

**Міністерство освіти і науки України  
Національний технічний університет України  
«Київський політехнічний інститут імені Ігоря Сікорського»  
Факультет інформатики та обчислювальної техніки  
Кафедра обчислювальної техніки**

**Лабораторна робота № 2.1**

з дисципліни  
«Алгоритми і структури даних»

Виконав

Студент групи ІП-03  
Пашковський Євгеній Сергійович  
номер у списку групи: 17

Перевірила:

Сергієнко А. А.

## Завдання

Дане натуральне число  $n$ . Знайти суму перших  $n$  членів ряду чисел, заданого рекурентною формулою. Розв'язати задачу трьома способами (написати три програми):

1) в програмі використати рекурсивну процедуру або функцію, яка виконує обчислення і членів ряду, і суми на рекурсивному спуску;

2) в програмі використати рекурсивну процедуру або функцію, яка виконує обчислення і членів ряду, і суми на рекурсивному поверненні;

3) в програмі використати рекурсивну процедуру або функцію, яка виконує обчислення членів ряду на рекурсивному спуску, а обчислення суми на рекурсивному поверненні.

Програми повинні працювати коректно для довільного натурального  $n$  включно з  $n = 1$ .

Варіант 17:

$$F_1 = x; \quad F_{i+1} = F_i \cdot x^2 / (4i^2 + 2i); \quad i > 0;$$

$$\sum F_i = \operatorname{sh} x, \quad |x| < 10^6;$$

## Текст програм

- 1) в програмі використати рекурсивну процедуру або функцію, яка виконує обчислення і членів ряду, і суми на рекурсивному спуску:

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

int main()
{
    system("cls");

    double x;
    unsigned int n = 0;

    printf("x: ");
    scanf("%lf", &x);

    while (n < 1) {
        printf("n: ");
        scanf("%u", &n);
    }

    clock_t begin = clock();

    double recursiveFunc(double x, double F, double sumF, unsigned int i)
    {
        F = i == 0 ? x : F * ((x * x) / (4 * i * i + 2 * i));
        sumF += F;
        i++;

        return i < n ? recursiveFunc(x, F, sumF, i) : sumF;
    }

    double sumF = recursiveFunc(x, 0, 0, 0);
    printf("sumF = %.15lf\n", sumF);
    printf("Delta: %.15lf\n", fabs((double)(sinh(x) - sumF)));

    clock_t end = clock();
    printf("Elapsed: %lf seconds\n", (double)(end - begin) / CLOCKS_PER_SEC);

    system("pause");
    return 0;
}
```

- 2) в програмі використати рекурсивну процедуру або функцію, яка виконує обчислення і членів ряду, і суми на рекурсивному поверненні:

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

int main()
{
    system("cls");

    double x;
    unsigned int n = 0;

    printf("x: ");
    scanf("%lf", &x);

    while (n < 1) {
        printf("n: ");
        scanf("%u", &n);
    }

    clock_t begin = clock();

    double sumF = 0;
    double recursiveFunc(double x, unsigned int i)
    {
        double F;
        if (i == 0)
        {
            F = x;
        } else {
            double m = (x * x) / (4 * i * i + 2 * i);
            F = m * recursiveFunc(x, i - 1);
        }
        sumF += F;
        return F;
    }

    double F = recursiveFunc(x, n);

    printf("sumF = %.15lf\n", sumF);
    printf("Delta: %.15lf\n", fabs((double)(sinh(x) - sumF)));

    clock_t end = clock();
    printf("Elapsed: %lf seconds\n", (double)(end - begin) / CLOCKS_PER_SEC);

    system("pause");
    return 0;
}
```

- 3) в програмі використати рекурсивну процедуру або функцію, яка виконує обчислення членів ряду на рекурсивному спуску, а обчислення суми на рекурсивному поверненні

