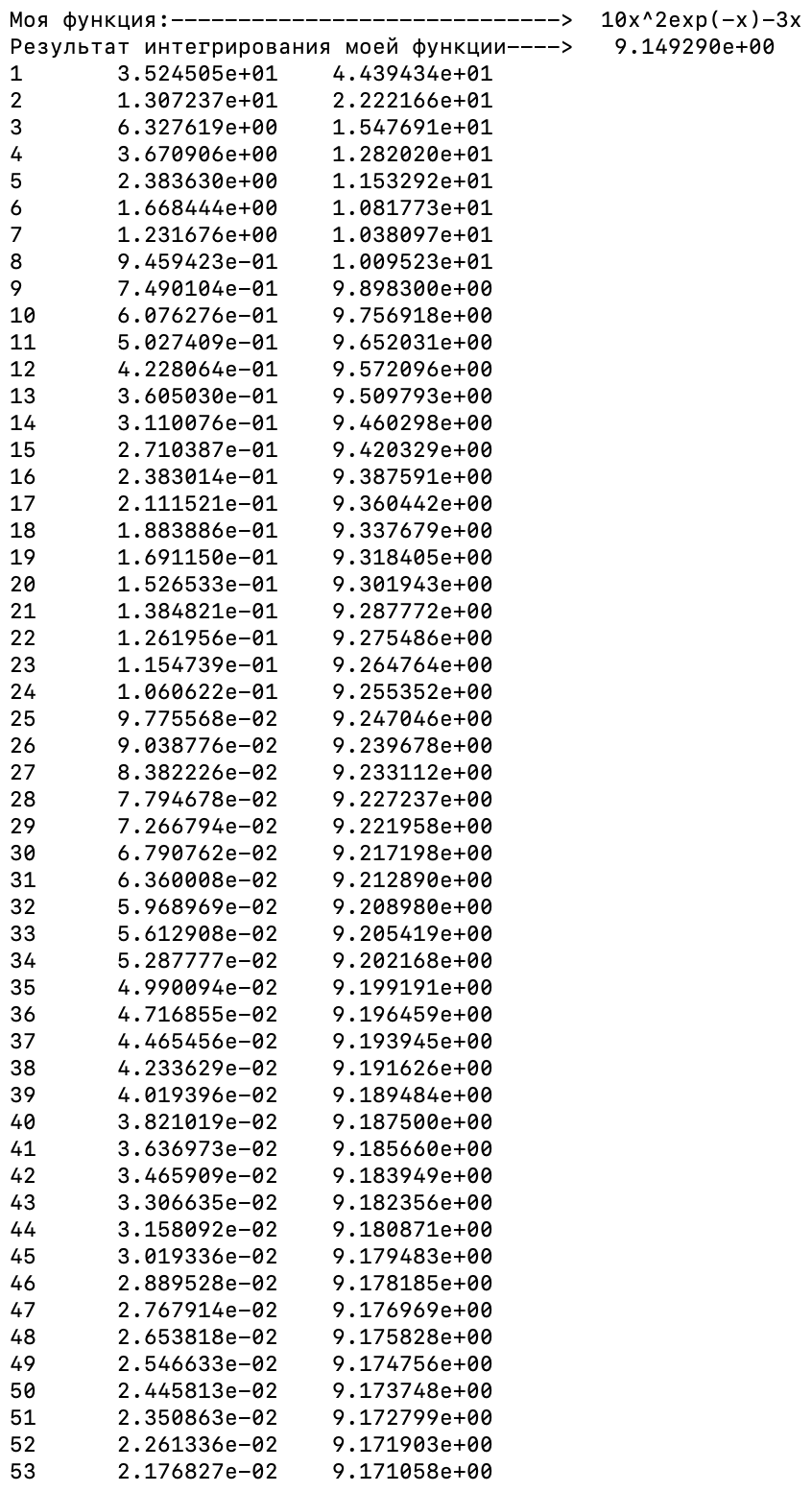
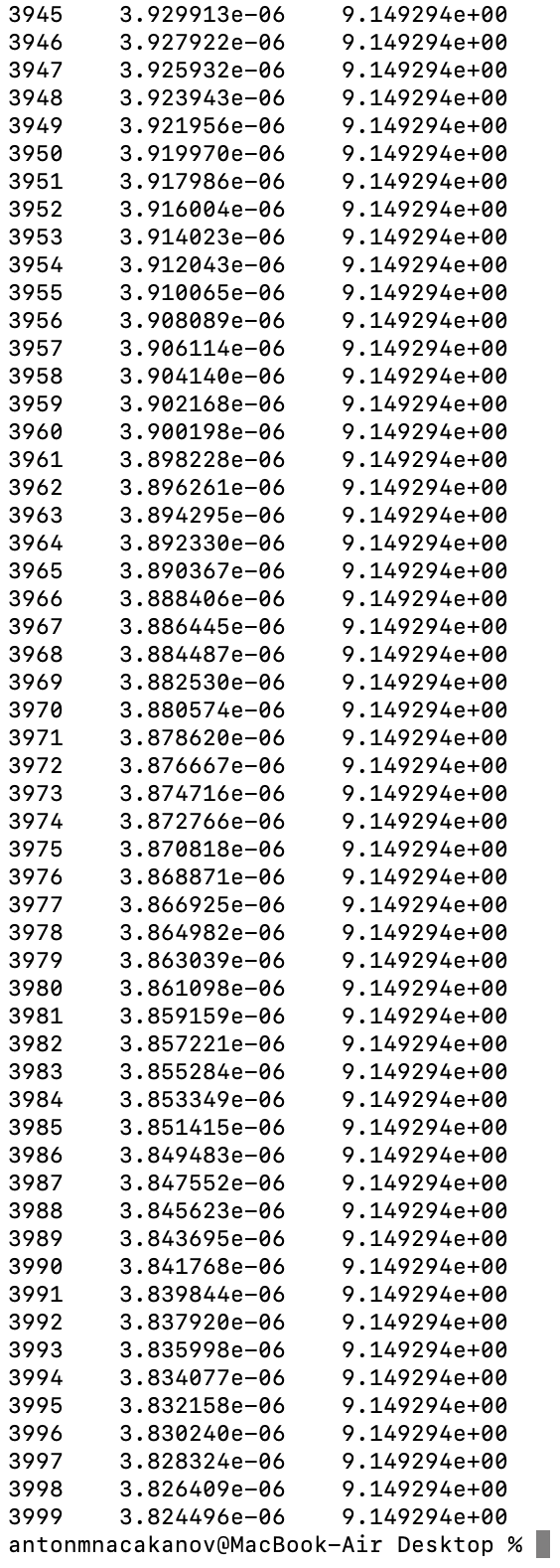
for(;n<4000; n+=1){

