Министерство науки и высшего образования Российской Федерации

Федеральное государственное автономное образовательное   
учреждение высшего образования

Национальный исследовательский Нижегородский государственный университет им. Н.И. Лобачевского

Институт информационных технологий, математики и механики

**Отчет по лабораторной работе**

**«Приблизительное вычисление функций sin(x) cos(x) exp(x) ln(x) через разложение функций в ряд Маклорена »**

**Выполнил**:

студент группы 3821Б1ПМ2

Гурьев Г.А

**Проверил**:

преподаватель каф. МОСТ,

Волокитин В.Д.

Нижний Новгород

2022

**Оглавление**

[Постановка задачи 3](#__RefHeading___Toc2373_4239795057)

[Методы решения 4](#__RefHeading___Toc2375_4239795057)

[Руководство пользователя](#__RefHeading___Toc2377_4239795057) 5

[Описание программной реализации](#__RefHeading___Toc2379_4239795057) 6

[Подтверждение корректности](#__RefHeading___Toc2381_4239795057) 8

[Результаты экспериментов](#__RefHeading___Toc2383_4239795057) 9

[Заключение 2](#__RefHeading___Toc2385_4239795057)1

# Постановка задачи

Целью лабораторной работы является написание сортировки выбором, сортировки слиянием, а также Быстрой и Поразрядной сортировок на языке программирования СИ с использованием типа данных floatU+002e

# Методы решения

## Прямое суммирование

Суммирование элементов ряда маклорена с первого по n , где n зависит от точности алгоритма.

Реализуется через нахождение следующего элемента ряда через предыдущий и рекурсивную сумму этих элементов.

## Обратное суммирование

Для большей точности сначала находим n-ый элемент и начинаем ссумировать с n элемента до начала.

**Попарное суммирование**

. Ссумируются элементы сначала с первого по n/2-1, а затем с n по n/2.

# Руководство пользователя

Пользователю необходимо ввести тип функции для проверки корректности работы алгоритма. Программа выводит погрешности в формате  
%значение x% %абсолютная погрешность при прямом ссумировании% %относительная погрешность при прямом ссумировании% %абсолютная погрешность при обратномссумировании% %относительная погрешность при обратном ссумировании% %абсолютная погрешность при попарном ссумировании% %относительная погрешность при попарном ссумировании%

# Описание программной реализации

**Прямое сложение**

**функция double** sum(char func\_type, double x, double first\_element)

принимает на вход:

char func\_type – тип функции.

Double x - переменная x

double first\_element – первый элемент ряда маклорена

возвращает значение суммы элементов ряда маклорена.

**double** recursion\_sum(char func\_type,double prev\_elem, double x, int i)

char func\_type

double x

double first element

возвращает значение суммы элементов ряда маклорена со второго по n-ый

**Реверсивное сложение**

Функция double invers\_sum(char func\_type, double first\_elem, double x)

принимает на вход:

char func\_type – тип функции.

Double x - переменная x

double first\_element – первый элемент ряда маклорена

Возвращает сумму всех элементов ряда начиная с n-ого по первый

**Попарное сложение**

Функция double pairwise\_sum(char func\_type, double first\_elem, double x)

принимает на вход:

char func\_type – тип функции.

Double x - переменная x

double first\_element – первый элемент ряда маклорена

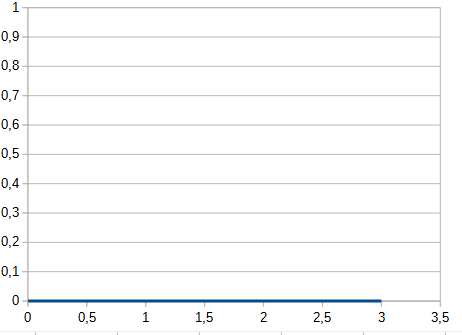
Возвращает сумму всех элементов ряда начиная с первого по n/2-1 и начиная с n-ого по n/2.

