



Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Санкт-Петербургский государственный технологический институт
(технический университет)»

Дисциплина: «Программирование»

Отчёт по лабораторной работе № 4

Лабораторная работа №4. Использование библиотек динамической компоновки.

Выполнил студент группы №485:
Зобнин Илья Михайлович

Проверили:
Иван Григорьевич Корниенко
Алексей Константинович Федин

Санкт-Петербург
2019

1. Постановка задачи

Необходимо написать программу, в которой для вычисления функции, используется динамически подключенная библиотека `dll`. В результате работы на экране пользователя должна быть выведена таблица с вычисленными значениями рядной $(\sum_{n=1}^{\infty} (-1)^{n+1} \frac{2^{2n-1} x^{2n}}{(2n)!})$ и стандартной $(\sin^2 x)$ функций и их невязки $(\delta = \sqrt{MyFunc(x)^2 - Function(x)^2})$ в интервале $x_{start} - x_{end}$ для заданной точности ϵ . Вычисление ряда проводить до выполнения следующего условия: $|F(x_n) - F(x_{n-1})| < eps$. Также нужно вывести значения невязки для x_{ideal} со значениями точности ϵ от 0.1 до 0.1⁷.

2. Исходные данные

В качестве исходных данных используются начальный x_{start} , шаг Δx , конечный x_{end} , точность ϵ , а также x_{ideal} , вводимые пользователем.

3. Особые ситуации

- Если пользователь, при вводе числа, введёт вместо цифры любой другой символ, программа попросит ввести число заново.
- При $x > 17$ или $x < -17$ рядная функция начинает вести себя неестественно, поэтому при значениях больше или меньше вышеупомянутых рядная функция вернёт ноль, и невязка тоже будет равна нулю.
- Если программа не найдёт `dll` библиотеку, будет выведена ошибка, и программа закроется.

