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»

Информатика

Отчет по лабораторной работе № 1

Вариант № 7

Выполнила бригада группы М3О-111Бк-21

Багиров Э. Р.

Нуриев Н. Н.

Проверил Секретарев Виталий Евгеньевич

Москва 2021 г.

Оглавление

[**Задание** 3](#_Toc86736331)

[**Диаграмма алгоритма** 4](#_Toc86736332)

[**Алгоритмический вид программы** 5](#_Toc86736333)

[**Исходный код программы** 6](#_Toc86736334)

[**Некорректные тесты** 7](#_Toc86736335)

[**Корректные тесты** 8](#_Toc86736336)

[**Протокол расчета ожидаемого результата** 11](#_Toc86736337)

[**Вывод** 12](#_Toc86736338)

# **Задание**

Кафедра: 304 Курс: ИНФОРМАТИКА

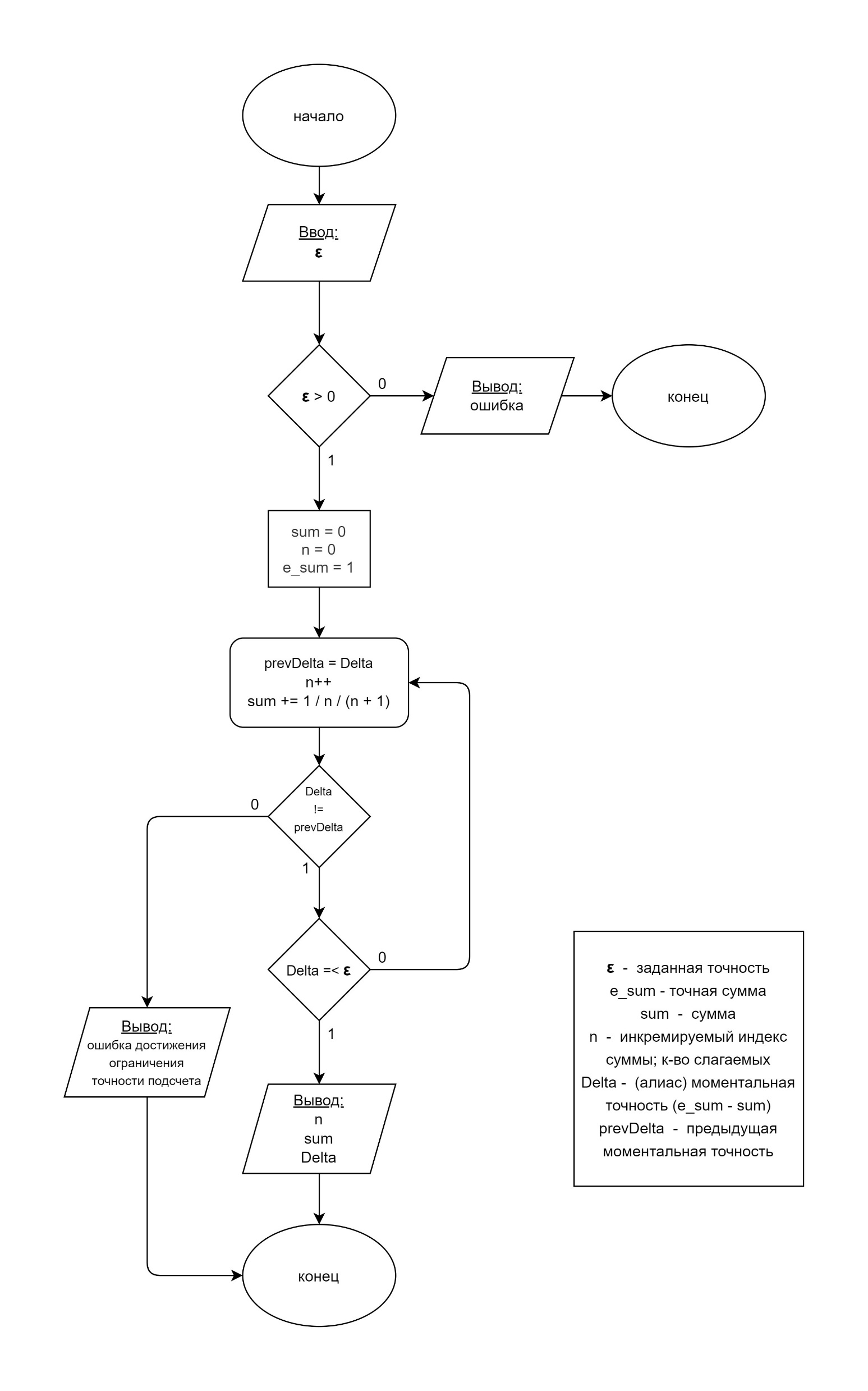
Задание 1: Вычисление суммы бесконечного числового ряда

