МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение

высшего образования

«Кубанский государственный университет»

Факультет компьютерных технологий и прикладной математики

Кафедра математического моделирования

**ОТЧЕТ О ПРОХОЖДЕНИИ УЧЕБНОЙ ПРАКТИКИ**

(практике по получению первичных профессиональных умений и навыков, в том числе первичных умений и навыков научно-исследовательской деятельности)

Выполнила                                                                                        М.М. Шадрина

Направление подготовки 02.03.03 Математическое обеспечение и администрирование информационных систем  
Курс   1

Руководитель учебной практики

канд. физ.-мат. наук, доцент кафедры

математического моделирования                                                       С.Е. Рубцов

Краснодар

2023

**СОДЕРЖАНИЕ**

[1 Постановка задачи 3](#_Toc133411023)

[2 Описание численного метода (метод левых прямоугольников) 5](#_Toc133411024)

[3 Аналитическое вычисление интеграла 7](#_Toc133411025)

[4 Описание работы программы 10](#_Toc133411026)

[5 Результаты численных расчетов 11](#_Toc133411027)

[Список используемых источников 17](#_Toc133411028)

[Приложение Текст программы 18](#_Toc133411029)

# 1 Постановка задачи

Пусть задана функция в виде интеграла с параметром:

1) Написать программу на языке высокого уровня (СИ, Visual Basic, Delphi) для расчёта значений этой функции на промежутке:

Для вычисления интеграла использовать метод левых прямоугольников с шагом h:

Произвести расчёты для различных значений *N* разбиения интервала интегрирования. (Например, при ). В программе предусмотреть ввод исходных данных c, d, N.

2) Аналитически вычислить интеграл. В той же программе произвести сравнение точного и приближённого решений: вычислить максимальную невязку (наибольшую по абсолютной величине разность между точным и приближённым решениями для различных значений )

3) В одной системе координат построить графики точного и приближённого решений. Для построения графиков использовать графические возможности выбранного языка программирования или Microsoft Excel. (При использовании Microsoft Excel должен быть описан процесс передачи данных).

4) Создать в электронном виде отчёт по учебной практике. При этом использовать стандарты, принятые для оформления курсовых и дипломных работ.

Отчет должен включать: постановку задачи, описание численного метода, используемого в работе, аналитическое вычисление интеграла, графики точного и приближенных решений, полученных для различных значений N, вычисленные значения максимальных невязок и текст программы.

# 2 Описание численного метода (метод левых прямоугольников)

Пусть на отрезке , задана непрерывная функция . Требуется вычислить интеграл , численно равный площади соответствующей криволинейной трапеции.

Разобьем отрезок на *n* частей точками .

Внутри каждого отрезка выберем точку . Так как по определению определенный интеграл есть предел интегральных сумм при бесконечном уменьшении длины элементарного отрезка разбиения:

следовательно, любая из таких интегральных сумм является приближенным значением интеграла

Если отрезок интегрирования разбить на равные части длины *h* точками:

и в качестве точек выбрать левые границы элементарных отрезков (), то получим приближенное равенство, соответствующее формуле метода левых прямоугольников:

# 3 Аналитическое вычисление интеграла

Для вычисления определённого интеграла:

Первоначально рассмотрим вычисление соответственного неопределённого интеграла:

Для этого используем подстановку:

Вынесем вещественный коэффициент за знак интеграла и осуществим вторую подстановку:

Используем третью подстановку:

Используем табличный интеграл для решения:

Проведём обратные замены:

Далее вычислим соответствующий определённый интеграл по формуле Ньютона-Лейбница:

Так как *x* принимает значение 0 на заданном отрезке, из за чего при решении мы получим деление на 0, необходимо отдельно рассмотреть случай x=0:

Вычислим определённый интеграл по формуле Ньютона-Лейбница и получим:

# 4 Описание работы программы

Для вычисления численного значения интеграла воспользуемся возможностями языка высокого уровня C++.

Определим функцию Function, для вычисления точного значения функции при данных значений x и t, функцию IntegralSum, для вычисления значения интеграла в заданной точке х, для количества разбиений n и с шагом h.

Определим переменные типа double: x, y, h, a, b, c, d, для хранения значения х, значения функции, шага, нижней границы интегрирования, верхней границы интегрирования, нижней границы для х и верхней границы для x соответственно.

Определим два массива типа double: N[4] (он будет хранить информацию о необходимых количествах разбиений), и discrepancy[4] (он будет хранить значения максимальной невязки).

Для промежуточного хранения данных выдаваемых программой определим строку s.

