

ОГЛАВЛЕНИЕ

Общая информация	2
1. Приближённые числа и источники погрешностей	4
1.1. Абсолютная и относительная погрешности	4
2. Вычисление функций	5
3. Приближённое решение уравнений	6
3.1. Графическое решение уравнений	6
3.2. Метод половинного деления	6
3.3. Метод хорд	6
3.4. Метод Ньютона	7
3.5. Метод итерации	7
3.6. Метод Ньютона для комплексных корней	7
3.7. Метод Бернулли решения алгебраических уравнений	8
4. Элементы теории матриц	9
4.1. Сложение и умножение матриц	9
4.2. Определитель	9
4.3. Обратная матрица	10
5. Линейные уравнения	11
5.1. Метод Крамера	11
5.2. Метод Гаусса	11
5.3. Схема Халецкого	11
5.4. Метод итераций	12
5.5. Метод Зейделя	12
6. Векторные преобразования	13
6.1. Скалярное и векторное произведения векторов	13
6.2. Ортогонализация методом Грамма-Шмидта	13
6.3. Преобразование Фурье	13
6.4. Матрица унитарного преобразования Фурье	13
6.5. Прямое и обратное преобразование Фурье на основе умножения на матрицу преобразования Фурье	14
6.6. Интерполяционные формулы Ньютона	Ошибка! Закладка не определена.
7. Интерполяция и приближение функций	15
7.1. Интерполяционные формулы Ньютона	15
7.2. Интерполяция Гаусса	15
7.3. Сплайн интерполяция	15
7.4. Фурье интерполяция	16
8. Приближённое дифференцирование	17
8.1. Конечные разности	17
8.2. Производная с помощью преобразования Фурье	17
8.3. Интерполяционные формулы Ньютона	17
9. Приближённое интегрирование	18
9.1. Функции для интегрирования	18
9.2. Метод прямоугольников	18
9.3. Метод трапеций	18
9.4. Метод Симпсона	18
9.5. Метод Ньютона-Кортеса	19
СПИСОК ЛИТЕРАТУРЫ	20

ОБЩАЯ ИНФОРМАЦИЯ

Данный документ содержит перечень индивидуальных заданий по курсу «Вычислительная математика». Регулярное и своевременное выполнение домашних заданий является необходимым фактором успешного обучения и положительной аттестации. При выполнении домашнего задания следует внимательно и не спеша ознакомиться с теорией, обдумать метод решения и алгоритм реализации. При необходимости обратиться к дополнительным учебным материалам, список основных учебных материалов содержится в завершительной части данного документа.

Необходимо помнить, что выполнение домашнего задания это способ получения навыков работы, технического специалиста которые понадобятся в дальнейшей трудовой деятельности.

При оформлении домашнего задания следует руководствоваться следующими нормативными документами и стандартами:

- ГОСТ 7.32–2017 межгосударственный стандарт. Система стандартов по информации, библиотечному и издательскому делу. Отчет о научно-исследовательской работе. Структура и правила оформления;

- ГОСТ 19.701-90 (ИСО 5807-85) межгосударственный стандарт. Единая система программной документации. Схемы алгоритмов, программ, данных и систем. Обозначения условные и правила выполнения

- ГОСТ 7.80-2000 межгосударственный стандарт. Система стандартов по информации, библиотечному и издательскому делу. Библиографическая запись. Заголовок;

- ГОСТ 7.1-2003 межгосударственный стандарт. Система стандартов по информации, библиотечному и издательскому делу. Библиографическая запись. Библиографическое описание. Общие требования и правила составления.

Отчёт о выполнении домашнего задания должен обязательно содержать:

- титульный лист (по ГОСТ 7.32–2017);
- текст задания;
- элементы теории и методов решения поставленной задачи;
- решение и проверку решения задачи, в том числе: блок-схему алгоритма (ГОСТ 19.701-90); листинг программы (C или C++); результаты выполнения программы; результаты проверки; графики, рисунки и чертежи по теме задания и пр.

- заключение, выводы;

- список литературы.

Программы необходимо писать на языке C или C++ на основе функционального или объектно-ориентированного подхода, с соблюдением хорошего стиля и логического структурирования программы. В случае недостаточной уверенности в знаниях языков программирования C или C++ следует обратиться к литературе (список имеется в приложении). Все программы пишутся в консольном исполнении, т.е. без графического интерфейса. Исходные данные вводятся из консоли, при достаточно больших данных, например, для ввода матриц рекомендуется предусмотреть в

программе ввод данных из файла. Для компиляции программ рекомендуется использовать в OS Linux g++ или gcc, в OS Windows MinGW.