ВАРИАНТ № 7

Определить с заданной точностью сумму бесконечного степенного ряда:

Предусмотреть ввод точности и печать: количества просуммированных элементов, суммы, разности суммы и точного значения, которое равно: **1**

# **Диаграмма алгоритма**



# **Алгоритмический вид программы**

АЛГ Вариант №7

ПЕР N, Sum, Eps, interrupted, prevDelta

НАЧАЛО

// Инициализация переменных

Eps // заданная пользователем точность подсчета Эпсилон

Sum = 0 // сумма

N = 0 // индекс суммы

prevDelta = 1 // моментальная точность предыдущей итерации

interrupted = false // флаг о выходе из цикла до достижения необходимой точности

e\_sum = 1 // точная сумма

// Входной контроль

ВВОД Eps

ЕСЛИ Eps <= 0

// Предотвращение некорректного ввода точности подсчета

ТО ПЕЧАТЬ (сообщение об ошибке)

КОНЕЦ

КОНЕСЛИ

// Расчет суммы

НЦ ПОВТОРЯТЬ

N = N + 1 // инкремент индекса

Sum += 1.0 / N / (N + 1) // подсчет нового слагаемого, прибавление его к сумме

ЕСЛИ (prevDelta + Sum - e\_sum == 0)

// Если точность не увеличилась за последнюю итерацию,

// достигнут предел точности, необходимо завершить программу

ТО

ПЕЧАТЬ ошибка достижения ограничения точности

КОНЕЦ

КОНЕСЛИ

prevDelta = e\_sum - Sum // сохранение новой моментальной точности

КЦ ПОКА (1 - Sum) > Eps

ПЕЧАТЬ (N, Sum, e\_sum - Sum)

КОНЕЦ

# **Исходный код программы**

#include <iostream>

#include <iomanip>

using namespace std;

int main()

{

setlocale(LC\_ALL, "Russian");

// Инициализация переменных

long double Eps; // заданная пользователем точность подсчета Эпсилон

double Sum = 0; // сумма

unsigned long long N = 0; // индекс суммы

double prevDelta = 1; // моментальная точность предыдущей итерации

const char e\_sum = 1; // точная сумма

// Входной контроль

cout << "Введите Eps: ";

cin >> Eps;

cout << "\n";

if (Eps <= 0) {

// Предотвращение некорректного ввода точности подсчета

cout << "ОШИБКА - Eps должен быть больше 0." << endl;

return 0;

}

// Расчет суммы

do {

N++; // инкремент индекса

Sum += 1.0 / N / (N+1); // подсчет нового слагаемого, прибавление его к сумме

if (prevDelta + Sum - e\_sum == 0) {

// Если точность не увеличилась за последнюю итерацию,

// достигнут предел точности, необходимо завершить цикл

cout << "ОШИБКА - Достигнуто ограничение по точности подсчета.\n";

return 0;

}

prevDelta = e\_sum - Sum; // сохранение новой моментальной точности

} while (prevDelta > Eps);

// Вывод результатов

cout << "Cумма получена\nКоличество просумированных элементов N = " << N << endl;

cout << "Сумма Sum = " << Sum << endl;

cout << "Достигнутая точность Delta = " << (e\_sum - Sum) << endl;

return 1;

}

# **Некорректные тесты**

**Тест №1**

Цель теста: проверить работу программы на границе некорректной области исходных данных

Исходные данные: Eps = 0

Ожидаемый результат: вывод сообщения об ошибке: “ ОШИБКА - Eps должен быть больше 0.”

Полученный результат:

Изображение выглядит как текст

Автоматически созданное описание

Вывод: полученный результат совпал с ожидаемым. Тест ошибки не обнаружил.

**Тест №2**

Цель теста: проверить работу программы в некорректной области исходных данных

Исходные данные: Eps = -0.1

Ожидаемый результат: вывод сообщения об ошибке: “ ОШИБКА - Eps должен быть больше 0.”

Полученный результат:

Изображение выглядит как текст

Автоматически созданное описание

Вывод: полученный результат совпал с ожидаемым. Тест ошибки не обнаружил.

# **Корректные тесты**

**Тест №1**

Цель теста: проверить работу программы в корректной области

Исходные данные: Eps = 0.5

Ожидаемый результат:

Расчет ожидаемой суммы:

|  |  |  |  |
| --- | --- | --- | --- |
| N | Формула | Sum | Delta (Eps) |
| 1 |  | .5 | .5 |

Вывод сообщения:

Сумма получена

Количество просуммированных элементов N = 1

Сумма Sum = 0.5

Достигнутая точность Delta = 0.5

Полученный результат:

Изображение выглядит как текст

Автоматически созданное описание

Вывод: полученный результат совпал с ожидаемым. Тест ошибки не обнаружил.

