

Московский Авиационный Институт

Институт №3
Программная инженерия
Кафедра 304

**Отчёт по лабораторной работе
по учебной дисциплине "Информационные технологии"
на тему
"Вычисление суммы бесконечного ряда"**

Выполнили: студенты группы МЗО-111Б-22

Яковченко Н.Р.

Деккер Сергей, Сергей, Серёга, извини

Москва, 2022 г.

Содержание

1	Задание	2
2	Схема алгоритма	3
3	Текст программы	4
4	Тесты	5
4.1	Некорректные тесты	5
4.2	Корректные тесты	5

1. Задание

Кафедра 304

Курс: ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ

Задание: **Вычисление суммы бесконечного числового ряда**

Вариант №11

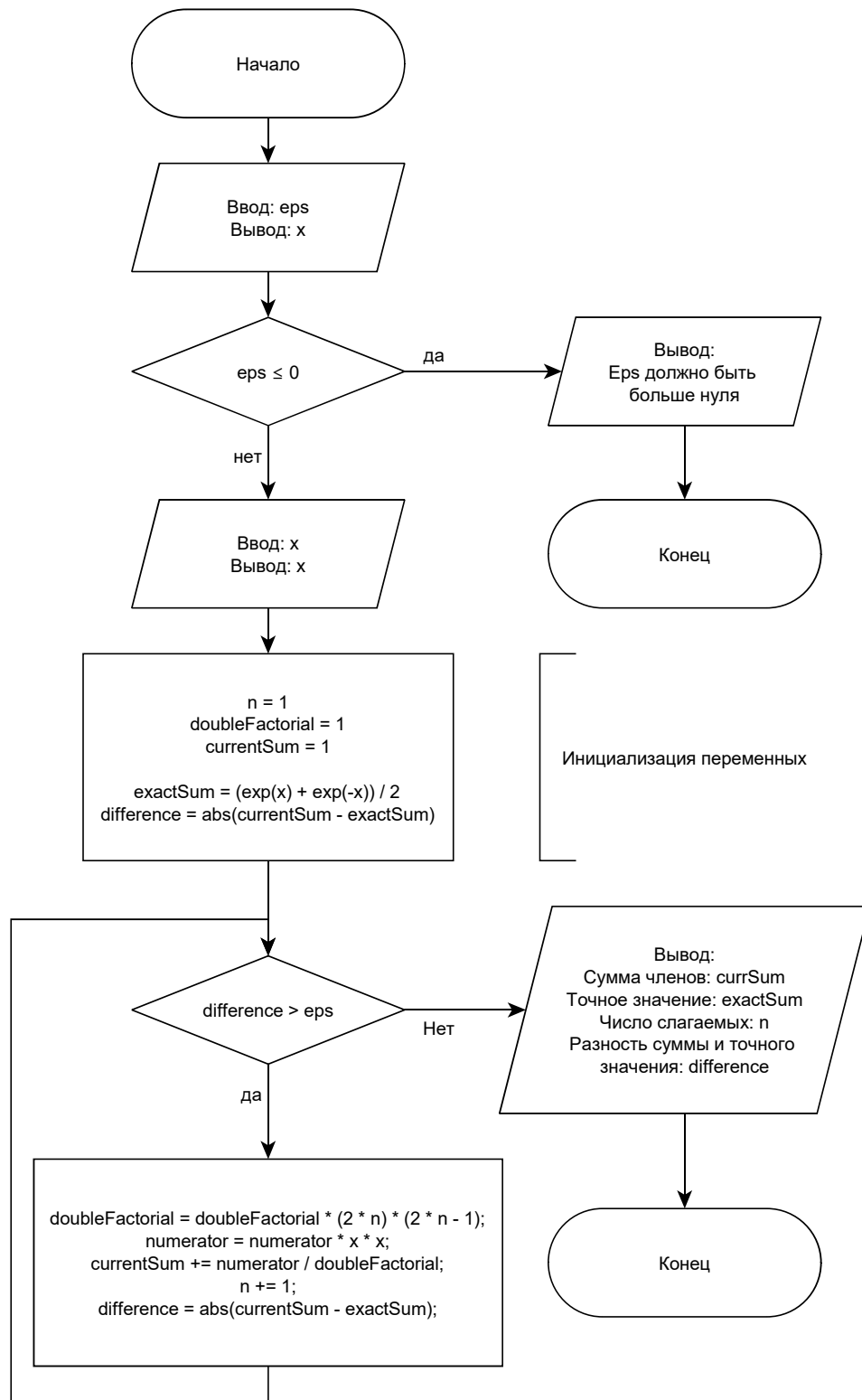
Определить с заданной точностью сумму членов бесконечного степенного ряда:

$$1 + \sum_{n=1} \frac{x^{2n}}{(2n)!} = 1 + \frac{x^2}{2!} + \frac{x^4}{4!} + \frac{x^6}{6!} + \dots$$