Для построения графиков, данные программы, разработанной в рамках выполнения задания можно вывести в файл, а затем, используя полученный файл, построить графики в Excel, MatLab или иной программе.

Необходимо помнить, что полученное правильное решение для одного или нескольких частных случаев не подтверждает правильности алгоритма для всех случаев. Потому, для проверки программы, необходимо применять не только расчёт некоторых частных примеров, но и комплексный анализ программы на предмет соответствия теоретическим основам.

В завершении каждого задания необходимо выполнить анализ полученных результатов, произвести асимптотическую оценку вычислительной сложности, сравнить реализованный метод решения с другими предназначенными для подобных задач, написать заключение и выводы.

Желательно унифицировать внутренние и внешние интерфейсы программ, это упростит повторное использование кода и выполнение заданий. Для задач, в которых размер входных данных может изменяться, необходимо использовать динамическое выделение памяти.

1. ПРИБЛИЖЁННЫЕ ЧИСЛА И ИСТОЧНИКИ ПОГРЕШНОСТЕЙ

1.1. Абсолютная и относительная погрешности

Задание.

Наименование задачи:

Вид решения: программа и отчёт.

Реализация решения: язык С или С++.

Разработать и написать программу вычисления абсолютной и относительной погрешности сложения (вычитания) и произведения (частного).

2. ВЫЧИСЛЕНИЕ ФУНКЦИЙ

2.1. Схема Горнера

Задание.

Наименование задачи: схема Горнера.

Вид решения: программа и отчёт.

Реализация решения: язык С или С++.

Разработать и написать программу реализующую схему Горнера для деления многочленов на двучлены и вычисляющую коэффициенты многочлена-частного и остаток. Результат выводить в виде многочлена-частного и остатка. Входные параметры: коэффициенты многочлена, величина ξ , являющаяся свободным членом делящего двучлена $(x - \xi)$. Программу снабдить проверкой правильности расчётов через умножение.

2.2. Ряд Тейлора и Маклорена

Задание.

Наименование задачи: ряды Тейлора.

Вид решения: программа и отчёт.

Реализация решения: язык С или С++.

Разработать и написать программу для вычисления логарифмической и гармонической функции с помощью рядов Тейлора (Маклорена), с возможностью выбора числа членов рядов вычисления функций, точность вычисления определить с помощью расчёта остаточного члена. Осуществить проверку вычислений: путём вычисления значений функции в точке и сравнение ошибки вычисления и величины остаточного члена. Построить график зависимости ошибки, точного значения и остаточного члена в зависимости от числа членов ряда.

3. ПРИБЛИЖЁННОЕ РЕШЕНИЕ УРАВНЕНИЙ

3.1. Графическое решение уравнений

Задание.

Наименование задачи:

Вид решения: программа и отчёт.

Реализация решения: язык С или С++.

Графически определить верхние и нижние границы корней уравнений, проверить справедливость теоремы об отделении корней.

$$5x^4 - 2x^3 + 3x^2 + x - 4 = 0;$$

$$x^5 - 3x^4 + 7x^2 + x - 8 = 0.$$

3.2. Метод половинного деления

Задание.

Наименование задачи:

Вид решения: программа и отчёт.

Реализация решения: язык С или С++.

Методом половинного деления найти хотя бы один вещественный корень уравнения, решение снабдить графиками функций и указать найденные корни:

$$2x^3 + x^2 - 7 = 0;$$

$$5 \cos(3x) + 0.5x = 2, x \in [0; 2\pi];$$

$$x^5 - 2x^4 + 6x^2 + 2x - 4 = 0;$$

$$x^3 - 0.2x^2 - 0.2x - 1.2 = 0.$$

$$\ln(|x^3| + 1) + x^3 = 2.$$

Произвести оценку вычислительной сложности метода. Посчитать число итераций для решения уравнения с заданной точностью.

3.3. Метод хорд

Задание.

Наименование задачи:

Вид решения: программа и отчёт.

Реализация решения: язык С или С++.

