|  |  |
| --- | --- |
| https://lh7-us.googleusercontent.com/QuBaagabzZYLr7U3ZbYOtMusd5FbWWxIvMUTN8jrLHzSLMg534z9gXRTIG1Us4i_lOwmWlaBxKedNt-SQ26dm4WmyqwjGDmEO6z8GE3QrZosqvHM88J2EFeVf1u0GzyCZQlhWmp1Zeo85tKo4LJVXQ | МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ  ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ «МОСКОВСКИЙ АВИАЦИОННЫЙ ИНСТИТУТ (НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ)» |

Институт № 3 «Системы управления, информатика и электроэнергетика»

Кафедра 304 «Вычислительные машины, системы и сети»

**Отчет по лабораторной работе**

по учебной дисциплине «Информационные технологии»

на тему «Вычисление суммы бесконечного числового ряда»

Вариант №3

Руководители работы:

доц. каф.304, канд. физ.-мат. наук Чечиков Юрий Борисович

Секретарёв Виталий Евгеньевич

Исполнители:

обучающиеся группы М3О-119Б-24

Ефимов Ярослав Сергеевич

Тихонов Павел Сергеевич

Москва 2024

Содержание

[1. Задание 3](#_Toc180709497)

[2. схема алгоритма 4](#_Toc180709498)

[3. исходный код программного обеспечения 5](#_Toc180709499)

[4. тестирование программного обеспечения 6](#_Toc180709500)

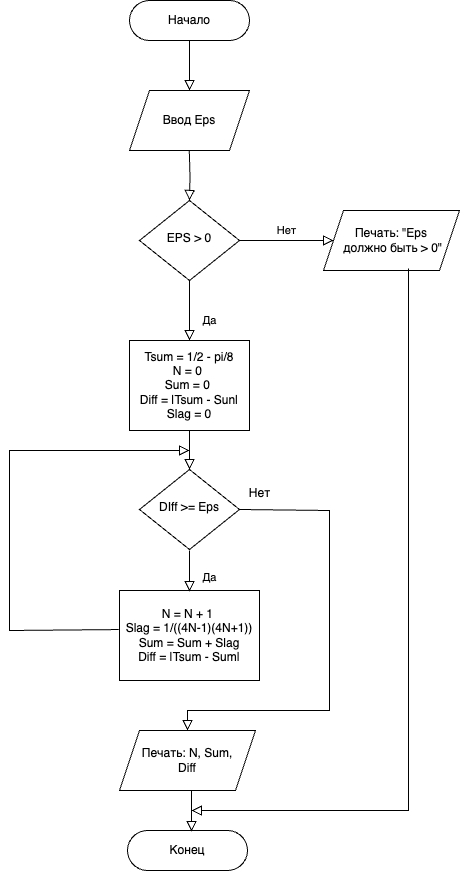
[4.1. Тесты с недопустимыми входными данными 6](#_Toc180709501)

[4.2. Тесты с допустимыми входными данными 7](#_Toc180709502)

[Вывод по работе 10](#_Toc180709503)

# Задание

# схема алгоритма



Ввод исходных данных

Входной контроль

Инициализация переменных

s

Вычисление суммы ряда

Печать результата

1

# исходный код программного обеспечения

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
\* Курс КОМПЬЮТЕРНЫЕ ТЕХНОЛОГИИ \*  
\*----------------------------------------------------------------------------------\*  
\* Project Type : Win64 Console Application \*  
\* Project Name : Labs1 \*  
\* File Name : Labs1.exe \*  
\* Language : CPP, MSVS 2022 \*  
\* Programmers : Тихонов Павел Сергеевич, Ефимов Ярослав Сергеевич, М3О-119БB-24 \*  
\* Modifyed By : \*  
\* Created : 22.10.2024 \*  
\* Last Revision : 22.10.2024 \*  
\* Comment(s) : Вычисление суммы бесконечного числового ряда \*  
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  
  
#include <cmath>  
#include <iostream>  
using namespace std;  
  
int main() // начало программы  
{  
 setlocale(LC\_ALL, "Russian"); // подключение русского языка  
  
 // ОБЪЯВЛЕНИЕ ПЕРЕМЕННЫХ  
 int N; // количество просуммированных слагаемых (степень)  
 double Tsum; // точная сумма

double Sum; // сумма  
 double Diff;// модуль разности точной суммы и суммы,

double Slag; //слагаемое,

double Eps; //погрешность  
  
 // ВВОД ИСХОДНЫХ ДАННЫХ  
 cout << "Введите погрешность: ";  
 cin >> Eps;  
  
 // ВХОДНОЙ КОНТРОЛЬ  
 if (Eps <= 0) {  
 cout << "Погрешность должна быть больше '0'"; // сообщение об ошибке  
 return 0;  
 }  
  
 // ИНИЦИАЛИЗАЦИЯ ПЕРЕМЕННЫХ  
 TSum = 0.5 - (M\_PI / 8); // ТОЧНАЯ СУММА БЕСКОНЕЧНОГО РЯДА  
 N = 0; // КОЛИЧЕСТВО СЛАГАЕМЫХ БЕСКОНЕЧНОГО РЯДА  
 Sum = 0; // CУММА БЕСКОНЕЧНОГО РЯДА   
 Diff = abs(TSum - Sum); // РАЗНОСТЬ МЕЖДУ ТОЧНОЙ СУММОЙ И СУММОЙ  
 Slag = 0; // СЛАГАЕМОЕ БЕСКОНЕЧНОГО РЯДА  
  