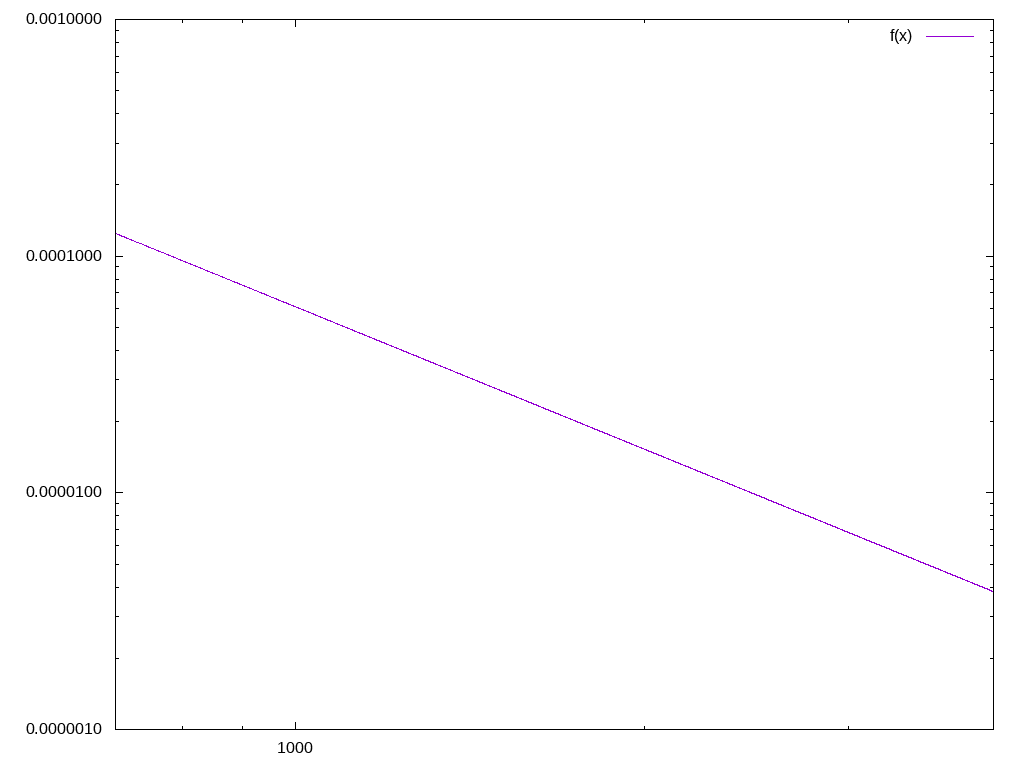
DOUBLE It = near\_integral(a, b, n);