4. Математические методы и алгоритмы решения задач

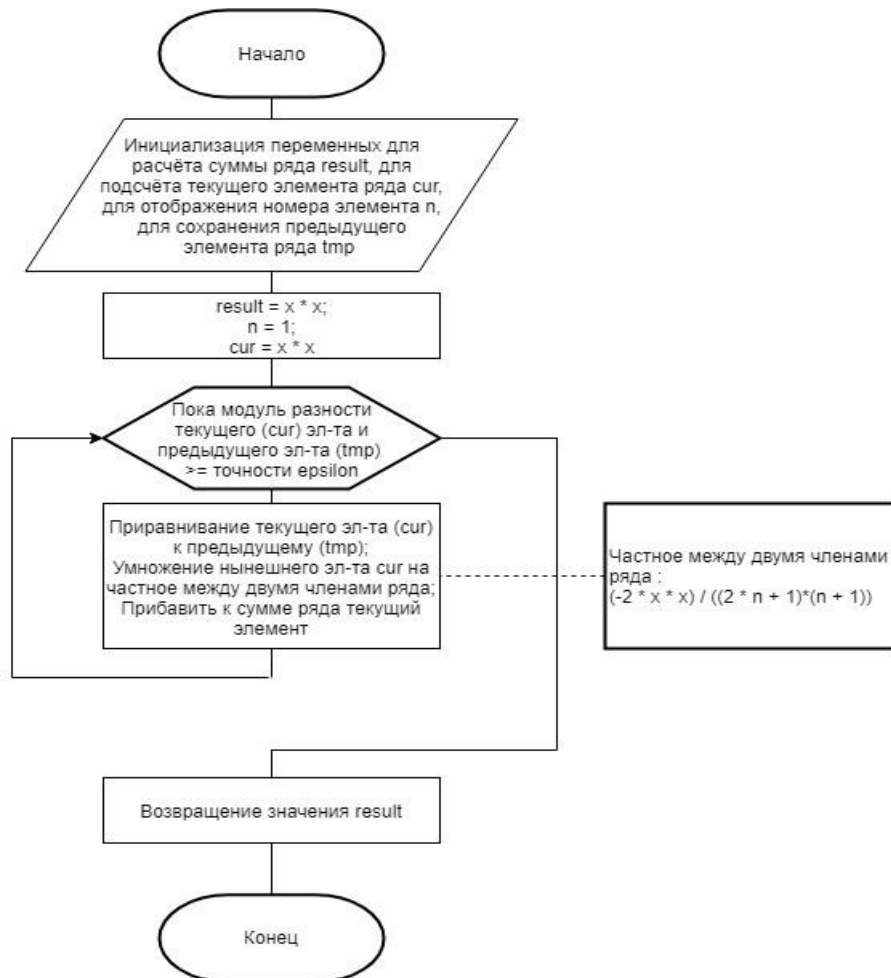
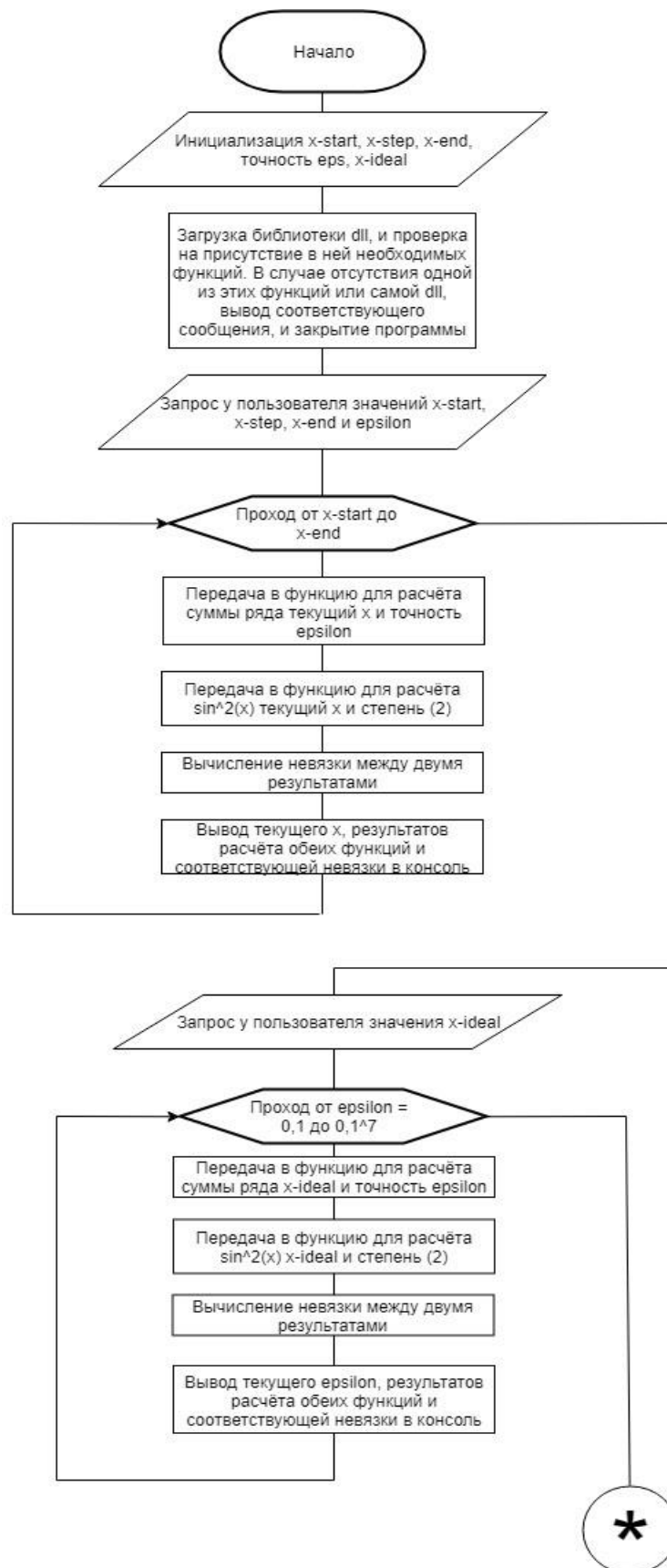