Впоследствии, перебирая x, будем высчитывать сначала значение интеграла аналитическим методом и записывать его в строку s, разделяя знаками табуляции. После, тоже проделаем для четырёх различных разбиений. Сравнивая значения, полученные разными способами, будем записывать в массив discrepancy[4] их разницу, если значение невязки получается больше чем то, которое до этого находилось в массиве.

Для дальнейшего использования записанной в строке s информации в Microsoft Excel, заменим в s все символы «.» на «,», в цикле от 0 до длины строки s.

Подключим текстовый файл table.txt и запишем туда все данные из строки s. После чего запишем в этот же файл значения максимальных невязок для каждого из значений разбиения и закроем файл table.txt.

# 5 Результаты численных расчетов

Результаты численных расчётов связанных с решением данной задачи представлены в двух таблицах 1 и 2.

Таблица 1 Максимальная невязка при соответствующих разбиениях интервала интегрирования.

|  |  |
| --- | --- |
| Количество точек в разбиении | Максимальная невязка |
| 5 | 0.141421 |
| 10 | 0.0707107 |
| 20 | 0.0353553 |
| 50 | 0.0141421 |

Таблица 2 Данные, используемые для построения графиков приближённого и точного решений.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Х | Точное решение | Приближённые решения при соответствующих разбиениях | | | |
| 5 | 10 | 20 | 50 |
| 0 | 0,707107 | 0,565685 | 0,636396 | 0,671751 | 0,692965 |
| 0,1 | 0,698197 | 0,558563 | 0,62838 | 0,663288 | 0,684233 |
| 0,2 | 0,689153 | 0,551343 | 0,62025 | 0,654702 | 0,675373 |
| 0,3 | 0,679992 | 0,544036 | 0,612017 | 0,646005 | 0,666398 |
| 0,4 | 0,670729 | 0,536652 | 0,603695 | 0,637213 | 0,657323 |
| 0,5 | 0,661379 | 0,5292 | 0,595296 | 0,628339 | 0,648164 |
| 0,6 | 0,65196 | 0,521692 | 0,586834 | 0,619399 | 0,638936 |
| 0,7 | 0,642487 | 0,514137 | 0,578321 | 0,610407 | 0,629656 |
| 0,8 | 0,632978 | 0,506547 | 0,569773 | 0,601378 | 0,620339 |
| 0,9 | 0,623448 | 0,498931 | 0,5612 | 0,592327 | 0,611 |

Продолжение таблицы 2

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Х | Точное решение | Приближённые решения при соответствующих разбиениях | | | |
| 5 | 10 | 20 | 50 |
| 1 | 0,613914 | 0,491301 | 0,552618 | 0,583268 | 0,601656 |
| 1,1 | 0,60439 | 0,483666 | 0,544037 | 0,574216 | 0,592321 |
| 1,2 | 0,594891 | 0,476036 | 0,535472 | 0,565183 | 0,583009 |
| 1,3 | 0,585432 | 0,468422 | 0,526932 | 0,556184 | 0,573733 |
| 1,4 | 0,576026 | 0,460831 | 0,51843 | 0,547229 | 0,564507 |
| 1,5 | 0,566685 | 0,453274 | 0,509976 | 0,53833 | 0,555342 |
| 1,6 | 0,55742 | 0,445757 | 0,50158 | 0,529498 | 0,546251 |
| 1,7 | 0,548242 | 0,43829 | 0,493251 | 0,520743 | 0,537242 |
| 1,8 | 0,539162 | 0,430879 | 0,484997 | 0,512074 | 0,528325 |
| 1,9 | 0,530186 | 0,423532 | 0,476827 | 0,503499 | 0,519509 |
| 2 | 0,521324 | 0,416253 | 0,468747 | 0,495025 | 0,510802 |

Далее на рисунке 1 представлены данные выводимые программой на консоль, при входных данных: 0 1 0 2 5 10 20 50.

