Министерство образования и науки Российской Федерации

Новосибирский национальный исследовательский государственный университет

ЭВМ и периферийные устройства

Отчет по лабораторной работе № 1

«Определение времени работы прикладных программ»

Студент: Комбаров Олег Алексеевич, 24210

Преподаватель: Мичуров Михаил Антонович

Новосибирск, 2025.

## **Цель работы**

1. Изучение методики измерения времени работы подпрограммы.

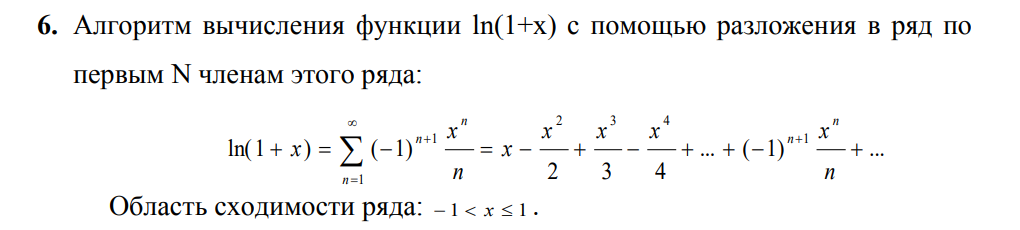
2. Изучение приемов повышения точности измерения времени работы подпрограммы.

3. Изучение способов измерения времени работы подпрограммы.

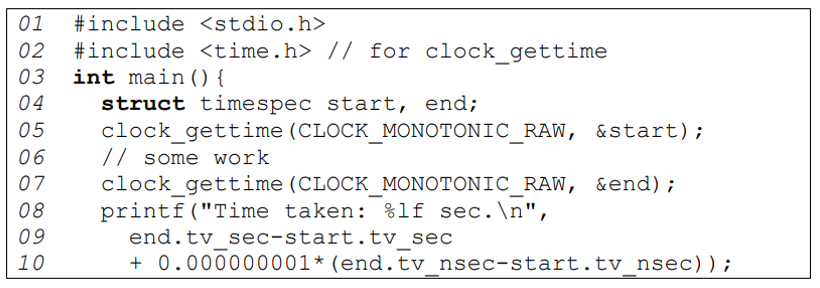
4. Измерение времени работы подпрограммы в прикладной программе.

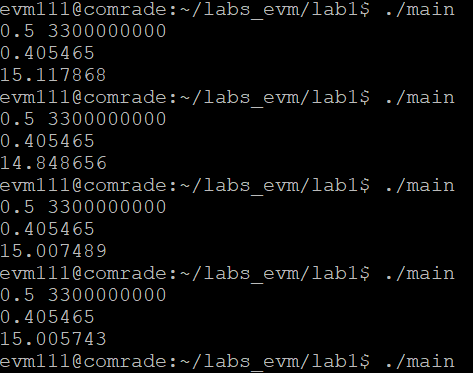
## **Описание работы**

В качестве задания я выбрал:



Я использовал метод с использованием библиотеки time.h



Я подобрал n, при котором программа выполняется примерно 15 сек.  
**n = 3300000000**Для повышения точности результата, провел замер времени несколько раз.Тесты показали:

Время выполнения программы = 14.848656  
Относительная погрешность = 0,000001

Абсолютная погрешность = 0.0000000000000673461625…

## **Приложения**

Ссылка на репозиторий github:

https://github.com/CherNuShka-825/evm-labs

Команда компиляции: gcc main.c -o main  
 Команда для запуска: ./main

Код программы:

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32 | #include <stdio.h>  #include <time.h>  **int** **main**(**void**) {  **double** x;  **long** **long** n;  **if** (scanf("%lf", &x) != **1**) **return** **-1**;  **if** (scanf("%lld", &n) != **1**) **return** **-1**;  **if** (x <= **-1.0** || x > **1.0**) {  **return** **-2**;  }  **if** (n < **1**) {  **return** **-3**;  }  **double** ln = **0**;  **double** x\_pow = x;  **struct** **timespec** start, end;  clock\_gettime(CLOCK\_MONOTONIC, &start);  **for** (**long** **long** i = **1**; i <= n; i++) {  **if** (i % **2** == **0**) {  ln -= x\_pow / i;  } **else** {  ln += x\_pow / i;  }  x\_pow \*= x;  }  clock\_gettime(CLOCK\_MONOTONIC, &end);  printf("%lf**\n**", ln);  printf("%lf", end.tv\_sec - start.tv\_sec + (**0.000000001** \* (end.tv\_nsec - start.tv\_nsec)));  **return** **0**;  } |