**Тест №2**

Цель теста: проверить работу программы в корректной области

Исходные данные: Eps = 0.3

Ожидаемый результат:

Расчет ожидаемой суммы:

|  |  |  |  |
| --- | --- | --- | --- |
| N | Формула | Sum | Delta (Eps) |
| 1 |  | .5 | .5 |
| 2 |  | .(6) | .(3) |
| 3 |  | .75 | .25 |

Вывод сообщения:

Сумма получена

Количество просуммированных элементов N = 3

Сумма Sum = 0.75

Достигнутая точность Delta = 0.25

Полученный результат:

Изображение выглядит как текст

Автоматически созданное описание

Вывод: полученный результат совпал с ожидаемым. Тест ошибки не обнаружил.

**Тест №3**

Цель теста: проверить работу программы в корректной области

Исходные данные: Eps = 0.2

Ожидаемый результат:

Расчет ожидаемой суммы:

|  |  |  |  |
| --- | --- | --- | --- |
| N | Формула | Sum | Delta (Eps) |
| 1 |  | .5 | .5 |
| 2 |  | .(6) | .(3) |
| 3 |  | .75 | .25 |
| 4 |  | .8 | .2 |

Вывод сообщения:

Сумма получена

Количество просуммированных элементов N = 4

Сумма Sum = 0.8

Достигнутая точность Delta = 0.2

Полученный результат:

Изображение выглядит как текст

Автоматически созданное описание

Вывод: полученный результат совпал с ожидаемым. Тест ошибки не обнаружил.

**Тест №4**

Цель теста: проверить работу программы в корректной области

Исходные данные: Eps = 0.15

Ожидаемый результат:

Расчет ожидаемой суммы:

|  |  |  |  |
| --- | --- | --- | --- |
| N | Формула | Sum | Delta (Eps) |
| 1 |  | .5 | .5 |
| 2 |  | .(6) | .(3) |
| 3 |  | .75 | .25 |
| 4 |  | .8 | .2 |
| 5 |  | .8(3) | .1(6) |
| 6 |  | .(857142) | .(142857) |

Вывод сообщения:

Сумма получена

Количество просуммированных элементов N = 6

Сумма Sum = 0.857143

Достигнутая точность Delta = 0. 142857

Полученный результат:

Изображение выглядит как текст

Автоматически созданное описание

Вывод: полученный результат совпал с ожидаемым. Тест ошибки не обнаружил.

**Тест №5**

Цель теста: проверить работу программы в корректной области

Исходные данные: Eps = 0.1

Ожидаемый результат:

Расчет ожидаемой суммы:

|  |  |  |  |
| --- | --- | --- | --- |
| N | Формула | Sum | Delta (Eps) |
| 1 |  | .5 | .5 |
| 2 |  | .(6) | .(3) |
| 3 |  | .75 | .25 |
| 4 |  | .8 | .2 |
| 5 |  | .8(3) | .1(6) |
| 6 |  | .(857142) | .(142857) |
| 7 |  | .875 | .125 |
| 8 |  | .(8) | .(1) |
| 9 |  | .9 | .1 |

Вывод сообщения:

Сумма получена

Количество просуммированных элементов N = 9

Сумма Sum = 0.9

Достигнутая точность Delta = 0.1

Полученный результат:

Изображение выглядит как текст

Автоматически созданное описание

Вывод: полученный результат совпал с ожидаемым. Тест ошибки не обнаружил.

**Тест №6**

Цель теста: проверить работу программы в корректной области

Исходные данные: Eps = 0.00000001

Ожидаемый результат: вывод сообщения об ошибке:

ОШИБКА - Достигнуто ограничение по точности подсчета.

Полученный результат:

Изображение выглядит как текст

Автоматически созданное описание

Вывод: полученный результат совпал с ожидаемым. Тест ошибки не обнаружил.

# **Протокол расчета ожидаемого результата**

|  |  |  |  |
| --- | --- | --- | --- |
| N | Формула | Sum | Delta (Eps) |
| 1 |  | .5 | .5 |
| 2 |  | .(6) | .(3) |
| 3 |  | .75 | .25 |
| 4 |  | .8 | .2 |
| 5 |  | .8(3) | .1(6) |
| 6 |  | .(857142) | .(142857) |
| 7 |  | .875 | .125 |
| 8 |  | .(8) | .(1) |
| 9 |  | .9 | .1 |
| 10 |  | .(90) | .(09) |

# **Вывод**

Проанализировали алгоритм по вычислению суммы бесконечного числового ряда. В ходе выполнения лабораторной работы были составлены диаграмма алгоритма и алгоритмический текст программы для дальнейшего написания программы. Проведены некорректные тесты для проверки соответствия исходных данных и корректные тесты, которые показали правильность выполнения алгоритма.