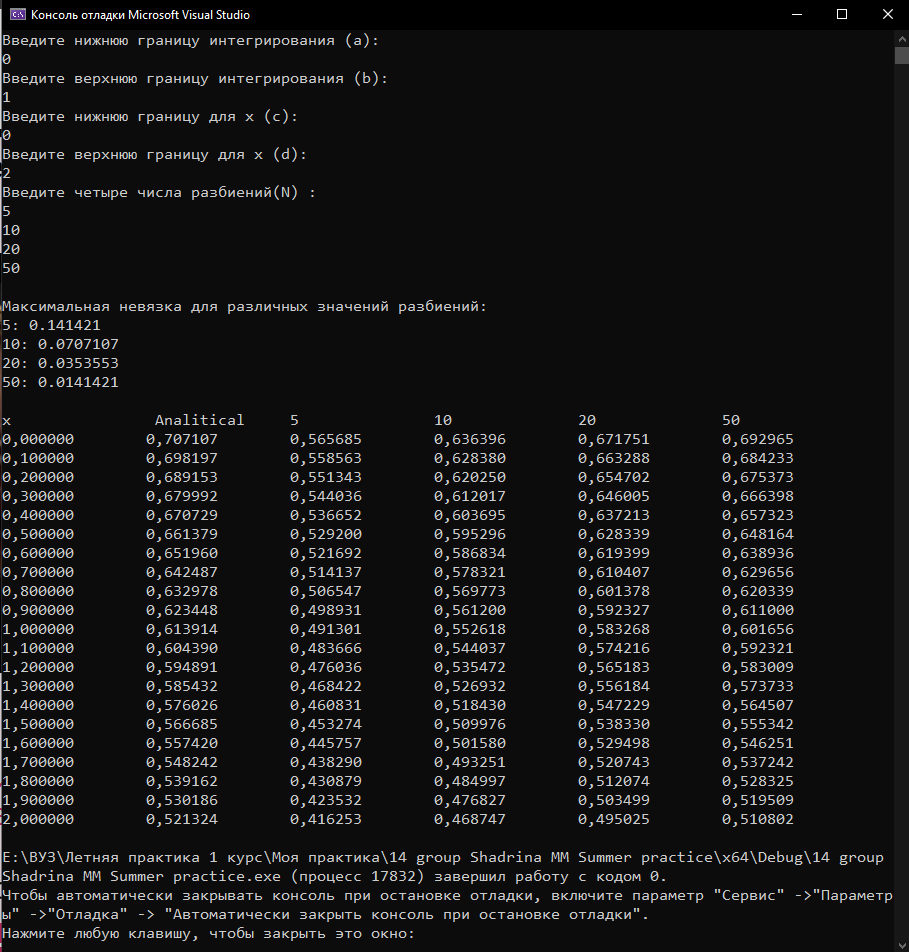


Рисунок 1 – Данные выводимые программой на консоль.

Для создания графиков функций используем Excel.

Откроем файл table.txt (рисунок 2) через Excel, с помощью мастера текстов зададим необходимые нам параметры импорта. В качестве символа-разделителя выберем знак табуляции.

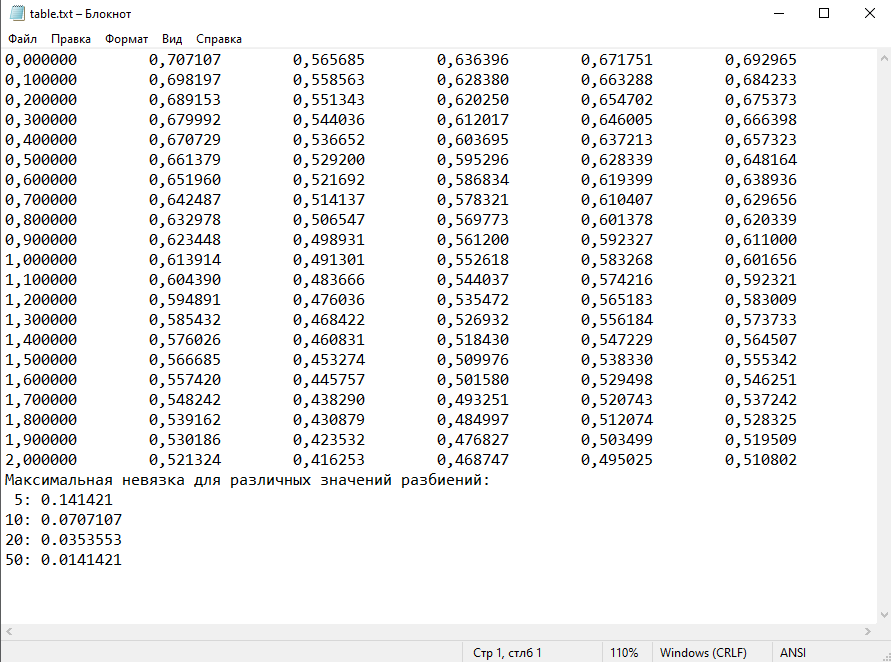


Рисунок 2 – Заполненный программой файл table.txt.

Далее по данным файла построим графики, основываясь на полученных приближённых и точных значениях, в одной системе координат (рисунок 3 и 4).