Методом хорд найти хотя бы один вещественный корень уравнения, решение снабдить графиками функций и указать найденные корни:

$$5^x \sqrt{8^{x-1}} - 189 = 0;$$

$$x^3 - x^2 + 2x - 5 = 0;$$

$$2 \lg x^2 - 5 \lg^2 x - 4 = 0;$$

$$2 \sin(2x) - \cos(3x) = 0.5, x \in [0; 2\pi];$$

$$2x^3 - 7x^2 - 7x - 2.5 = 0.$$

Произвести оценку вычислительной сложности метода. Посчитать число итераций для решения уравнения с заданной точностью.

3.4. Метод Ньютона

Задание.

Наименование задачи:

Вид решения: программа и отчёт.

Реализация решения: язык С или С++.

Методом Ньютона найти хотя бы один вещественный корень уравнения, решение снабдить графиками функций и указать найденные корни:

$$2\lg x - \cos x = 0, x \in (0; 4\pi];$$

$$2x^3 - 5x^2 - 1 = 0;$$

$$2\sin^3(2x) - \cos(x) = 0, x \in [0; \pi];$$

$$x^5 - 3x^4 + 8x^2 + 2x - 7 = 0;$$

$$0.5x^2 + 5\cos(2x) - 2 = 0, x \in [-\pi; \pi].$$

Произвести оценку вычислительной сложности метода. Посчитать число итераций для решения уравнения с заданной точностью.

3.5. Метод итерации

Задание.

Наименование задачи:

Вид решения: программа и отчёт.

Реализация решения: язык С или С++.

Методом итераций найти хотя бы один вещественный корень уравнения, решение снабдить графиками функций и указать найденные корни:

$$\cos x = 0.1, x \in (0; 4\pi];$$

$$x^3 + x = 1000;$$

$$x^5 - x^4 - x^2 - x - 5 = 0;$$

$$x^3 - x - 1 = 0;$$

$$\ln x + x = 2.25.$$

Произвести оценку вычислительной сложности метода. Посчитать число итераций для решения уравнения с заданной точностью.

3.6. Метод Ньютона для комплексных корней

Задание.

Наименование задачи:

Вид решения: программа и отчёт.

Реализация решения: язык С или С++.

Методом Ньютона с заданной точностью найти хотя бы один комплекснозначный корень следующих уравнений:

$$4z^4 + 2z^2 + 1.3 = 0; z^2 + 2.71 = 0; 2e^z + \sqrt{2} = 0.$$

Произвести оценку вычислительной сложности метода. Посчитать число итераций для решения уравнения с заданной точностью.

3.7. Метод Бернулли решения алгебраических уравнений

Задание.

Наименование задачи:

Вид решения: программа и отчёт.

Реализация решения: язык С или С++.

Методом Бернулли с заданной точностью найти хотя бы один корень следующих уравнений.

$$5x^4 - 2x^3 + 3x^2 + x - 4 = 0.$$

$$x^5 + 5x^4 - 5 = 0.$$

Произвести оценку вычислительной сложности метода. Посчитать число итераций для решения уравнения с заданной точностью.

4. ЭЛЕМЕНТЫ ТЕОРИИ МАТРИЦ

4.1. Сложение и умножение матриц

Задание.

Наименование задачи: элементарные операции над матрицами.

Вид решения: программа и отчёт.

Реализация решения: язык C или C++.

Разработать алгоритм и написать программу реализующую: ввод комплекснозначных матриц из консоли или из файла и следующие операции над матрицами, включая проверку реализуемости операции: сложение, вычитание, умножение на комплексное число, умножение матрицы на матрицу. Оценить вычислительную сложность различных операций.

4.2. Определитель

Задание.

Наименование задачи: вычисление определителя на основе элементарных преобразований матрицы.

Вид решения: программа и отчёт.

Реализация решения: язык C или C++.

Разработать алгоритм и написать программу реализующую: ввод вещественнозначных матриц из консоли и из файла, и вычисление определителя на основе элементарных преобразований матрицы. Оценить вычислительную сложность вычисления определителя на основе элементарных преобразований матрицы.

Задание.

Наименование задачи: вычисление определителя матрицы с комплексными элементами на основе элементарных преобразований матрицы.

Вид решения: программа и отчёт.

Реализация решения: язык C или C++.