Рисунок 1 – Функция для расчёта суммы ряда



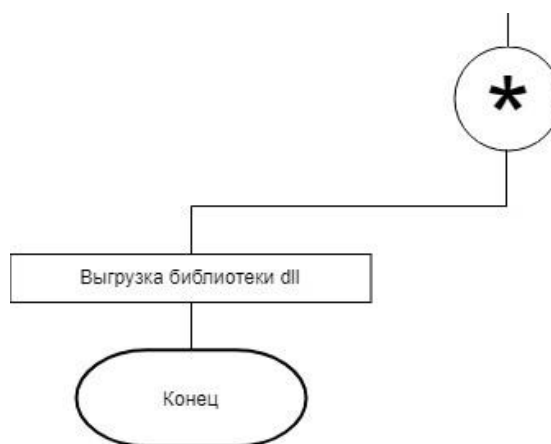


Рисунок 2 – Функция, загружающая библиотеку dll и, выполняющая с её помощью расчёты

5. Форматы представления данных

Программа использует следующие переменные:

Таблица 1 – Переменные, используемые в программе

Имя	Тип	Описание
running	bool	Определяет запущена ли программа
start	double	Начальная граница промежутка вычисления
step	double	Шаг
end	double	Конечная граница промежутка вычисления
ideal	double	Для расчёта невязки при eps от 0,1 до $0,1^7$
cur	long double	Вычисление текущего эл-та ряда
tmp	long double	Запоминание предыдущего эл-та ряда
result	long double	Сумма ряда
funcName	char*	Имя функции
userInput	typename T	Проверка и возврат введенных пользователем данных

6. Структура программы

Из-за необходимости подключения библиотеки dll, решение было разделено на 2 проекта, один для создания dll, второй для самой программы. В проекте, отведённом для dll библиотеки (MathDll), содержится один модуль,

отвечающий за расчёт функции $\sin^2(x)$, суммы ряда, а также вывод имени функции. В проекте самой программы (Lab4) содержится 3 модуля. Один отвечает за запуск программы, второй за её меню и проверку введённых пользователем данных, и третий за выполнение задание и вывод результатов работы программы.

Модуль MathDLL:

Таблица 2 – Функции, составляющие модуль MathDLL проекта MathDLL

Имя	Описание
Зобнин_func	Расчёт суммы ряда
myf_485	Получение значения x , точности ϵ , вызов функции, производящей расчёт суммы ряда
myf_math	Расчёт $\sin^2(x)$ путём функций библиотеки <code>cmath</code>
FName	Вывод имени функции

Модуль Lab4:

Таблица 3 – Функции, составляющие модуль Lab4 проекта Lab4

Имя	Описание
main	Запуск программы

Модуль Math:

Таблица 4 – Функции, составляющие модуль Math проекта Lab4

Имя	Описание
InputAndCalc	Выполнение задания и вывод результатов в консоль

Модуль MenuColorAndCheck:

Таблица 5 – Функции, составляющие модуль MenuColorAndCheck проекта Lab4

Имя	Описание
SetColor	Изменение цвета текста в консоли
Menu	Вывод меню в консоль
GetInput	Проверка и возврат введённых пользователем данных
GetBool	Ввод пользователем значения типа bool
GetDouble	Ввод пользователем значения типа double

7. Описание хода выполнения лабораторной работы

- В ходе лабораторной работы было создано решение (Lab4) в интегрированной среде разработки Microsoft Visual Studio C++ 2017. В нём было создано 2 проекта.
- При получении пользовательского ввода необходимо было добавить проверку на его соответствие предполагаемому типу данных и условиям выбора.
- Перед повторением программы необходимо выгружать библиотеку dll для того, чтобы не возникало непредвиденных ошибок.