# Подтверждение корректности

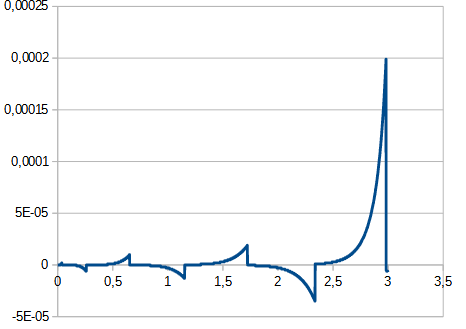
Для проверки работы алгоритма используется сравнение с соответствующими функциями из библиотеки math.h.

# Результаты экспериментов

## Нахождение синуса



абсолютная ошибка при прямом ссумировании

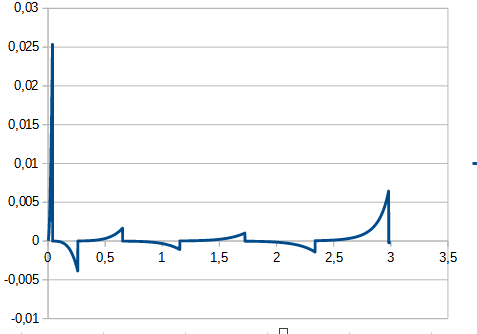


относительная погрешность при прямом ссумировании

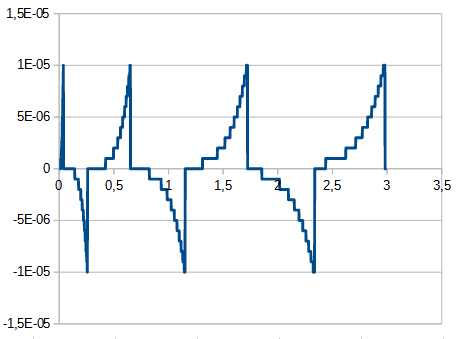
## 

## 

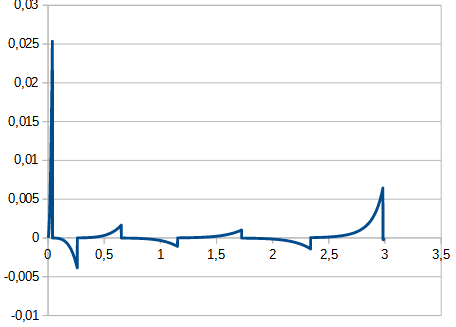
абсолютная погрешность при обратном суммироваии