Разработать алгоритм и написать программу реализующую: ввод комплекснозначных матриц из консоли и из файла, и вычисление определителя на основе элементарных преобразований матрицы. Оценить вычислительную сложность вычисления определителя на основе элементарных преобразований матрицы.

Задание.

Наименование задачи: вычисление определителя методом Гаусса.

Вид решения: программа и отчёт.

Реализация решения: язык C или C++.

Текст задания

Разработать алгоритм и написать программу реализующую: вычисление определителя методом Гаусса, для матриц заданных на множестве вещественных чисел. Предусмотреть возможность вычисления определителя для матриц различного размера без перекомпиляции программы. Исходную

матрицу вводить в программу из консоли или из файла. Оценить вычислительную сложность решения задачи.

4.3. Обратная матрица

Задание.

Наименование задачи: вычисление обратной матрицы.

Вид решения: программа и отчёт.

Реализация решения: язык С или С++.

Разработать алгоритм и написать программу реализующую: ввод вещественнозначных матриц из консоли и из файла, и вычисление обратной матрицы на основе элементарных преобразований матрицы. В алгоритм программы интегрировать проверку путём умножения вычисленной обратной матрицы на исходную. Оценить вычислительную сложность вычисления обратной матрицы на основе элементарных преобразований матрицы.

5. ЛИНЕЙНЫЕ УРАВНЕНИЯ

5.1. Метод Крамера

Задание.

Наименование задачи: решение системы линейных уравнений методом Крамера на множестве вещественных чисел.

Вид решения: программа и отчёт.

Реализация решения: язык C или C++.

Разработать алгоритм и написать программу реализующую: решение системы линейных уравнений, на множестве вещественных чисел, методом Крамера. Предусмотреть возможность решения уравнений различного порядка без перекомпиляции программы. Исходные данные (матрицу коэффициентов уравнения, вектор правых частей), вводить в программу из консоли или из файла. Оценить вычислительную сложность решения задачи.

5.2. Метод Гаусса

Задание.

Наименование задачи: решение системы линейных уравнений методом Гаусса на множестве вещественных чисел.

Вид решения: программа и отчёт.

Реализация решения: язык C или C++.

Разработать алгоритм и написать программу реализующую: решение системы линейных уравнений, на множестве вещественных чисел, методом Гаусса. Предусмотреть возможность решения уравнений различного порядка без перекомпиляции программы. Исходные данные (матрицу коэффициентов уравнения, вектор правых частей), вводить в программу из консоли или из файла. Оценить вычислительную сложность решения задачи.

Задание.

Наименование задачи: решение системы линейных уравнений методом Гаусса с ведущим элементом на множестве комплексных чисел.

Вид решения: программа и отчёт.

Реализация решения: язык C или C++.

Разработать алгоритм и написать программу реализующую: решение системы линейных уравнений, на множестве комплексных чисел, методом Гаусса с ведущим элементом. Предусмотреть возможность решения уравнений различного порядка без перекомпиляции программы. Исходные данные (матрицу коэффициентов уравнения, вектор правых частей), вводить в программу из консоли или из файла. Оценить вычислительную сложность решения задачи.

5.3. Схема Халецкого

Задание.

Наименование задачи: решение системы линейных уравнений с помощью метода *схема Халецкого*.

Вид решения: программа и отчёт.

Реализация решения: язык C или C++.

Разработать алгоритм и написать программу реализующую: решение системы линейных уравнений, на множестве вещественных чисел, методом *схема Халецкого*. Предусмотреть возможность решения уравнений различного порядка без перекомпиляции программы. Исходные данные (матрицу коэффициентов уравнения, вектор правых частей), вводить в программу из консоли или из файла. Оценить вычислительную сложность решения задачи.

5.4. Метод итераций

Задание.

Наименование задачи: решение системы линейных уравнений с помощью метода итераций.

Вид решения: программа и отчёт.

Реализация решения: язык C или C++.

Разработать алгоритм и написать программу реализующую: решение системы линейных уравнений, на множестве вещественных чисел, методом итераций. Предусмотреть возможность решения уравнений различного порядка без перекомпиляции программы. Исходные данные (матрицу коэффициентов уравнения, вектор правых частей), вводить в программу из консоли или из файла. Оценить вычислительную сложность решения задачи.

5.5. Метод Зейделя

Задание.

Наименование задачи: решение системы линейных уравнений с помощью метода Зейделя.

Вид решения: программа и отчёт.