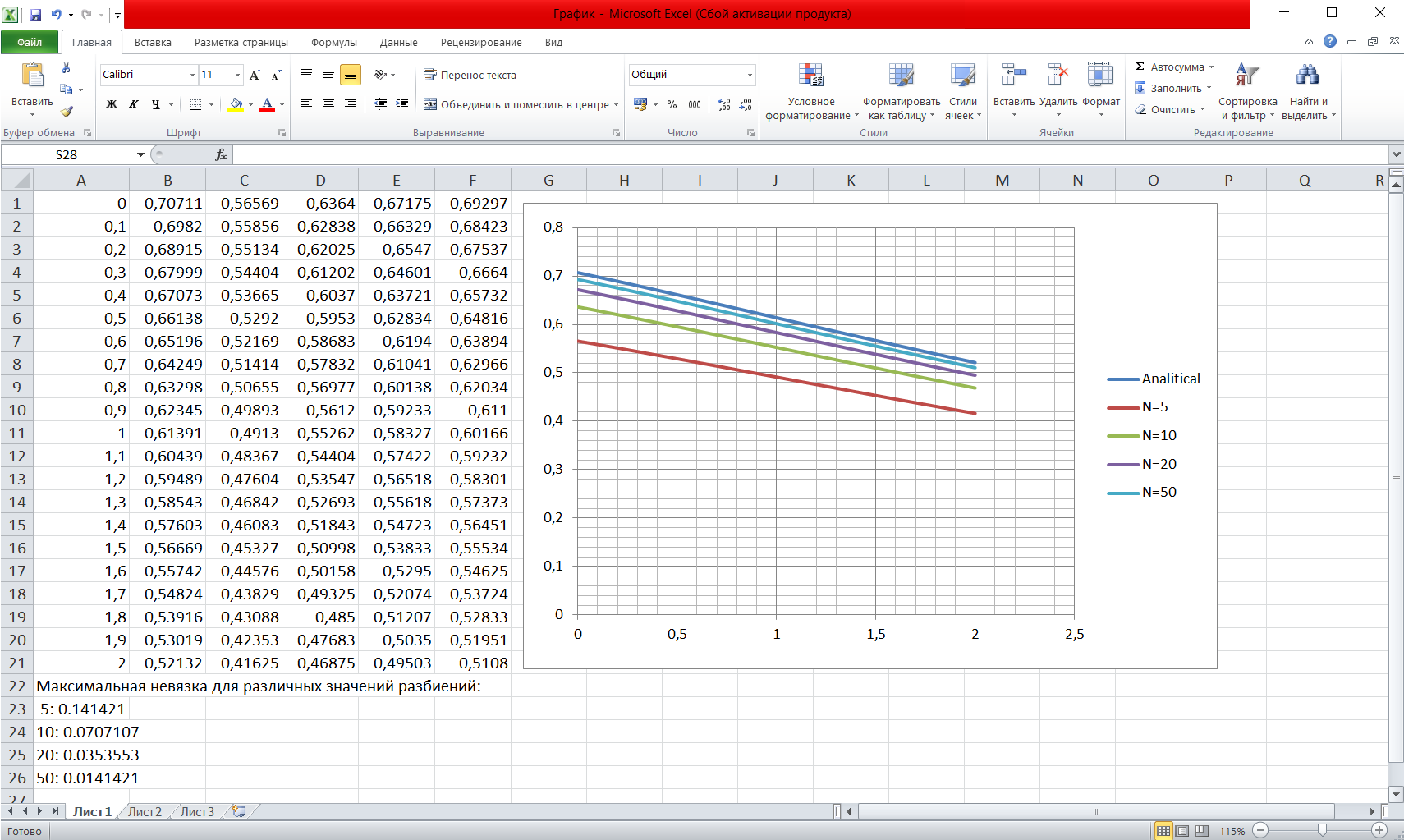


Рисунок 3 – График с данными из файла table.txt.