8. Результат работы программы

В результате работы на экране пользователя должна быть выведена таблица с вычисленными значениями рядной и стандартной функций и их невязки в интервале $x_{start} - x_{end}$ для заданной точности ε . Также будут выведены значения невязки для x_{ideal} со значениями точности ε от 0.1 до 0.1⁷.

```

Здравствуйте, эта программа сравнивает значения функции  $\sin^2(x)$ , вычисленной с помощью функций из библиотеки cmath,
с суммой соответствующего ряда Маклорена.
Input x start: 0.1
Input x end: dfds
Try again:
gjdifjg
Try again:
gdf ggdfd
Try again:
_

```

Рисунок 3 – Проверка вводимых пользователем данных

```

Здравствуйте, эта программа сравнивает значения функции  $\sin^2(x)$ , вычисленной с помощью функций из библиотеки cmath,
с суммой соответствующего ряда Маклорена.
Input x start: -20
Input x end: 20
Input x step: 3
Input epsilon: 0.1
x          My sin^2(x)          sin^2(x)          disperency
-20        0                0.833469                0
-17        0.92428          0.924285          0.00315701
-14        0.981676        0.981303          0.0270783
-11        1.00027          0.99998          0.0241264
-8         0.979058        0.97883          0.0211627
-5         0.91971         0.919536         0.0179067
-2         0.826919        0.826822         0.0126974
1          0.707937        0.708073          0.0139236
4          0.572405        0.57275          0.0198816
7          0.431189        0.431631          0.0195309
10         0.295408        0.295959          0.0180429
13         0.17585         0.17654          0.0155945
16         0.0819005       0.0828883         0.0127585
19         0                0.0224632         0
Input x ideal:

```

Рисунок 4 – Вывод результатов вычислений функций и их невязки для заданных границ x (-20, 20), шага x (3) и точности (0,1)

```

Input x ideal: 1.45
epsilon    My sin^2(x)          sin^2(x)          disperency
0.1        0.985832          0.985479          0.0263735
0.01       0.985463          0.985479          0.00569993
0.001      0.98548          0.985479          0.00107098
0.0001     0.985479          0.985479          0.000178018
1e-05      0.985479          0.985479          2.65346e-05
1e-06      0.985479          0.985479          2.65346e-05
1e-07      0.985479          0.985479          3.58524e-06
[0] - Exit the program
[1] - Keep running the program
Your choice: _

```

Рисунок 5 – Вывод результатов вычислений функций и их невязки для x -ideal (1,45).