Реализация решения: язык C или C++.

Разработать алгоритм и написать программу реализующую: решение системы линейных уравнений, на множестве вещественных чисел, методом Зейделя. Предусмотреть возможность решения уравнений различного порядка без перекомпиляции программы. Исходные данные (матрицу коэффициентов уравнения, вектор правых частей), вводить в программу из консоли или из файла. Оценить вычислительную сложность решения задачи.

6. ВЕКТОРНЫЕ ПРЕОБРАЗОВАНИЯ

6.1. Скалярное и векторное произведения векторов

Задание.

Наименование задачи: скалярное и векторное произведения векторов.

Вид решения: программа и отчёт.

Реализация решения: язык C или C++.

Разработать алгоритм и написать программу реализующую: скалярное и векторное произведения векторов (строк на строки и столбцов на столбцы) из одной матрицы. Матрицу и её размер вводить в программу из консоли или из файла. Предусмотреть возможность выбора размера матрицы без перекомпиляции программы, размер матрицы $N \times N$, где $N = 1, 2, 3, \dots$. Оценить вычислительную сложность скалярного и векторного произведения векторов.

6.2. Ортогонализация методом Грамма-Шмидта

Задание.

Наименование задачи: ортогонализация методом Грамма-Шмидта.

Вид решения: программа и отчёт.

Реализация решения: язык C или C++.

Разработать алгоритм и написать программу реализующую: ортогонализацию методом Грамма-Шмидта строк из одной матрицы. Матрицу и её размер вводить в программу из консоли или из файла. Предусмотреть возможность выбора размера матрицы без перекомпиляции программы, размер матрицы $N \times N$, где $N = 1, 2, 3, \dots$. Выполнить проверку ортогональности и нормировки с помощью вычисления скалярного произведения и расчёта длин векторов. Оценить вычислительную сложность скалярного и векторного произведения векторов.

6.3. Преобразование Фурье

Задание.

Наименование задачи: преобразование Фурье.

Вид решения: программа и отчёт.

Реализация решения: язык C или C++.

Разработать алгоритм и написать программу реализующую: преобразование Фурье для разложения последовательности комплекснозначных чисел в ряд Фурье. Длина последовательности произвольная. Оценить вычислительную сложность преобразования Фурье.

6.4. Матрица унитарного преобразования Фурье

Задание.

Наименование задачи: матрица преобразования Фурье.

Вид решения: программа и отчёт.

Реализация решения: язык С или С++.

Разработать алгоритм и написать программу реализующую: вычисление, вывод в консоль матрицы преобразования Фурье и проверки полученной матрицы на унитарность. Предусмотреть возможность выбора размера матрицы без перекомпиляции программы, размер матрицы $N \times N$, где $N = 2^n, n = 1, 2, 3, \dots$.

6.5. Прямое и обратное преобразование Фурье на основе умножения на матрицу преобразования Фурье

Задание.

Наименование задачи: прямое и обратное преобразование Фурье на основе умножения на матрицу преобразования Фурье.

Вид решения: программа и отчёт.

Реализация решения: язык С или С++.

Разработать алгоритм и написать программу реализующую: прямое и обратное преобразование Фурье на основе умножения на матрицу преобразования Фурье. Предусмотреть возможность выбора размера последовательности преобразования без перекомпиляции программы, размер последовательности N , где $N = 2^n, n = 1, 2, 3, \dots$. Оценить вычислительную сложность прямого и обратного преобразования Фурье на основе умножения на унитарную матрицу преобразования Фурье.

7. ИНТЕРПОЛЯЦИЯ И ПРИБЛИЖЕНИЕ ФУНКЦИЙ

7.1. Интерполяционные формулы Ньютона

Задание.

Наименование задачи: вторая интерполяционная формула Ньютона.

Вид решения: программа и отчёт.

Реализация решения: язык C или C++.

Разработать алгоритм и написать программу реализующую: первую интерполяционную формулу Ньютона. Отчёт снабдить графиками. Произвести анализ результатов. Предусмотреть возможность выбора размера последовательности преобразования без перекомпиляции программы, размер последовательности N , где $N = 5, 6, 7 \dots$. Оценить вычислительную сложность.

7.2. Интерполяция Гаусса

Задание.

Наименование задачи: первая и вторая интерполяционная формула Гаусса.

