ФИО: Гольчанский Максим Дмитриевич

Тема: Функции, итерационные алгоритмы

Группа: 3530904/10003

Дата отправки: 20.11.2021

Постановка задачи

Разработать детальные требования и тест план для следующей задачи:

**Напишите функцию (в соответствии с указанным вариантом) для вычисления значения предложенной суммы в заданной точке (x) с указанной абсолютной погрешностью вычисления (absError) и максимальным числом слагаемых (numberMax)**

**Задача :**

1. В последовательность вводятся значения  погрешности, шага последовательности, максимального числа слагаемых:

arth(x) = x + x^3/3+x^5/5+ ….      x < 1 && x > -1

2.   Детальные требования

2.1   Последовательность должна быть задана корректно.

2.1.1 nTermsMax – целое положительное число

    2.1.1.1 Если numberMax не является целым положительным числом , выбрасываем исключение: «Enter new nTermMax».

    2.1.2 absError – вещественное число

2.1.2.1 Если погрешность не является положительным вещественным числом, выбрасываем исключение: «The calculation accuracy is set incorrectly».

2.1.3 Step — вещественное число

2.1.3.1 Если шаг интервала не является положительным вещественным числом, выбрасываем исключение: «The step value is set incorrectly».

2.1.4 Модуль term < погрешность

2.1.4.1 Если модуль значения рассматриваемого члена искомой последовательности, то выбрасываем исключение: «The required accuracy was not achieved in the entered number of steps».

2.1.5. Значения головы и хвоста последовательности ограничены -1 и 1

2.1.5.1. Если значение «начала последовательности» <= -1 или же значение «хвоста последовательности» >= 1, выбрасываем исключение : «Incorrect argument»

3. Тест-план

| Требование | Детальные требования | Данные | Ожидаемый результат |
| --- | --- | --- | --- |
| 2.1 Последовательность должна быть задана корректно   (numberMax – целое число, absError – вещественное число, Step — вещественное число , abs(temp) < absError) | | | |
| 2.1.1 nTermsMax – целое и положительное  число | 2.1.1.1 Если nTermsMax не является целым и положительным числом , выбрасываем исключение: «Введите новый nTermMax» | 5,5 0,001 0,05 | Исключение: «Введите новый  nTermMax» |
| 2.1.2  absError – вещественное число | 2.1.2.1 Если погрешность не является положительным вещественным числом, выбрасываем исключение: «Точность расчета установлена неправильно». | 5,5  -1 0,05 | Исключение: «Точность расчета установлена неправильно». |
|  | | | |
| 2.1.3 Step — вещественное число | 2.1.3.1 Если шаг интервала не является положительным вещественным числом, выбрасываем исключение: «Значение шага неверно» | 5,5 0,001 z | Исключение: «Значение шага неверно» |
| 2.1.4 abs( temp) < погрешность | 2.1.4.1 Если модуль значения рассматриваемого члена искомой последовательности, то выбрасываем исключение: «Требуемая точность не была достигнута в введенном количестве шагов» | x2n–12n−1 <absError | Исключение: «Требуемая точность не была достигнута в введенном количестве шагов». |
| 2.1.5 start <= -1 || end >= 1 | 2.1.5.1. Если значение «начала последовательности» <= -1 или же значение «хвоста последовательности» >= 1, выбрасываем исключение : «Неверный аргумент» | -1 2 | Исключение: «Неверный аргумент» |

4.Реализация

// Function.h

#ifndef FUNCTION\_H

#define FUNCTION\_H

namespace Funсtion {

double ATANH(double x, int nTermsMax, double absError);

}

#endif

// Function.cpp

#include <iostream>

#include <stdexcept>

#include <cmath>

#include "Function.h"

double Function::ATANH(double x, int nTermsMax, double absError)

{

int i = 1;

double b = 1.0;

double sum = 0.0;

double term = x;

double powerX = x;

try {

while (i <= nTermsMax && std::abs(term) > absError) {

sum += term;

b += ((2.0 \* i) + 1);

powerX \*= x \* x;

term = powerX / b;

++i;

}

if (i > nTermsMax && std::abs(term) > absError) {

throw std::logic\_error(“Требуемая точность не достигается для количества” “шагов”);

}

}

catch (const std::logic\_error& k) {

std::cerr << "Error" << k.what() << '\n';

}

return sum;

}

//main.cpp

#include <iostream>

#include <iomanip>

#include <cmath>

#include <stdexcept>

#include "Function.h"

int main()

setlocale(LC\_ALL, "Russian");

double absError = 0;

int nTermsMax = 0;

double start = 0;

double end = 0;

double step = 0;

try {

std::cout << "Введите начало последовательности : ";

std::cin >> start;

std::cout << "Введите конец последовательности : ";

std::cin >> end;

if (start <= -1.0 || end >= 1.0)

{

throw std::logic\_error("Неверный аргумент");

}

std::cout << "Enter the error: ";

std::cin >> absError;

if (absError < 0.0)

{

throw std::logic\_error("Точность расчета установлена неправильно");

}

std::cout << "Введите число элементов последовательности : ";

std::cin >> nTermsMax;

if (nTermsMax <= 0)

{

throw std::logic\_error("Неверно задано максимальное количество терминов");

}

std::cout << "Введите шаг последовательности: ";

std::cin >> step;

if (!std::cin) {

std::cerr << "Ошибка: неверный ввод\n";

return 1;

}

for (double x = start; x <= end; x += step)

{

std::cout << std::setprecision(6) << std::left <<

std::setw(15) << Function::ATANH(x, nTermsMax, absError) <<

std::setw(15) << std::atanh(x) << "\n";

}

}

catch (const std::logic\_error& e)

{

std::cerr << "Ошибка: " << e.what() << "\n";

}

return 0;

}