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

int main()
{
    system("cls");

    double x;
    unsigned int n = 0;

    printf("x: ");
    scanf("%lf", &x);

    while (n < 1) {
        printf("n: ");
        scanf("%u", &n);
    }

    clock_t begin = clock();

    double recursiveFunc(double x, double F, unsigned int i)
    {
        F = i == 0 ? x : F * ((x * x) / (4 * i * i + 2 * i));
        i++;
        return i < n ? F + recursiveFunc(x, F, i) : F;
    }

    double sumF = recursiveFunc(x, 0, 0);

    printf("sumF = %.15lf\n", sumF);
    printf("Delta: %.15lf\n", fabs((double)(sinh(x) - sumF)));

    clock_t end = clock();
    printf("Elapsed: %lf seconds\n", (double)(end - begin) / CLOCKS_PER_SEC);

    system("pause");
    return 0;
}
```

## Тестування програми

*Циклічна програма:*

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

int main()
{
    system("cls");

    double x;
    unsigned int n = 0;

    printf("x: ");
    scanf("%lf", &x);

    while (n < 1) {
        printf("n: ");
        scanf("%u", &n);
    }

    clock_t begin = clock();

    double F = x;
    double sumF = F;
    for (int i = 1; i < n; i++)
    {
        F = F * ((x * x) / (4 * i * i + 2 * i));
        sumF += F;
    }

    // printf("\nF%u = %.15lf\n", n, recursiveFunc(x));
    printf("sumF = %.15lf\n", sumF);
    printf("Delta: %.15lf\n", fabs((double)(sinh(x) - sumF)));

    clock_t end = clock();
    printf("Elapsed: %lf seconds\n", (double)(end - begin) / CLOCKS_PER_SEC);

    system("pause");
    return 0;
}
```

*Результат циклічної програми:*

```
C:\Users\eeegen\OneDrive\Рабочий стол\ДЗ\Лабы\АСД\2 семестр\2.1\Programs\Lab2.1-2(cycle).exe
x: 3
n: 5
sumF = 10.013169642857143
Delta: 0.004705284552759
Elapsed: 0.001000 seconds
Для продолжения нажмите любую клавишу . . . |
```

```
C:\Users\eeegen\OneDrive\Рабочий стол\ДЗ\Лабы\АСД\2 семестр\2.1\Programs\Lab2.1-2(cycle).exe
x: 3
n: 1
sumF = 3.000000000000000
Delta: 7.017874927409902
Elapsed: 0.000000 seconds
Для продолжения нажмите любую клавишу . . . |
```

*Результат трёх программ:*

```
C:\Users\eeegen\OneDrive\Рабочий стол\ДЗ\Лабы\АСД\2 семестр\2.1\Programs\Lab2.1-2(1).exe
x: 3
n: 5
sumF = 10.013169642857143
Delta: 0.004705284552759
Elapsed: 0.001000 seconds
Для продолжения нажмите любую клавишу . . .
```

```
C:\Users\eeegen\OneDrive\Рабочий стол\ДЗ\Лабы\АСД\2 семестр\2.1\Programs\Lab2.1-2(1).exe
x: 3
n: 1
sumF = 3.000000000000000
Delta: 7.017874927409902
Elapsed: 0.000000 seconds
Для продолжения нажмите любую клавишу . . .
```

```
C:\Users\eeegen\OneDrive\Рабочий стол\ДЗ\Лабы\АСД\2 семестр\2.1\Programs\Lab2.1-2(2).exe
x: 3
n: 5
sumF = 10.013169642857143
Delta: 0.004705284552759
Elapsed: 0.001000 seconds
Для продолжения нажмите любую клавишу . . .
```



```
Выбрать C:\Users\eeegen\OneDrive\Рабочий стол\ДЗ\Лабы\АСД\2 семестр\2.1\Programs\Lab2.1-2(2).exe
x: 3
n: 1
sumF = 3.0000000000000000
Delta: 7.017874927409902
Elapsed: 0.000000 seconds
Для продолжения нажмите любую клавишу . . . |
```

```
C:\Users\eeegen\OneDrive\Рабочий стол\ДЗ\Лабы\АСД\2 семестр\2.1\Programs\Lab2.1-2(3).exe
x: 3
n: 5
sumF = 10.013169642857143
Delta: 0.004705284552759
Elapsed: 0.000000 seconds
Для продолжения нажмите любую клавишу . . . |
```

```
C:\Users\eeegen\OneDrive\Рабочий стол\ДЗ\Лабы\АСД\2 семестр\2.1\Programs\Lab2.1-2(3).exe
x: 3
n: 1
sumF = 3.0000000000000000
Delta: 7.017874927409902
Elapsed: 0.000000 seconds
Для продолжения нажмите любую клавишу . . . |
```

Обчислення за допомогою калькулятора:

Гиперболические функции

X

3

Точность вычисления  
Знаков после запятой: 20

РАССЧИТАТЬ

Значения гиперболических функций

↓

Обозначение	Наименование	Значение
sh	Гиперболический синус	10.0178749274099

Графік залежності похибки від  $x$  при  $n = 5$ :