9. Текст программы

```
[--- Начало программы ---]
// MathDLL.cpp
// Лабораторная работа №4.
// Студент группы 485, Зобнин Илья Михайлович. 2019 год
#include "MathDLL.h"
#include <cmath>
#include <string>

using namespace std;

//функция для вычисления рядной функции
double Зобнин_func(double x, double eps) {
    long double result = x * x;
    size_t n = 1;
    long double cur = result;
    long double tmp;
    do {
        tmp = cur;
        cur *= (-2 * x * x) / ((2 * n + 1)*(n + 1));
        result += cur;
        n++;
    } while (abs(cur - tmp) >= eps);
    return result;
}

// для вызова внутренней функции
double MATHDLL_API myf_485(double x, double y, double eps) {
    if (x > 17 || x < -17)
        return 0;
    return Зобнин_func(x, eps);
}

//вычисление показательной функции
double MATHDLL_API myf_math(double x, double a) {
    return pow(sin(x), a);
}

//возвращает имя функции
char MATHDLL_API *FName() {
    string str = "sin^2(x)";

    char* funcName = new char[str.length()];
    for (int i = 0; i < str.length(); ++i) {
        funcName[i] = str[i];
    }
    funcName[str.length()] = '\0';
    return funcName;
}

// MathDLL.h
// Лабораторная работа №4.
// Студент группы 485, Зобнин Илья Михайлович. 2019 год
#ifdef MATHDLL_EXPORTS
#define MATHDLL_API __declspec(dllexport)
#else
#define MATHDLL_API __declspec(dllimport)
#endif
```

```
extern "C" {
    //вычисление рядной функции
    double MATHDLL_API myf_485(double x, double a, double eps);
    //вычисление показательной функции
    double MATHDLL_API myf_math(double x, double a);
    //возвращает имя функции
    char MATHDLL_API *FName();
}
```

```
// Lab4.cpp
// Лабораторная работа №4.
// Студент группы 485, Зобнин Илья Михайлович. 2019 год
#include <iostream>
#include <cmath>
#include <string>
#include "MenuColorAndCheck.h"
#include "Math.h"

using namespace std;

int main() {
    setlocale(LC_ALL, "rus");
    SetColor(azure);
    cout << "Здравствуйте, эта программа сравнивает значения функции  $\sin^2(x)$ ,
вычисленной с помощью функций из библиотеки cmath,"
    << endl << "с суммой соответствующего ряда Маклорена." << endl;
    SetColor(white);
    bool running = true;
    while (running) {
        InputAndCalc();
        running = Menu();
    }
    system("PAUSE");
}
```

```
// Math.cpp
// Лабораторная работа №4.
// Студент группы 485, Зобнин Илья Михайлович. 2019 год
#include <iostream>
#include <vector>
#include <cmath>
#include <Windows.h>
#include <iomanip>
#include "MenuColorAndCheck.h"
#include "Math.h"

using namespace std;

struct FirstTable {
    double x;
    double myFuncResult;
    double mathFuncResult;
    double discrepancy = 0;
    FirstTable(double x, double myFuncResult, double mathFuncResult) : x(x),
myFuncResult(myFuncResult),
    mathFuncResult(mathFuncResult) {
```

```

        if (static_cast<long long>(myFuncResult*10e7) != 0)
            discrepancy = sqrt(abs(pow(myFuncResult, 2) - pow(mathFuncResult,
2)))));
    }
};
struct SecondTable {
    double eps;
    double myFuncResult;
    double mathFuncResult;
    double discrepancy = 0;
    SecondTable(double eps, double myFuncResult, double mathFuncResult) : eps(eps),
myFuncResult(myFuncResult),
    mathFuncResult(mathFuncResult) {
        if (static_cast<long long>(myFuncResult*10e7) != 0)
            discrepancy = sqrt(abs(pow(myFuncResult, 2) - pow(mathFuncResult,
2)))));
    }
};
void InputAndCalc() {
    HINSTANCE hGetProcIDDLL = LoadLibrary(TEXT("MathDLL.dll")); //загрузка библиотеки
    if (!hGetProcIDDLL) {
        cout << "Error opening dll. Press any key to exit." << endl;
        getchar();
        exit(1);
    }

    Myf_485 myf_485 = (Myf_485)GetProcAddress(hGetProcIDDLL, "myf_485"); //загрузка
функции
    if (!myf_485) {
        cout << "Error loading function. Press any key to exit." << endl;
        getchar();
        exit(2);
    }

    FName FName = (FName)GetProcAddress(hGetProcIDDLL, "FName"); //загрузка функции
    if (!FName) {
        cout << "Error loading function. Press any key to exit." << endl;
        getchar();
        exit(2);
    }

    Myf_math myf_math = (Myf_math)GetProcAddress(hGetProcIDDLL, "myf_math");
//загрузка функции
    if (!myf_math) {
        cout << "Error loading function. Press any key to exit." << endl;
        getchar();
        exit(2);
    }
    vector<FirstTable> firstTable;
    double start, step, end, eps, ideal, a;
    cout << "Input x start: ";
    start = GetDouble();
    do {
        cout << "Input x end: ";
        end = GetDouble();
    } while (end < start);
    do {
        cout << "Input x step: ";
        step = GetDouble();
    } while (start + step <= start);
    do {
        cout << "Input epsilon: ";
        eps = GetDouble();
    } while (eps > 0.1 || eps < 10e-7);

