

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
НОВОСИБИРСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
Факультет информационных технологий
Кафедра параллельных вычислений**

ОТЧЕТ

О ВЫПОЛНЕНИИ ЛАБОРАТОРНОЙ РАБОТЫ

«Определение времени работы прикладных программ»

студента Бородина Артёма Максимовича 2 курса, 19205 группы
Направление 09.03.01 – «Информатика и вычислительная техника»

Преподаватель:
к.т.н, доцент
А.Ю. Власенко

Новосибирск 2020

СОДЕРЖАНИЕ

ЦЕЛЬ	3
ЗАДАНИЕ	3
ОПИСАНИЕ РАБОТЫ	4
ЗАКЛЮЧЕНИЕ	5
Приложение 1. Код программы, вычисляющей число π	6
Приложение 2. Утилита <i>time</i>	7

ЦЕЛЬ

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.

ЗАДАНИЕ

1. Написать программу на языке C или C++, которая реализует выбранный алгоритм из задания.
2. Проверить правильность работы программы на нескольких тестовых наборах входных данных.
3. Выбрать значение параметра N таким, чтобы время работы программы было порядка 15 секунд.
4. По приведенной методике определить время работы подпрограммы тестовой программы с относительной погрешностью не более 1%.
5. Составить отчет по лабораторной работе.

1 вариант:

Алгоритм вычисления числа π с помощью разложения в ряд (ряд Грегори-Лейбница) по формуле Лейбница N первых членов ряда

ОПИСАНИЕ РАБОТЫ

1. Были изучены методы измерения времени работы программ с использованием системных таймеров, виды погрешностей (абсолютная, относительная), способы определения времени работы с их достоинствами и недостатками.
2. Мною были выбраны методики измерения времени с помощью утилиты `time` и библиотечных функций `ctime`.
3. Была написана программа на языке C++ (см. Приложение 1), вычисляющая число π по формуле Лейбница с использованием библиотеки замера времени.
4. Для получения времени работы программы в ~15 секунд потребовалось $N=360000000$ (Приложение 2).
5. После компиляции программы был использован таймер ОС Linux.

ЗАКЛЮЧЕНИЕ

В ходе этой лабораторной работы были изучены методы исследования времени работы программ или их частей, способы уменьшения влияния посторонних факторов на время выполнения программ.

Приложение 1. Код программы, вычисляющей число π

```
/home/evmpu/1~ab2/main.cpp  [----] 45 L:[ 1+21 22/ 28] *(523 / 691b) 0059 0x03B UTF-[*][X]
#include <iostream>
#include <ctime>
#include <cmath>
#include <cstdlib>

long double LeibnizFormula(int n) {
    long double pi = 0;
    for (int i = 0; i < n; ++i) {
        long double appendix = 1 / (long double) (2 * i + 1);
        pi += appendix*pow(-1,i) ;
    }
    pi *= 4;
    return pi;
}

int main(int argc, char *argv[]) {
    int N = atoi(argv[1]);
    long double startTime = clock();
    long double pi = LeibnizFormula(N);
    long double endTime = clock();
    long double runTime = endTime - startTime;
    //std::cout << "N = " << N << std::endl;
    //std::cout << "Pi = " << pi << std::endl;
    //std::cout << "Time = " << runTime/1000000 << "sec";
    return 0;
}
```

Приложение 2. Утилита time

```
evmpu@fitnsu-1135-2:~/19205/Borodin/Lab2$ g++ main.cpp -o main.out
evmpu@fitnsu-1135-2:~/19205/Borodin/Lab2$ time ./main.out 10000000

real    0m0.462s
user    0m0.458s
sys     0m0.004s
evmpu@fitnsu-1135-2:~/19205/Borodin/Lab2$
```