Національний технічний університет України

«Київський політехнічний інститут імені Ігоря Сікорського»

Кафедра теоретичних основ радіотехніки

**ЗВІТ З ЛАБОРАТОРНОЇ РОБОТИ №3**

з дисципліни: «Інформатика 1»

|  |  |
| --- | --- |
|  | Виконав: Філобоченко Ліза  Група: РЕ-11  Викладачі: доцент Катін П.Ю.  Оцінка: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Підпис: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |

Київ – 2021

**Мета роботи:** розробка програми розв’язання нелінійних рівнянь

**Ключові моменти:**

**1**. Пдключаємо бібліотеки <conio.h> та <time.h> перша повертає значення часу в даний момент, друга для того щоб отримати затрачений час та знаходження корення.

CLOCKS\_PER\_SEC змінна задана за допомогою бібліотеки <time.h> й означає 1000

**2**. Використовую getch для запису у змінну яку ввів користувач. Використовуючи таблицю Аскі для зчитування с клавіатури симболів, в нашому випадку ESC, в таблиці позначене під 27.

if((int)i==27) Ми прирівнюємо чи дорівнює і=27.

**3**. clock\_t T2=clock() тут записую Т2 значення часу в даний момент

(double)(T2-T1)/CLOCKS\_PER\_SEC отримаємо значення за який час программа отримала корінь

**Код:**

#include <stdio.h>

#include <stdlib.h>

#include <math.h>

#include <conio.h> //библиотека для getch

#include <time.h>

double Funk(double x) // Основная функция

{

return 5\*pow(x,3)-2\*pow(x,2)+3.5\*x+15;

}

void Dump(double Xi, double fault,int N, double Timer)

{

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

printf("Fault = %lf \n", fault);

printf("Count of iterations = %d \n", N);

printf("Calculation time = %lf \n", Timer);

}

int main()

{

// My function y=5\*pow(x,3)-2\*pow(x,2)+3.5\*x+15

double X1, X2, Xi, Timer,fault;

unsigned int N, n=0, varik;

char i; //символ (аля строка)

printf("1-chord method \n 2-half chord method \n");

scanf("%d",&varik);

switch(varik)

{

case 1:

do

{

printf("Left limit: \n");

scanf("%lf", &X1);

printf("Right limit: \n");

scanf("%lf",&X2);

}

while(Funk(X1)>0 && Funk(X2)<0); //сравниваем допустимые параметры,должны иметь разные знаки. например Х1=-1 а Х2=1

printf("Fault: \n");

scanf("%lf", &fault);

printf("Max count of iterations: \n");

clock\_t T1= clock();// записываем в перременную Т1 значение функции слок, возвращает значение времени с начала запуска программы.

scanf("%d", &N);

do

{

Xi=((Funk(X2)\*X1)-(Funk(X1)\*X2))/(Funk(X2)-Funk(X1));

if(Funk(Xi)>0)

{

X2=Xi;

}

else

{

X1=Xi;

}

if(n==N)

{

printf("Maximum number of iterations reached \n"); //Достигнуто максимальное количество итераций

printf("Press only the ESC key to complete or any other to continue \n"); //Нажмите только клавищу ЕСК для завершения или любую другую для продолжения

i=getch(); //програма ожидает любого нажатия клавишь

if((int)i==27)//в таблице аски ESC имеет номер 27, мы переводим чар в инт и сравниваем их значения

{

system("cls"); //чистим консоль

break;

}

else

{

system("cls");

}

}

n++;

}

while(fabs(Funk(Xi))>fault);

clock\_t T2=clock();

Dump(Xi, fabs(Funk(Xi)), n, (double)(T2-T1)/CLOCKS\_PER\_SEC);//Вызиваем функцию и передаем в неё 4 параметра

break;

case 2:

do

{

printf("Left limit: \n");

scanf("%lf", &X1);

printf("Right limit: \n");

scanf("%lf",&X2);

}

while(Funk(X1)>0 && Funk(X2)<0);

printf("Fault: \n");

scanf("%lf", &fault);

printf("Max count of iterations: \n");

clock\_t T11=clock();

scanf("%d", &N);

do

{

Xi=(X1+X2)/2;

if(Funk(Xi)>0)

{

X2=Xi;

}

else

{

X1=Xi;

}

if(n==N)

{

printf("Maximum number of iterations reached \n");

printf("Press only the ESC key to complete or any other to continue \n");

i=getch();

if((int)i==27)

{

system("cls");

break;

}

else

{

system("cls");

}

}

n++;

}

while(fabs(Funk(Xi))>fault);

clock\_t T22=clock();

Dump(Xi, fabs(Funk(Xi)), n, (double)(T22-T11)/CLOCKS\_PER\_SEC); //CLOCKS\_PER\_SEC = константа что равна 1000

break;

}

return 0;

}