**МИНОБРНАУКИ РОССИИ**

**Санкт-Петербургский государственный**

**электротехнический университет**

**«ЛЭТИ» им. В.И. Ульянова (Ленина)**

**Кафедра** **САПР**

отчет

**по «Практическому заданию №2»**

Тема: Итерационные вычисления суммы

**с** вещественными числами

|  |  |  |
| --- | --- | --- |
| Студент гр. 4335 |  | Иванов Г.Д. |
| Руководитель |  | Калмычков В.А. |

Санкт-Петербург

2025

**Формулировка задания:**

Написание программы с использованием циклов (итерационные алгоритмы с переходным коэффициентом q)

**Дано:**

Формула для слагаемого частичной суммы:

**Найти:**

Значение частичной суммы , где:

**Контрольный пример:**

Проведем ручной расчет значения частичной суммы для первых 20 слагаемых. Расчетные данные приведены в виде таблицы:

|  |  |  |
| --- | --- | --- |
| Номер слагаемого | Значение слагаемого | Значение частичной суммы |
| 1 | -0,24300000 | -0,24300000 |
| 2 | 0,07873200 | -0,16426800 |
| 3 | -0,03644167 | -0,20070967 |
| 4 | 0,01967850 | -0,18103117 |
| 5 | -0,01159243 | -0,19262359 |
| 6 | 0,00722297 | -0,18540062 |
| 7 | -0,00468049 | -0,19008111 |
| 8 | 0,00312216 | -0,18695895 |
| 9 | -0,00212964 | -0,18908859 |
| 10 | 0,00147858 | -0,18761001 |
| 11 | -0,00104143 | -0,18865144 |
| 12 | 0,00074233 | -0,18790911 |
| 13 | -0,00053448 | -0,18844359 |
| 14 | 0,00038814 | -0,18805545 |
| 15 | -0,00028397 | -0,18833942 |
| 16 | 0,00020911 | -0,18813031 |
| 17 | -0,00015486 | -0,18828517 |
| 18 | 0,00011526 | -0,18816990 |
| 19 | -0,00008618 | -0,18825609 |
| 20 | 0,00006470 | -0,18819139 |

Таблица 1. Ручной расчет значения частной суммы

Как можно видеть из таблицы 1, при увеличении номера слагаемого, его модуль уменьшается, стремясь к 0. Для наглядности, построим график:

Рисунок 1. График зависимости модуля слагаемого частичной суммы от его номера

Из этого следует, что действительно, граничное условие имеет смысл, причем число должно быть выбрано положительным и близким к 0.

**Способ решения:**

Для упрощения организации вычислений сделаем переход от исходного выражения к выражению в виде формулы с переходным коэффициентом:

Для получения переходного коэффициента *q* требуется найти частное двух подряд идущих слагаемых и :

Подставим в это выражение нашу исходную формулу:

Таким образом, следующее слагаемое частичной суммы может быть получено умножением предыдущего слагаемого на коэффициент .

Первое слагаемое рассчитывается по исходной формуле:

**Текст программы:**

Файл *main.cpp:*

#include <iostream>

#include <iomanip>

namespace {

const char credits [] = "===========================================================\n"

"Практическая работа №2 по дисциплине \"Программирование\"\n"

"Автор: Иванов Григорий Денисович\n"

"Группа: 4335 (подгруппа 1)\n"

"Версия: 1\n"

"Период выполнения работы: 16.01.2025 - 17.01.2025\n"

"===========================================================";

const char description [] = "Описание:\n"

"В этой программе будет проведен расчёт частичной суммы\n"

"S\_n(x) = SUM (u\_1(x), u\_2(x), ... u\_n(x)), номер последнего слагаемого\n"

"определяется в соответствии со значением числа e>0.\n"

"Формула для слагаемого u\_i(x):\n"

" u\_i(x)=(-1)^i \* 2^(2i-1) \* (i-1)! \* i! \* x^(2i+1) / (2i+1)!\n"