Вид решения: программа и отчёт.

Реализация решения: язык C или C++.

Разработать алгоритм и написать программу реализующую: интерполяцию на основе первой и второй интерполяционной формулы Гаусса. Произвести сравнение и анализ результатов. Предусмотреть возможность выбора размера последовательности преобразования без перекомпиляции программы, размер последовательности N , где $N = 5, 6, 7 \dots$. Оценить вычислительную сложность.

7.3. Сплайн интерполяция

Задание.

Наименование задачи: найти приближение функции заданной в равноотстоящих точках, т.е. функции заданной в виде последовательности чисел, с помощью сплайна третьей степени.

Вид решения: программа и отчёт.

Реализация решения: язык C или C++.

Разработать алгоритм и написать программу реализующую: найти приближение функции заданной в равноотстоящих точках, т.е. функции заданной в виде последовательности чисел, с помощью сплайна третьей степени. Произвести анализ результатов. Предусмотреть возможность выбора размера последовательности преобразования без перекомпиляции программы, размер последовательности N , где $N = 5, 6, 7 \dots$. Оценить вычислительную сложность.

7.4. Фурье интерполяция

Задание.

Наименование задачи: найти приближение функции заданной в равноотстоящих точках, т.е. функции заданной в виде последовательности чисел, с помощью интерполяции Фурье.

Вид решения: программа и отчёт.

Реализация решения: язык С или С++.

Разработать алгоритм и написать программу реализующую: найти приближение функции заданной в равноотстоящих точках, т.е. функции заданной в виде последовательности чисел, с помощью интерполяции Фурье. Произвести анализ результатов. Предусмотреть возможность выбора размера последовательности преобразования без перекомпиляции программы, размер последовательности N , где $N = 2^n, n = 1, 2, 3, \dots$. Оценить вычислительную сложность.

8. ПРИБЛИЖЁННОЕ ДИФФЕРЕНЦИРОВАНИЕ

8.1. Конечные разности

Задание.

Наименование задачи: дифференцирование функции, заданной в равноотстоящих точках, с помощью конечных разностей.

Вид решения: программа и отчёт.

Реализация решения: язык С или С++.

Разработать алгоритм и написать программу реализующую: дифференцирование функции (найти производную заданного порядка), заданной в равноотстоящих точках, с помощью правых и левых конечных разностей. Результаты, для производных первого, второго и третьего порядков одной и той же последовательности, отобразить на графике и сравнить.

8.2. Производная с помощью преобразования Фурье

Задание.

Наименование задачи: производная с помощью преобразования Фурье.

Вид решения: программа и отчёт.

Реализация решения: язык С или С++.

Разработать алгоритм и написать программу реализующую: вычисление производной с помощью преобразования Фурье. Результаты, для производной одной и той же последовательности, полученной с помощью преобразования Фурье и правых и левых конечных разностей отобразить на графике и сравнить. Оценить вычислительную сложность вычисления производной с помощью преобразования Фурье.

8.3. Интерполяционные формулы Ньютона

Задание.

Наименование задачи: производная с помощью интерполяционных формул Ньютона.

Вид решения: программа и отчёт.

Реализация решения: язык С или С++.

Разработать алгоритм и написать программу реализующую: вычисление производной с помощью интерполяционных формул Ньютона. Результаты, для производной одной и той же последовательности, полученной с помощью интерполяционных формул Ньютона и правых и левых конечных разностей отобразить на графике и сравнить. Оценить вычислительную сложность вычисления производной с помощью интерполяционных формул Ньютона.

9. ПРИБЛИЖЁННОЕ ИНТЕГРИРОВАНИЕ

9.1. Функции для интегрирования

Функции для интегрирования:

1. $f(x) = x^2, x \in [-5; 5];$
2. $f(x) = \sin^2 x, x \in [-\pi; \pi];$
3. $f(x) = \sin 2x + \cos 7x + 8, x \in [-\pi; \pi];$
4. $f(x) = 2x^4 + x^3 + 2x^2 + 3x + 24, x \in [-1; 3];$
5. $f(x) = \ln(x^2 + 1) + \sin \frac{x}{3} + 17, x \in [-100; 100];$
6. $f(x) = 5^x + \sin x + x + 11, x \in [-\pi; \pi];$
7. $f(x) = x^5 + 2x^4 + 3x^3 + 4x^2 + 5x + 6, x \in [-7; 7];$

9.2. Метод прямоугольников

Задание.