DOUBLE e = It - I;

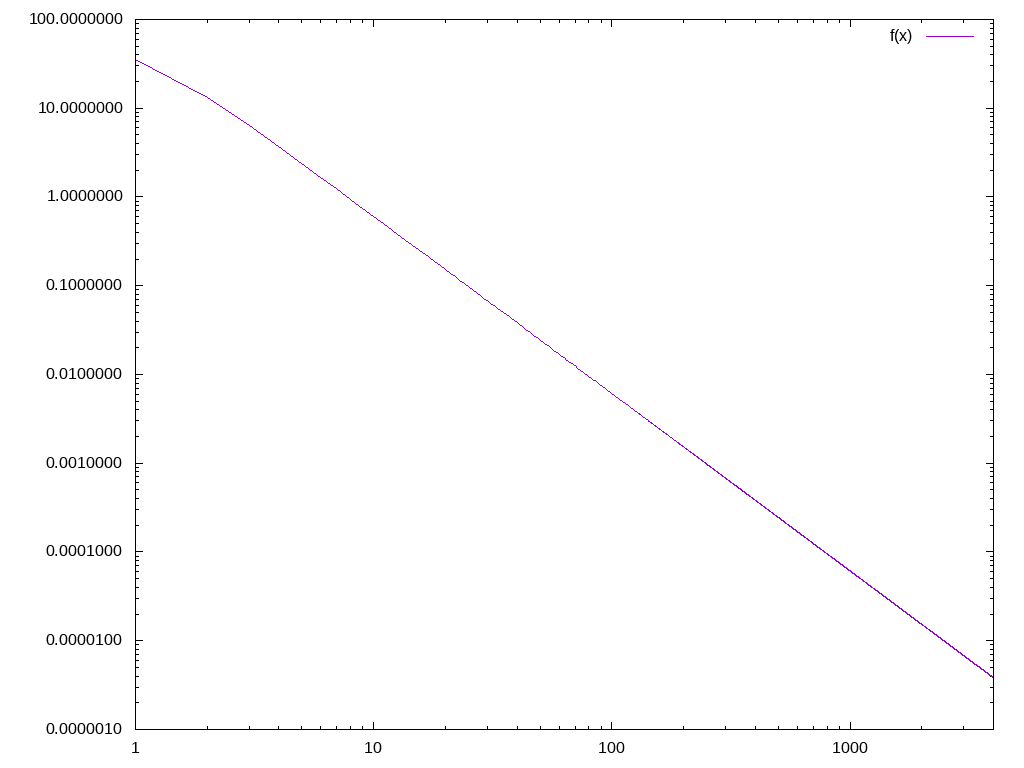
printf("%d\t%e\n", n, e);