относительная погрешность при обратном суммировании

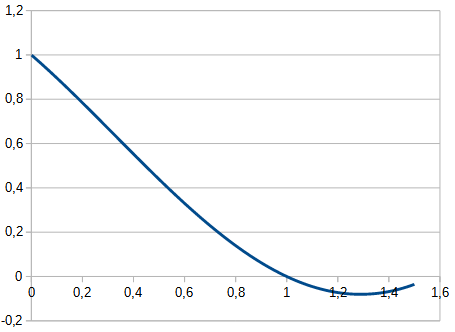


абсолютная погрешность при суммироваии попарном

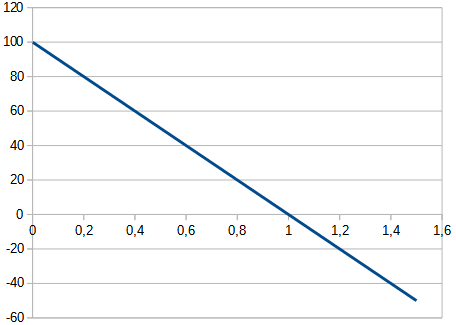


относительная погрешность при суммировании попарном

## Нахождение косинуса



абсолютная ошибка при прямом ссумировании

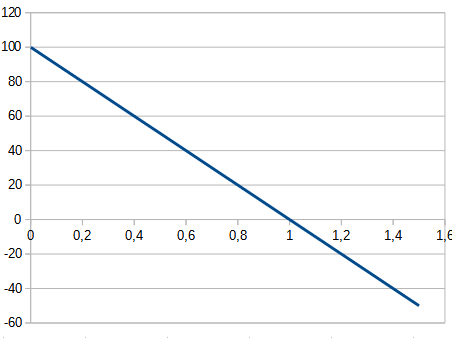


относительная погрешность при прямом ссумировании

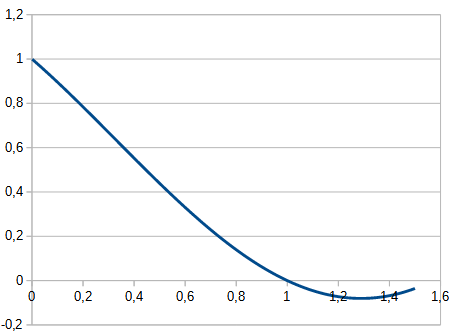
## 

## 

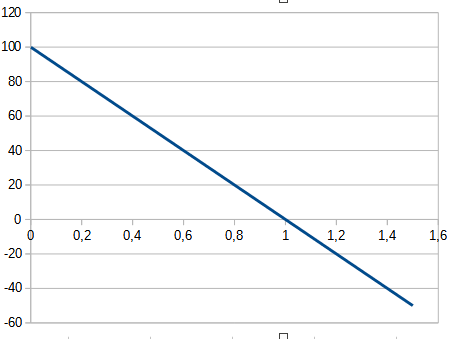
абсолютная погрешность при обратном суммироваии



относительная погрешность при обратном суммировании

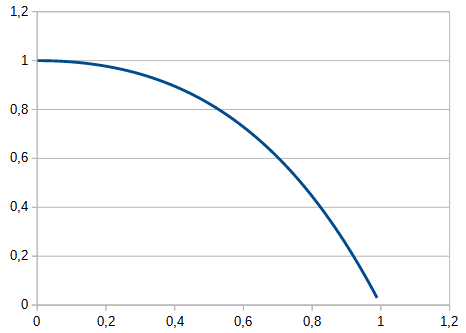


абсолютная погрешность при суммироваии попарном

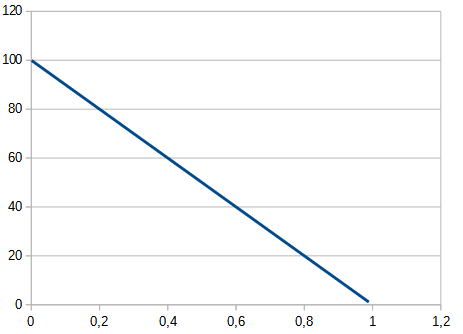


относительная погрешность при суммировании попарном

**Нахождение экспоненты**



абсолютная ошибка при прямом ссумировании

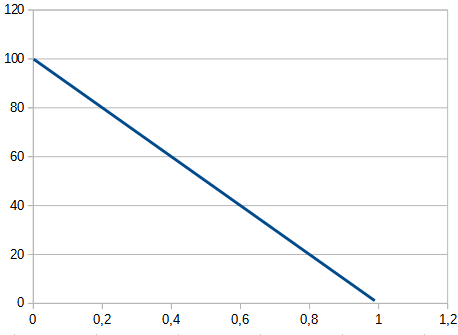


относительная погрешность при прямом ссумировании

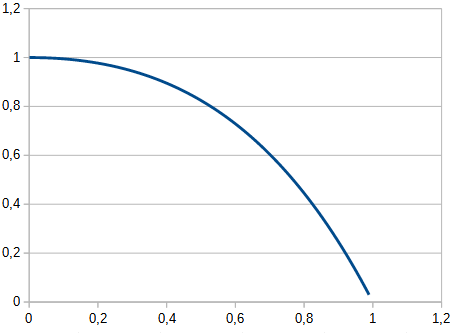
## 

## 

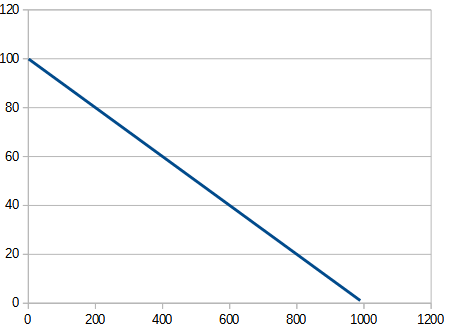
абсолютная погрешность при обратном суммироваии