"Упрощенная формула (формула с переходным коэффициентом):\n"

" u\_i+1(x) = (-2x^2\*i) / (2i+3) \* ui(x)\n"

"";

}

int main()

{

double x, tmp, sum = 0, ep;

unsigned int i {1};

std::cout << credits << std::endl;

std::cout << description << std::endl;

do

{

std::cout << "Введите X из диапазона (-1, 1):";

std::cin >> x;

}while(x <= -1 || x >= 1);

do

{

std::cout << "Введите e из диапазона (0, 1):";

std::cin >> ep;

}while(ep <= 0 || ep >= 1);

tmp = -x\*x\*x / 3;

std::cout << "|" << std::setw(10) << "i" //"Шаг"

<< "|" << std::setw(20) << "u\_i(x)" //"Слагаемое"

<< "|" << std::setw(20) << "s\_i(x)" //"Сумма"

<< "|" << std::endl;

for(;;i++)

{

sum += tmp;

std::cout << "|" << std::setw(10) << i

<< "|" << std::setw(20) << std::setprecision(10) << tmp

<< "|" << std::setw(20) << std::setprecision(10) << sum

<< "|" << std::endl;

tmp \*= -2 \* x \* x \* i;

tmp /= 2 \* i + 3;

if(i >= 1000 || (tmp < 0? -tmp : tmp) < ep)

break;

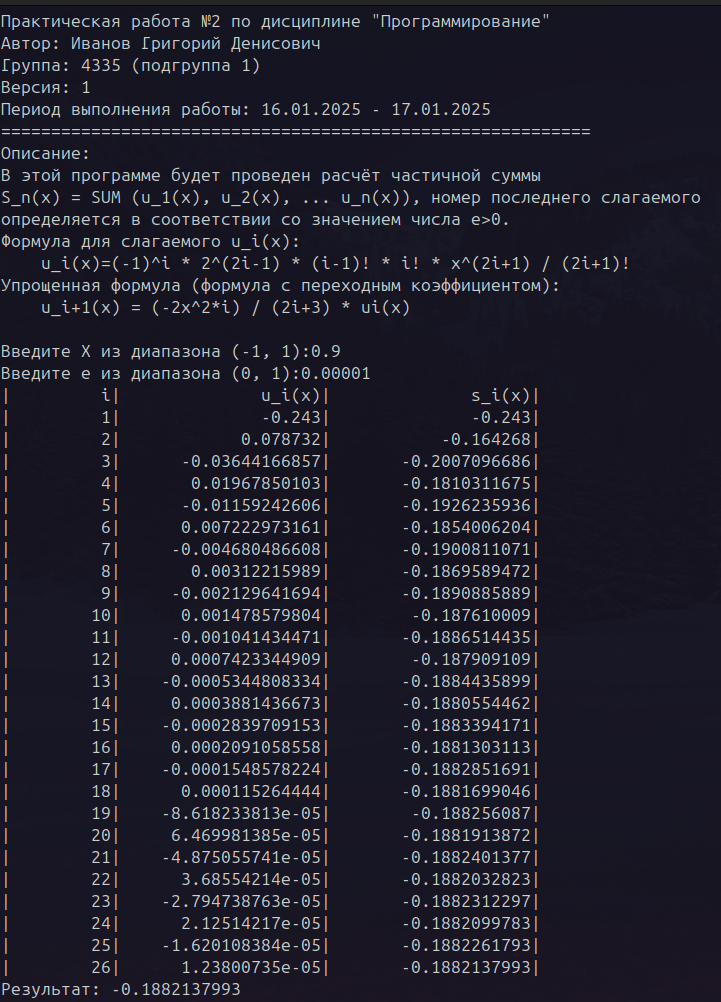
}

std::cout << "Результат: " << sum << std::endl;

return 0;

}

**Результаты работы программы:**



**Выводы:**

В этой практической работе мы освоили программирование и организацию разного вида циклов для реализации итерационного алгоритма нахождения частичной суммы математического выражения.