

Правительство Российской Федерации

**Федеральное государственное автономное образовательное учреждение
высшего профессионального образования
"Национальный исследовательский университет
"Высшая школа экономики"**

Московский институт электроники и математики Национального
исследовательского университета "Высшая школа экономики"

Департамент прикладной математики

ОТЧЕТ

По лабораторной работе №3

**ВЫЧИСЛЕНИЯ ПО РЕКУРРЕНТНЫМ ФОРМУЛАМ С ПОМОЩЬЮ ОПЕРАТОРОВ
ЦИКЛА**

По курсу «Алгоритмизация и программирование»

ФИО студента	Номер группы	Дата
Андреев Евгений Игоревич	БПМ-191	

Москва – 2019 г.

ЗАДАНИЕ (вариант №1)

Вычислить приближенное значение функции, вычислив сумму конечного числа элементов ряда двумя способами, используя разные типы циклов:

- 1) с заданной точностью;
- 2) для заданного количества членов ряда.

Переход к способу вычисления реализовать с помощью оператора выбора.

$$\operatorname{sh} x = x + \frac{x^3}{3!} + \frac{x^5}{5!} + \dots + \frac{x^{2m-1}}{(2m-1)!} + \dots, \quad x \in \mathbb{R},$$

РЕШЕНИЕ

```
#include <stdio.h>
#include <math.h>
#include <stdlib.h>

/* Введём функцию для вычисления факториала: */
long factorial(long a)
{
    long a1, a2;
    a1 = 1;
    a2 = 1;
    while(a >= a1)
    {
        a2 = a2 * a1;
        a1 = a1 + 1;
    }
    return a2;
}

int main( )
{
    int m, w, i;                /* введение целых переменных */
    float piece;                /* введение вещественной переменной */
    double x, result1 = 0, result2 = 0, delta, t; /* введение вещественных переменных двойной точности */

    printf("Введите цифру от 1 до 2-х, которая будет соответствовать номеру способа решения: ");
    /* вывод информации */
    if (scanf("%d", &w) == 0 || getchar() != '\n' || (w < 1) || (w > 2)) {
        printf("Ошибка! Введите цифру от 1 до 2-х.");
        return 0;
    }
    /* вывод ошибки при введении неверных символов */

    /* Способ 1 ( с заданной точностью ) : */

    if (w == 1){
        printf("Введите число, задающее точность вычисления: ");
        if (scanf("%lf", &t) == 0 || getchar() != '\n') {
            printf("Ошибка! Значение должно быть числом");
        }
    }
}
```

```

        return 0;
    }
    /* ввод целого числа типа int со знаком в десятичной системе счисления и
       вывод ошибки при введении неверных символов в числовые переменные */

    printf("Введите x: ");
    if (scanf("%lf", &x) == 0 || getchar() != '\n') {
        printf("Ошибка! Значение должно быть числом");
        return 0;
    }

    i = 1;
    delta = (powf(x, ((2 * (i + 1)) - 1)) / factorial((2 * (i + 1)) - 1) - powf(x, ((2 * i) - 1)) / factorial((2 * i) -
1));
    piece = powf(x, ((2 * i) - 1)) / factorial((2 * i) - 1);

    while (abs(delta) >= t){        // условие, что разность членов ряда будет не меньше заданной
точности
        result1 = result1 + piece;
        i = i + 1;
        piece = powf(x, ((2 * i) - 1)) / factorial((2 * i) - 1);
        delta = (powf(x, ((2 * (i + 1)) - 1)) / factorial((2 * (i + 1)) - 1) - powf(x, ((2 * i) - 1)) / factorial((2 * i) -
1));
    }
    printf("\nПриближенное значение функции: %f", result1); // вывод результата
    /* вычисление приближённого значения функции
       с заданной точностью с помощью цикла while */
}

/* Способ 2 ( для заданного количества членов ряда ) : */

else if (w == 2){
    printf("Введите количество членов ряда(> 0): ");
    if (scanf("%d", &m) == 0 || getchar() != '\n' || (m <= 0)) {
        printf("Ошибка! Значение должно быть целым положительным числом.");
        return 0;
    }
    /* ввод целого числа типа int со знаком в десятичной системе счисления и
       вывод ошибки при введении неверных символов в числовые переменные */

    printf("Введите x: ");
    if (scanf("%lf", &x) == 0 || getchar() != '\n') {
        printf("Ошибка! Значение должно быть числом");
        return 0;
    }

    for (int j = 1; j <= m; j++){
        result2 = result2 + powf(x, ((2 * j) - 1)) / factorial((2 * j) - 1);
    }
    /* вычисление приближённого значения функции с заданным
       количеством членов ряда с помощью цикла for */
    printf("\nПриближенное значение функции: %f", result2); // вывод результата
}
return 0;
}

```

ТЕСТЫ

Тест № 1

Способ 1:

```
Введите цифру от 1 до 2-х, которая будет соответствовать номеру способа решения: 1
Введите число, задающее точность вычисления: 0.2
Введите x: 10

Приближенное значение функции: 10989.042244
```

Способ 2:

```
Введите цифру от 1 до 2-х, которая будет соответствовать номеру способа решения: 2
Введите количество членов ряда(> 0): 10
Введите x: 10

Приближенное значение функции: 10989.042244
```

Проверка в WolframAlpha:

Input:

$\sinh(10)$

Decimal approximation:

11013.23287470339337723652455484636440290145119031934610383...

Этот тест показывает, что оба способа исправны.

Тест № 2

Способ 1:

```
Введите цифру от 1 до 2-х, которая будет соответствовать номеру способа решения: 1
Введите число, задающее точность вычисления: 0.5
Введите x: 5

Приближенное значение функции: 73.981528
```

Введите цифру от 1 до 2-х, которая будет соответствовать номеру способа решения: 1
Введите число, задающее точность вычисления: 0.0000004
Введите x: 5
Приближенное значение функции: 74.203200

Проверка в WolframAlpha:

Input:

$\sinh(5)$

Decimal approximation:

74.20321057778875897700947199606456559961940900442581698066...

Этот тест показывает, что при разной заданной точности Способ 1 выдаёт разные значения. Таким образом, во втором случае значение оказалось более точным, чем в первом.

Тест № 3

Способ 2:

Введите цифру от 1 до 2-х, которая будет соответствовать номеру способа решения: 2
Введите количество членов ряда(> 0): 6
Введите x: 6
Приближенное значение функции: 199.203121

Введите цифру от 1 до 2-х, которая будет соответствовать номеру способа решения: 2
Введите количество членов ряда(> 0): 10
Введите x: 6
Приближенное значение функции: 201.712700

Проверка в WolframAlpha:

Input:

$\sinh(6)$

Decimal approximation:

201.7131573702792281249820676879787314690041954387718343344...

Этот тест показывает, что при разных заданных значениях количества членов ряда Способ 2 выдаёт разные значения. Таким образом, во втором случае значение оказалось более точным, чем в первом.

Тест № 4

Способ 1:

Введите цифру от 1 до 2-х, которая будет соответствовать номеру способа решения: 1
Введите число, задающее точность вычисления: 0.001
Введите x : -5
Приближенное значение функции: -73.981528

Способ 2:

Введите цифру от 1 до 2-х, которая будет соответствовать номеру способа решения: 2
Введите количество членов ряда(> 0): 10
Введите x : -5
Приближенное значение функции: -74.203200

Проверка в WolframAlpha:

Input:

$\sinh(5)$

Decimal approximation:

74.20321057778875897700947199606456559961940900442581698066...

Данные тест показывает, что программа работает и с отрицательными значениями x .