}

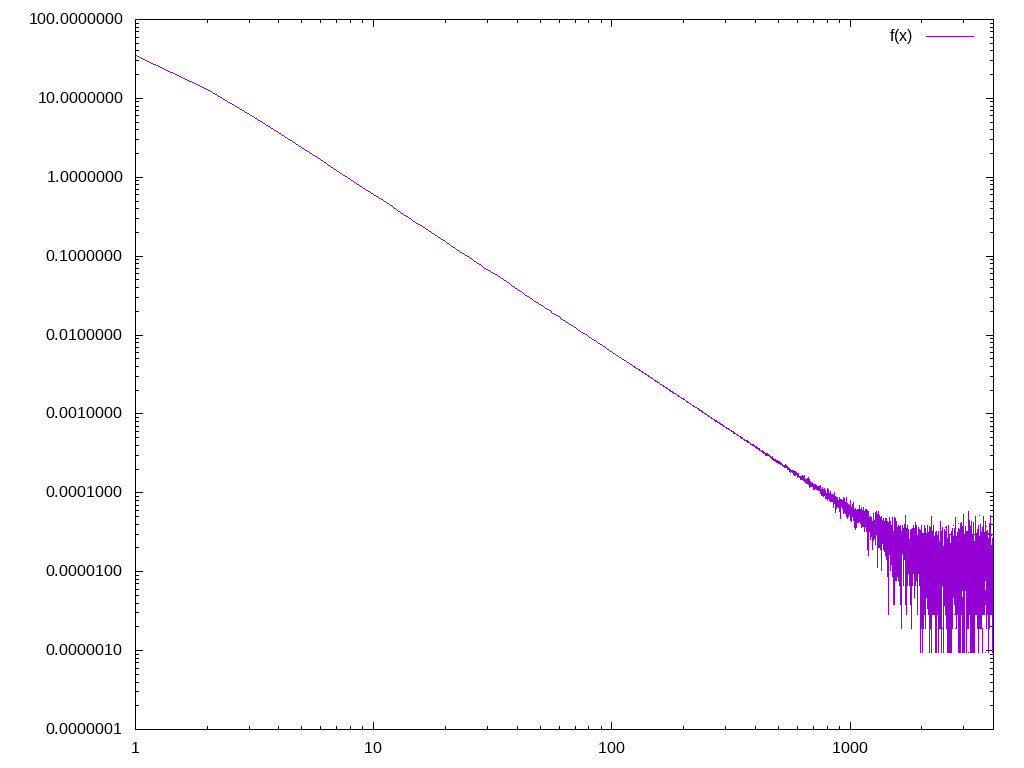
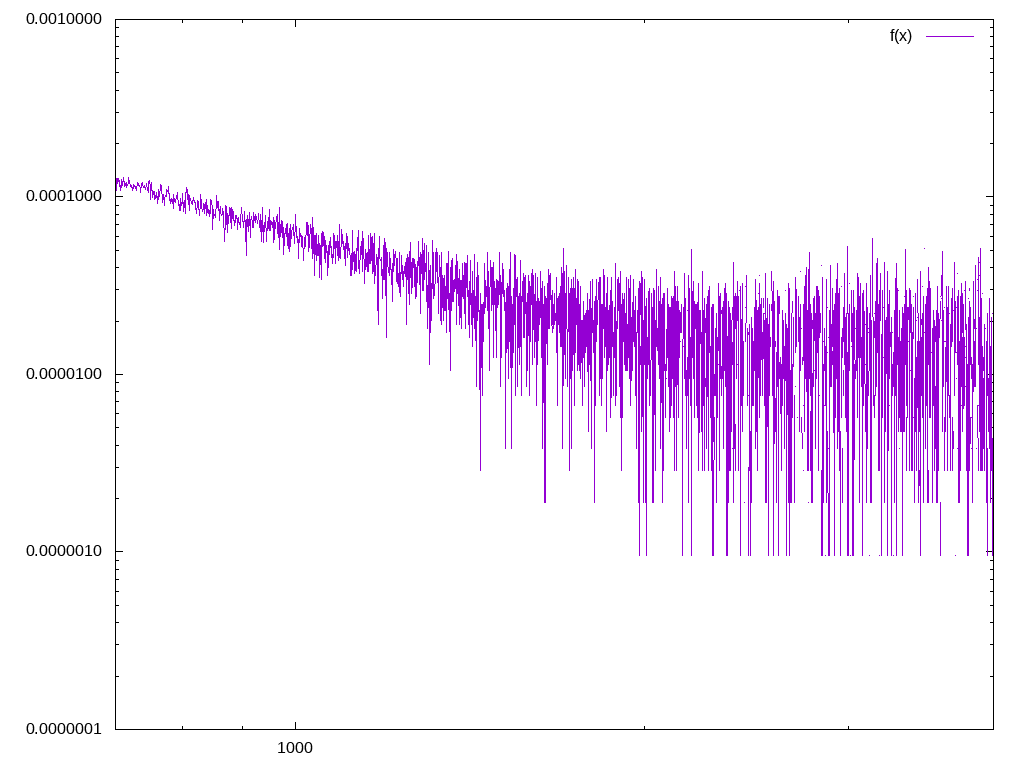
return 0;} **Результати**