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

$$\frac{e^x + e^{-x}}{2} = ch(x)$$

2. Схема алгоритма



3. Текст программы

```
// Подключение библиотек
#include <iostream>
#include <cmath>
using namespace std;

int main() // Начало программы
{
    setlocale(LC_ALL, "rus"); // Подключение русского языка

    int n; // Число слагаемых
    unsigned long long doubleFactorial; // Текущее значение удвоенного факториала
    double difference; // Разность текущей и точной сумм
    double currentSum; // Сумма
    double exactSum; // Точное значение суммы
    double x; // Переменная
    double eps; // Точность
    double numerator; // Числитель

    cout << "Введите точность: ";
    cin >> eps; // Ввод точности
    cout << eps << endl; // Эхо-печать

    if (eps <= 0) // Валидация входящих данных
    {
        // Вывод сообщения об ошибке
        cout << "Заданная точность должна быть больше нуля" << endl;
        // Завершение работы программы в случае некорректности введенных данных
        return 1;
    }

    cout << "Введите X: ";
    cin >> x; // Ввод переменной x
    cout << x << endl; // Эхо-печать

    // Инициализация переменных
    n = 1;
    doubleFactorial = 1;
    currentSum = 1;
    numerator = 1;

    exactSum = (exp(x) + exp(-x)) / 2; // Подсчет точной суммы
    difference = abs(currentSum - exactSum); // Подсчет разности

    while (difference > eps) // Начало цикла
    {
        // Вычисление суммы ряда
        doubleFactorial = doubleFactorial * (2 * n) * (2 * n - 1);
        numerator = numerator * x * x;
        currentSum += numerator / doubleFactorial;
        n += 1;
        difference = abs(currentSum - exactSum);
    } // Конец цикла

    // Вывод значений переменных
    cout << "Сумма членов: " << currentSum << endl;
    cout << "Точное значение: " << exactSum << endl;
    cout << "Число слагаемых: " << n << endl;
    cout << "Разность суммы и точного значения: " << difference << endl;
    return 0; // Возврат значения 0
}
```

4. Тесты

4.1. Некорректные тесты

Тест 1 Цель: проверить работу программы на границе некорректной области
Исходные данные: $\epsilon_{ps} = 0$
Ожидаемый результат: Заданная точность должна быть больше нуля
Полученный результат:

```
..sktop/lab/bin x + v
DESKTOP-P829JJ1:bin zero$ ./calculate_series (master)
Введите точность: 0
0
Заданная точность должна быть больше нуля
```

Вывод по тесту: Полученный результат совпал с ожидаемым. Тест ошибок не выявил.

Тест 2 Цель: проверить работу программы на границе некорректной области
Исходные данные: $\epsilon_{ps} = -1.02$
Ожидаемый результат: Заданная точность должна быть больше нуля
Полученный результат:

```
..sktop/lab/bin x + v
DESKTOP-P829JJ1:bin zero$ ./calculate_series (master*)
Введите точность: -1.02
-1.02
Заданная точность должна быть больше нуля
```

Вывод по тесту: Полученный результат совпал с ожидаемым. Тест ошибок не выявил.

4.2. Корректные тесты

Тест 1 Цель: проверить работу цикла по расчёту суммы ряда при высокой погрешности
Исходные данные: $\epsilon_{ps} = 100$, $x = 1$

Ожидаемый результат: Сумма членов: 1
Точное значение: 1,5308
Число слагаемых: 1
Разность суммы и точного значения: 0,5308

Полученный результат:

```
..sktop/lab/bin x + v
DESKTOP-P829JJ1:bin zero$ ./calculate_series (master*)
Введите точность: 100
100
Введите X: 1
1
Сумма членов: 1
Точное значение: 1.54308
Число слагаемых: 1
Разность суммы и точного значения: 0.543081
```

Вывод по тесту: Полученный результат совпал с ожидаемым. Тест ошибок не выявил.

Тест 2 Цель: проверить работу цикла по расчёту суммы ряда
Исходные данные: $\text{eps} = 0.001$, $x = 3$

Ожидаемый результат: Сумма членов: 10,0676
Точное значение: 10,0677
Число слагаемых: 7
Разность суммы и точного значения: $\approx 0,00006$

n	currentSum	exactSum	difference
1	1		9,06766
2	5,5		4,56766
3	8,875		1,19266
4	9,8875	10,06766	0,18016
5	10,05022		0,01744
6	10,0665		0,00116
7	10,0676		0,00006

Полученный результат:

```
..sktop/lab/bin
DESKTOP-P829JJ1:bin zero$ ./calculate_series
Введите точность: 0.001
0.001
Введите X: 3
3
Сумма членов: 10.0676
Точное значение: 10.0677
Число слагаемых: 7
Разность суммы и точного значения: 5.69836e-05
(master*)
```

Вывод по тесту: Полученный результат совпал с ожидаемым. Тест ошибок не выявил.

Тест 3 Цель: проверить работу цикла по расчёту суммы ряда
Исходные данные: $\text{eps} = 50$, $x = 10$

Ожидаемый результат: Сумма членов: 11002,4
Точное значение: 11013.2
Число слагаемых: 11
Разность суммы и точного значения: $\approx 10,8$

n	currentSum	exactSum	difference
1	1	11013.2	11012.2
2	51		10962,2
3	467,6		10545,6
4	1856,5		9156,7
5	4436,7		6576,5
6	7092,4		3920,8
7	9180,1		1833,1
8	10327,19		686,01
9	10805,1		208,1
10	10961,3		51,9
11	11002,4		10,8

Полученный результат:

```
..sktop/lab/bin
DESKTOP-P829JJ1:bin zero$ ./calculate_series
Введите точность: 50
50
Введите X: 10
10
Сумма членов: 11002.4
Точное значение: 11013.2
Число слагаемых: 11
Разность суммы и точного значения: 10.7935
(master*)
```

Вывод по тесту: Полученный результат совпал с ожидаемым. Тест ошибок не выявил.