относительная погрешность при обратном суммировании

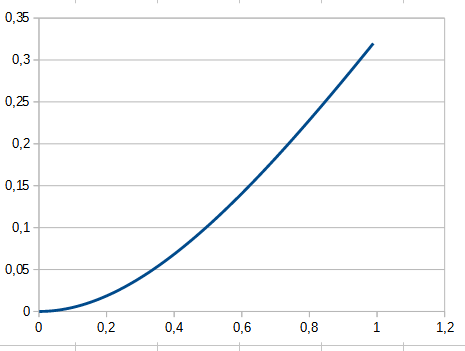


абсолютная погрешность при суммироваии попарном

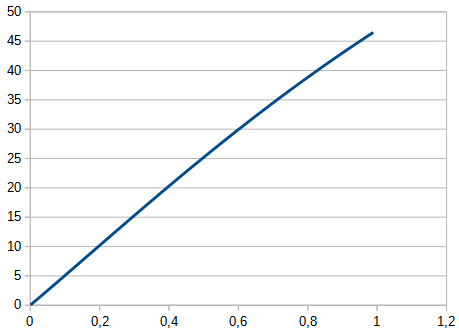


относительная погрешность при суммировании попарном

**Нахождение ln(1+x)**



абсолютная ошибка при прямом ссумировании

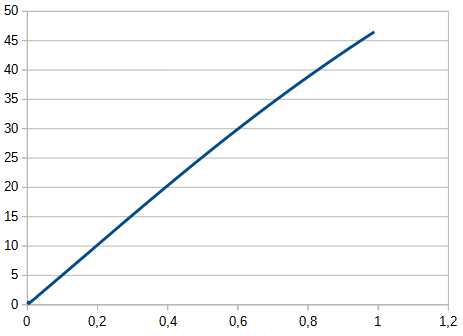


относительная погрешность при прямом ссумировании

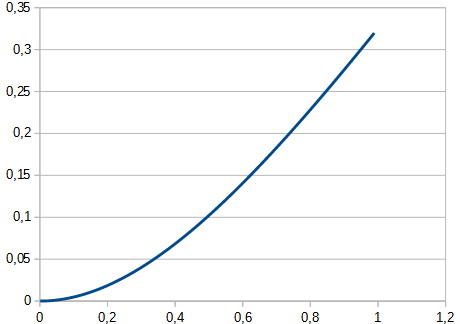
## 

## 

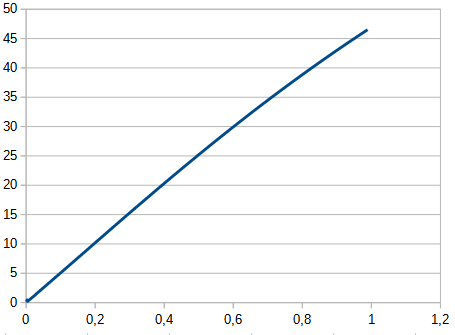
абсолютная погрешность при обратном суммироваии



относительная погрешность при обратном суммировании



абсолютная погрешность при суммироваии попарном



относительная погрешность при суммировании попарном

# Заключение

Были реализованы алгоритмы поиска sin(x) cos(x) e^x и log(x) на языке программирования C. Описаны алгоритмы вычисления приблизительного значения функции. Проведены эксперементы по измерению относительной и абсолютной погрешности. Эксперементально подтверждено, что различные методы ссумирования имеют незначительную разницу или не имеют таковой.