 // РАСЧЁТ СУММЫ БЕСКОНЕЧНОГО РЯДА  
 while (Diff >= Eps) {  
 N++;  
 Slag = 1.0 / ((4 \* N - 1) \* (4 \* N + 1));  
 Sum += Slag;  
 Diff = abs(TSum - Sum);  
 }  
  
 // ПЕЧАТЬ РЕЗУЛЬТАТОВ  
 cout << "Количество элементов суммы: " << N << ", сумма: " << Sum << ", разность: " << Diff;  
 return 0;  
} // конец программы

# тестирование программного обеспечения

## 4.1. Тесты с недопустимыми входными данными

**Тест 1**

Цель теста: проверить работоспособность программы при значении погрешности (Eps) из некорректной области

Исходные данные: Eps = -8

Ожидаемый результат: «Погрешность должна быть больше '0'»

Полученный результат:

Изображение выглядит как текст, Шрифт, снимок экрана, линия

Автоматически созданное описание

Вывод: полученный результат совпал с ожидаемым.

**Тест 2**

Цель теста: проверить работоспособность программы при значении погрешности (Eps) на границе некорректной области

Исходные данные: Eps = 0

Ожидаемый результат: «Погрешность должна быть больше '0'»

Полученный результат:

Изображение выглядит как текст, Шрифт, снимок экрана, линия

Автоматически созданное описание

Вывод: полученный результат совпал с ожидаемым.

## 4.2. Тесты с допустимыми входными данными

**Тест 1**

Цель теста: Проверить работоспособность программы при корректном Eps

Исходные данные: Eps = 6

Ожидаемый результат:

N=0

Tsum=1/2-π/8

Sum=0

Diff=|1/2-π/8-1|=0.107301

0.107301<6 (Да)

«Количество элементов суммы: 0, сумма: 0, разность: 0.107307»

Полученный результат:

Изображение выглядит как текст, Шрифт, снимок экрана, линия

Автоматически созданное описание

Вывод: полученный результат совпал с ожидаемым.

**Тест 2**

Цель теста: Проверить работоспособность программы при корректном Eps

Исходные данные: Eps = 0.2

Ожидаемый результат:

N=0

Tsum=1/2-π/8

Sum=0

Diff=|1/2-π/8-0|=0.107301

0.107301<0.2 (Да)

«Количество элементов суммы: 0, сумма: 0, разность: 0.107307»

Полученный результат:

Изображение выглядит как текст, Шрифт, снимок экрана, линия

Автоматически созданное описание

Вывод: полученный результат совпал с ожидаемым.

**Тест 3**

Цель теста: Проверить работоспособность программы при корректном Eps

Исходные данные: Eps = 0.05

Ожидаемый результат:

N=0

Tsum=1/2-π/8

Sum=0

Diff=|1/2-π/8-0|=0.107301

0.107301<0.05 (Нет)

N=1: ArgZnam=(4\*1-1)\*(4\*1+1)=3\*5, Znam=3\*5=15, Slag=1/15=0.0666667

Sum=0+0.0666667=0.0666667

Diff=|1/2-π/8-0.0666667|=0.0406343

0.0406343<0.05 (Да)

«Количество элементов суммы: 1, сумма: 0.066667, разность: 0.0406343»

Полученный результат:

Изображение выглядит как текст, Шрифт, снимок экрана

Автоматически созданное описание

Вывод: полученный результат совпал с ожидаемым.

**Тест 4**

Цель теста: Проверить работоспособность программы при корректном Eps

Исходные данные: Eps = 0.02

Ожидаемый результат:

N=0

Tsum=1/2-π/8

Sum=0

Diff=|1/2-π/8-0|=0.107301

0.107301<0.02 (Нет)

N=1: ArgZnam=(4\*1-1)\*(4\*1+1)=3\*5, Znam=3\*5=15, Slag=1/15=0.0666667

Sum=0+0.0666667=0.0666667

Diff=|1/2-π/8-0.0666667|=0.0406343

0.0406343<0.02 (Нет)

N=2: ArgZnam=(4\*2-1)\*(4\*2+1)=7\*9, Znam=7\*9=63, Slag=1/63=0.015873

Sum=0.0666667+0.015873=0.0825397

Diff=|1/2-π/8-0.0825397|=0.0247612

0.0247612<0.02 (Нет)

N=3: ArgZnam=(4\*3-1)\*(4\*3+1)=11\*13, Znam=11\*13=143, Slag=1/143=0.006993

Sum=0.0825397+0.006993=0.0895327

Diff=|1/2-π/8-0.0895327|=0.0177683

0.0177683<0.02 (Да)

«Количество элементов суммы: 3, сумма: 0.0895327, разность: 0.0177683»

Полученный результат:

Изображение выглядит как текст, Шрифт, снимок экрана

Автоматически созданное описание

Вывод: полученный результат совпал с ожидаемым.

# Вывод по работе

Разработка программы считается завершенной на следующих основаниях:

1. Полученные результаты совпали с ожидаемыми.
2. Набор тестов считается полным.