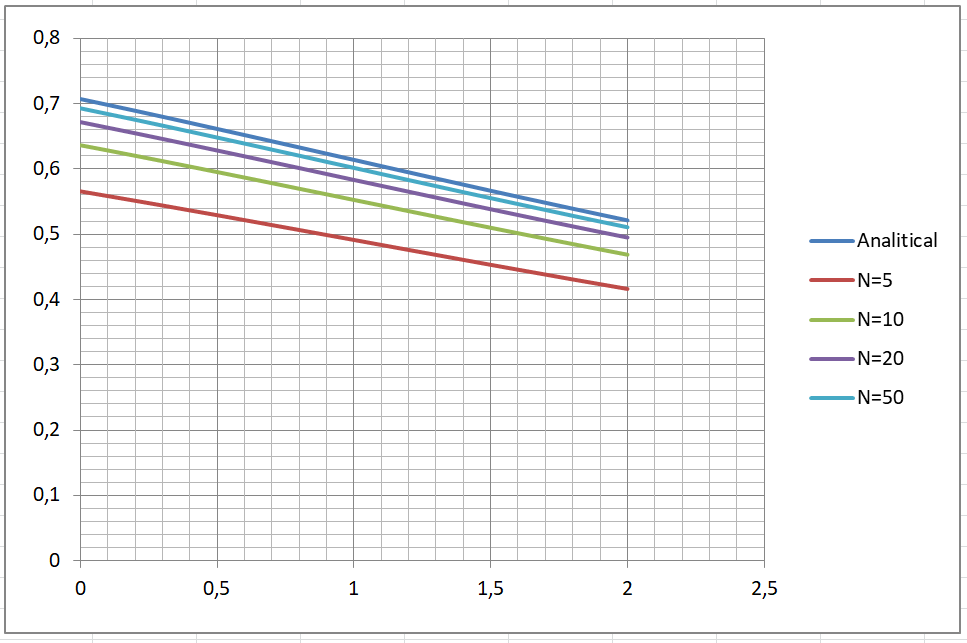


Рисунок 4 – График на основе данных программы.

# СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Мотов В.В. Word, Excel, Power Point: учебное пособие /  
   В.В. Мотов – М.: ИНФРА-М, 2010 – 206 с. – ISBN 978-5-16-003495-9.
2. Кудрявцев Л.Д. Сборник задач по математическому анализу. Том 2. Интегралы. / Л.Д. Кудрявцев – М.: ФИЗМАТЛИТ, 2003. – 504 с. – ISBN 5-9221-0307-5.
3. Методы программирования: учеб.-метод. пособие /  
    В.В. Подколзин и др. − Краснодар: Кубанский гос. ун-т, 2020 – 174 с. –   
   ISBN 978-5-8209-1768-4.

# ПРИЛОЖЕНИЕ

Текст программы.

#define \_USE\_MATH\_DEFINES

#include <iostream>;

#include <string>;

#include <fstream>;

#include <cmath>;

using namespace std;

double Function(double x, double t)

{

return (1 / (pow(1 + pow(M\_E, t \* x), 0.5)));

}

double IntegralSum(double x, double n, double h)

{

double sum = 0;

for (int i = 1; i < n; ++i)

sum += (Function(x, h \* i)) \* h;

return sum;

}

int main()

{

setlocale(LC\_ALL, "Russian");

double x, y, h, a, b, c, d, N[4], discrepancy[4] = { 0, 0, 0, 0 };

cout << "Введите нижнюю границу интегрирования (a): " << endl;

cin >> a;

cout << "Введите верхнюю границу интегрирования (b): " << endl;

cin >> b;

cout << "Введите нижнюю границу для х (c): " << endl;

cin >> c;

cout << "Введите верхнюю границу для х (d): " << endl;

cin >> d;

cout << "Введите четыре числа разбиений(N) : " << endl;

cin >> N[0] >> N[1] >> N[2] >> N[3];

string s = "";

for (int i = 0; i <= 20; i++)

{

x = c + i \* (d - c) / 20;

if (x == 0)

{

y = pow(2, 0.5) / 2;

s += to\_string(x) + '\t' + to\_string(y) + '\t';

}

else

{

y = ((log((pow(pow(M\_E, x) + 1, 0.5) - 1) / (pow(pow(M\_E, x) + 1, 0.5) + 1)) + log((pow(2, 0.5) + 1) / (pow(2, 0.5) - 1))) / x);

s += to\_string(x) + '\t' + to\_string(y) + '\t';

}

for (int j = 0; j < 4; j++)

{

h = (b - a) / N[j];

s += to\_string(IntegralSum(x, N[j], h)) + '\t';

if (abs(y - IntegralSum(x, N[j], h)) > discrepancy[j])

discrepancy[j] = abs(y - IntegralSum(x, N[j], h));

}

s += '\n';

}

for (int i = 0; i < s.length(); i++)

if (s[i] == '.')

s[i] = ',';

ofstream file("table.txt");

file << s;

file << "Максимальная невязка для различных значений разбиений: \n ";

cout << endl << "Максимальная невязка для различных значений разбиений:" << endl;

for (int i = 0; i < 4; i++)

{

file << N[i] << ": " << discrepancy[i] << "\n";

cout << N[i] << ": " << discrepancy[i] << "\n";

}

cout << endl << "x \t\t" << " Analitical" << "\t" << N[0] << "\t\t" << N[1] << "\t\t" << N[2] << "\t\t" << N[3] << "\t\t" << '\n' << s;

file.close();

return 0;

}