```

```

    cout << "Input a parametr: ";
    a = GetDouble();
    double i = start;
    double k = step;
    start = static_cast<long long>(start * 10e7);
    end = static_cast<long long>(end * 10e7);
    step = static_cast<long long>(step * 10e7);

    for (long long j = start; j <= end; j += step, i += k) {
        firstTable.emplace_back(i, myf_485(i, a, eps), myf_math(i, 2));
    }

    cout << setw(20) << left << "x" << setw(3) << left << "My " << setw(17) << left <<
    FName() << setw(20) << left << FName() << setw(20) << left << "disperency" << endl;
    SetColor(green);
    cout << fixed << setprecision(7);
    for (const auto it : firstTable) {
        cout << setw(20) << left << it.x << setw(20) << left << it.myFuncResult <<
        setw(20) << left << it.mathFuncResult << setw(20) << left << it.discrepancy << endl;
    }
    SetColor(white);
    cout << endl;

    vector<SecondTable> secondTable;
    cout << "Input x ideal: ";
    ideal = GetDouble();
    eps = 0.1;
    for (int j = 0; j < 7; j++, eps /= 10)
        secondTable.emplace_back(eps, myf_485(ideal, a, eps), myf_math(ideal, 2));

    cout << setw(20) << left << "epsilon" << setw(3) << left << "My " << setw(17) <<
    left << FName() << setw(20) << left << FName() << setw(20) << left << "disperency" <<
    endl;
    SetColor(purple);
    for (const auto it : secondTable)
        cout << setw(20) << left << it.eps << setw(20) << left << it.myFuncResult
        << setw(20) << left << it.mathFuncResult << setw(20) << left << it.discrepancy << endl;
    SetColor(white);

    firstTable.clear();
    secondTable.clear();
    FreeLibrary(hGetProcIDDLL);
}

```

// MenuColorAndCheck.cpp

// Лабораторная работа №4.

// Студент группы 485, Зобнин Илья Михайлович. 2019 год

#include <Windows.h>

#include <iostream>

#include "MenuColorAndCheck.h"

using namespace std;

```

void SetColor(const int color) { //функция для изменения цвета текста в консоли
    SetConsoleTextAttribute(GetStdHandle(STD_OUTPUT_HANDLE), color);
}

```

```

bool Menu() {
    cout << "[0] - Exit the program" << endl << "[1] - Keep running the program" <<
    endl << "Your choice: ";
    bool menu = GetBool();
}

```

```

        return menu;
    }

template <typename T>
T GetInput() { // Проверка ввода целого числа
    T userInput;
    cin >> userInput; //ввод пользователем необходимых данных
    while (cin.fail()) { //цикл пока ввод данных не соответствует заданному типу
        SetColor(red);
        cout << "Try again: " << endl;
        SetColor(white);
        cin.clear(); //обнуление cin.fail
        cin.ignore(INT_MAX, '\n'); //игнорирование введённых данных
        cin >> userInput; //повторный ввод переменной
    }
    cin.ignore(INT_MAX, '\n');
    return userInput;
}

bool GetBool() {
    return GetInput<bool>();
}

double GetDouble() {
    return GetInput<double>();
}

// Math.h
// Лабораторная работа №4.
// Студент группы 485, Зобнин Илья Михайлович. 2019 год
#pragma once
void InputAndCalc();

typedef double (*Myf_485)(double, double, double); //вычисление рядной функции
typedef double (*Myf_math)(double, double); //вычисление функции из math.h
typedef char* (*FName)(); //запрос имени функции

// MenuColorAndCheck.h
// Лабораторная работа №4.
// Студент группы 485, Зобнин Илья Михайлович. 2019 год
#pragma once
enum { blue = 9, green, azure, red, purple, yellow, white };

void SetColor(const int color);

double GetDouble();

bool GetBool();

bool Menu();

[--- Конец программы ---]

```