Тут була використана точність **double**

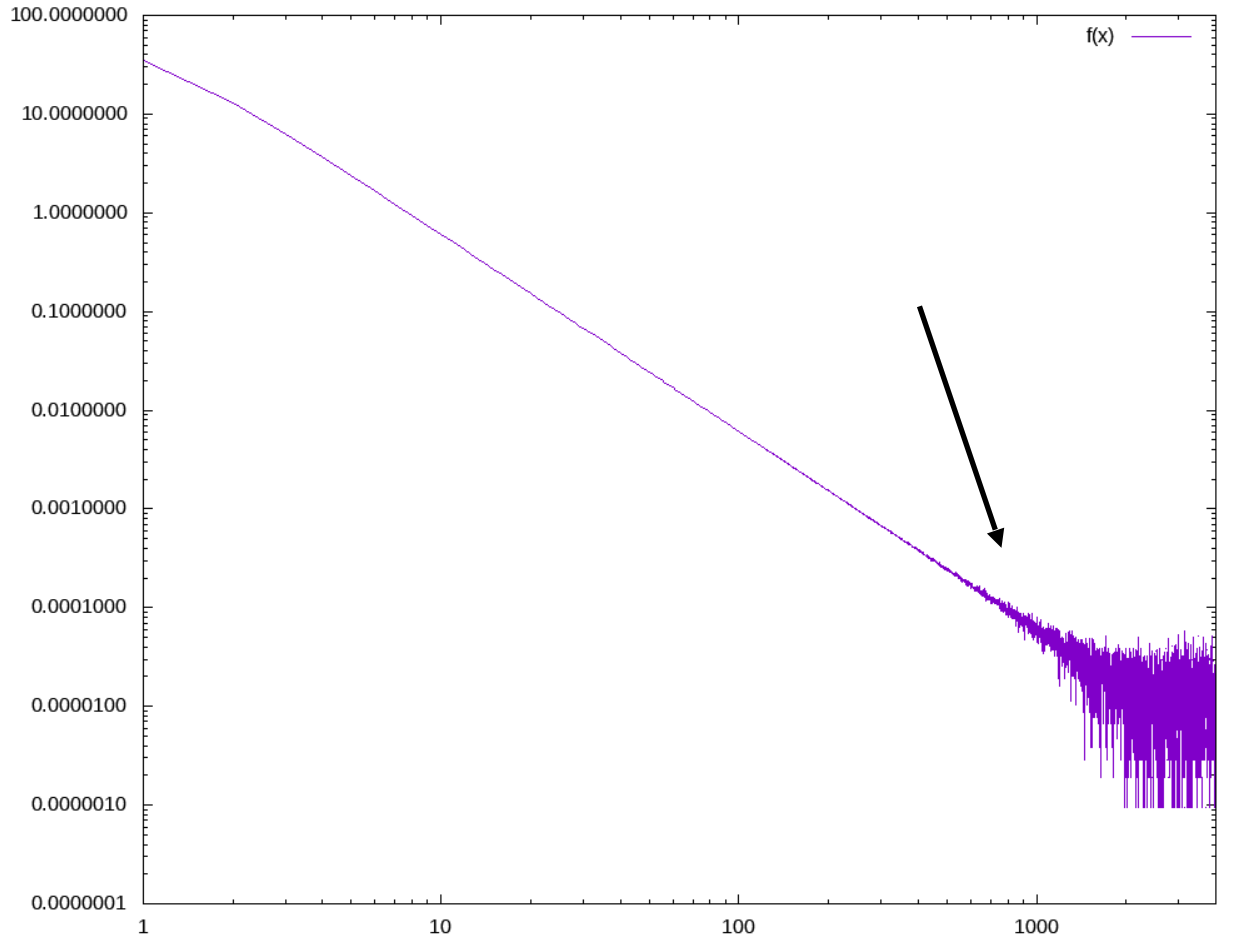
Починаючи з 700



Починаючи з 1

А тут так само тільки з **float**

**Висновок:** спочатку я запрограмуйвав обчислення I за аналітичним виразом своєї функції потім склав програму для обчислення наближеного значення цього ж інтегралу (І) за допомогою формули трапецій і отримав результат Iнаближ (вивів їх 3 стовпчиком) також обчислив похибку квадратурної формули e = I–Iнаближ та дослідив залежність e від числа підінтервалів n.

Виконавши всі обчислення з подвійною точністю можу сказати що похибка округлення стає більше ніж похибка інтегрування приблизно після цієї точки:

Тому подальше интегрування є безглуздим