###### 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 (ки) 2 курса, 19201 группы

**Хаецкой Дарьи Владимировны**

Направление 09.03.01 – «Информатика и вычислительная техника»

Преподаватель:

А.Ю. Власенко

Новосибирск 2020

**СОДЕРЖАНИЕ**

[ЦЕЛЬ 3](#_gjdgxs)

[ЗАДАНИЕ 3](#_30j0zll)

[ОПИСАНИЕ РАБОТЫ 4](#_1fob9te)

[ЗАКЛЮЧЕНИЕ 5](#_3znysh7)

ПРИЛОЖЕНИЕ 1. Исходный код программы для тестирования 6

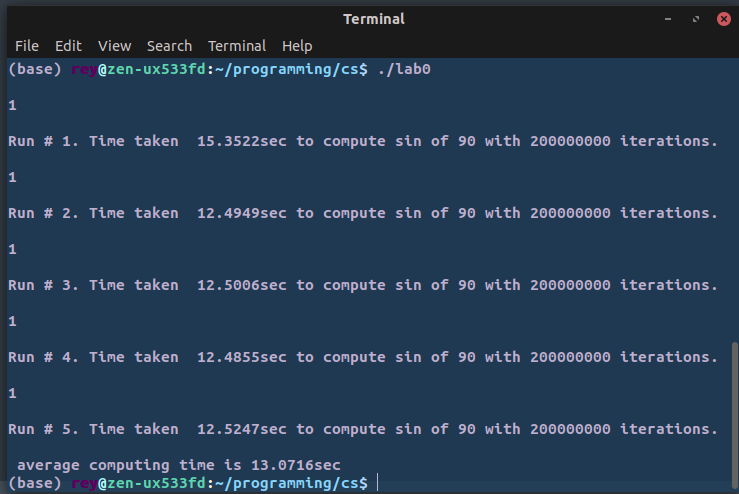
# ЦЕЛЬ

Изучение методов измерения времени работы программы и оптимизации этих измерений, а также практическое применение данных навыков.

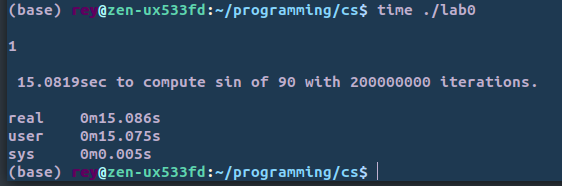
# ЗАДАНИЕ

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

# ОПИСАНИЕ РАБОТЫ

1. 1. На первом этапе был реализован алгоритм подсчёта синуса с помощью ряда Тейлора на языке программирования С++.
2. 2. Следующим шагом была выполнена проверка его работы на пяти значениях и добавлены команды замера времени из библиотеки rt (Приложение 1). Время замеряется пять раз и ищется среднее время выполнения. 

3. Произведен замер времени работы программы из командной строки с помощью команды time



**ЗАКЛЮЧЕНИЕ**

В ходе данной лабораторной работы мы познакомились с различными методами измерения работы программ и научились пользоваться ими на практике.

1. **Приложение 1:Исходный код программы для тестирования**

|  |
| --- |
| #include <ctime>  #include <iostream>  #include <cstdlib>  #include <vector>  #include <cmath>  #define pi 3.14159265  #define N 200000000  using namespace std;  double toRadians(double degrees){  return (pi / 180) \* degrees;  }  long double sinus(double x){  long double current, prev = x, sin = 0;  for (int n = 1, f = 1; n < N; n++, f+=2){  sin += prev;  current = (abs(prev) \* pow(-1, n) \* pow(x,2)) / ((f + 1) \* (f + 2));  prev = current;  }  return sin;  }  int main() {  struct timespec start, end;  double totalTime = 0;  int runs = 6;  int degrees = 90;  for (int i = 1; i < runs; i++){  clock\_gettime(CLOCK\_MONOTONIC\_RAW, &start);  cout << endl << sinus(toRadians(degrees)) << endl << endl;  cout << "Run # "<< i << ". Time taken ";  clock\_gettime(CLOCK\_MONOTONIC\_RAW, &end);  double time = end.tv\_sec-start.tv\_sec +1e-9 \* (end.tv\_nsec - start.tv\_nsec);  cout << " " << time << "sec to compute sin of "<< degrees << " with "<< N << " iterations." << endl;  totalTime += time;  }  cout << "\n average computing time is " << totalTime/(runs - 1) << "sec\n";  return 0;  } |