Наименование задачи: произвести интегрирование функций на заданном интервале методом прямоугольников.

Вид решения: программа и отчёт.

Реализация решения: язык С или С++.

Разработать алгоритм и написать программу реализующую: интегрирование функций на заданном интервале методом прямоугольников, расчёт остаточного члена. При равном шаге сетки, сравнить полученный результат с результатами полученными иными методами численного интегрирования. Оценить вычислительную сложность.

9.3. Метод трапеций

Задание.

Наименование задачи: произвести интегрирование функций на заданном интервале методом трапеций.

Вид решения: программа и отчёт.

Реализация решения: язык С или С++.

Разработать алгоритм и написать программу реализующую: интегрирование функций на заданном интервале методом трапеций, расчёт остаточного члена. При равном шаге сетки, сравнить полученный результат с результатами полученными иными методами численного интегрирования. Оценить вычислительную сложность.

9.4. Метод Симпсона

Задание.

Наименование задачи: произвести интегрирование функций на заданном интервале методом Симпсона.

Вид решения: программа и отчёт.

Реализация решения: язык С или С++.

Разработать алгоритм и написать программу реализующую: интегрирование функций на заданном интервале методом Симпсона, расчёт остаточного члена. При равном шаге сетки, сравнить полученный результат с результатами полученными иными методами численного интегрирования. Оценить вычислительную сложность.

9.5. Метод Ньютона-Кортеса

Задание.

Наименование задачи: произвести интегрирование функций на заданном интервале методом Ньютона-Кортеса 3-го и 4-го порядков.

Вид решения: программа и отчёт.

Реализация решения: язык С или С++.

Разработать алгоритм и написать программу реализующую: интегрирование функций на заданном интервале методом Ньютона-Кортеса 3-го и 4-го порядков, расчёт остаточного члена. При равном шаге сетки, сравнить полученный результат с результатами полученными иными методами численного интегрирования. Оценить вычислительную сложность.

СПИСОК ЛИТЕРАТУРЫ

1. Демидович Б.П., Марон И.А. Основы вычислительной математики // М.: Наука, 1970. – 664 с.
2. Калиткин Н.Н. Численные методы // М.: Наука, 1978. – 512 с.
3. Бахвалов Н.С. Численные методы // М.: Наука, 2008. – 636 с.: ил.
4. Седжевик Р. Алгоритмы на C++.: Пер. с англ. – М.: ООО «И.Д. Вильямс», 2011. – 1056 с.: ил.
5. Либерти Д., Бредли Д. Освой самостоятельно C++ за 21 день, 5-е издание. : Пер. с англ. – М.: Издательский дом «Вильямс», 2012. – 768 с.:ил.
6. Прата С. Язык программирования C++. Лекции и упражнения, 6-е изд.: Пер. с англ. – М.: ООО «И.Д. Вильямс», 2013. – 1248 с.: ил. – Парал. тит. англ.
7. Самарский А. А., Вабищевич П. Н. Численные методы решения обратных задач // М.: издательство ЛКИ, 2009. – 480 с.
8. Фоменко А. Т. Наглядная геометрия и топология // Изд. 3-е испр. и доп. – М.: ЛЕНАНД, 2018. – 480 с.: ил.
9. Гришенцев А.Ю. Цифровые системы широкополосной связи. Часть 1. Введение в пространства и методы преобразования сигналов // СПб: Университет ИТМО, 2019. – 76 с.
10. Новиков С. П., Тайманов И. А. Современные геометрические структуры и поля // М.: МЦНМО, 2005. – 584 с.: ил.
11. Гантмахер Ф. Р. Теория матриц // М.: «Наука», главная редакция физико-математической литературы, 1967. – 576 с.: ил.
12. Запорожец Г. И. Руководство к решению задач по математическому анализу // М.: «Высшая школа», 1962. – 404 с.: ил.
13. Дубровин Б. А., Новиков С. П., Фоменко А. Т. Современная геометрия: методы и приложения.Т.1: Геометрия поверхностей, групп преобразований и полей // М.: УРСС: Книжный дом «ЛИБРОКОМ», 2013. – 336 с.
14. Постников М. М. Лекции по геометрии. Семестр I: Аналитическая геометрия // М.: ЛЕНАНД, 2017. – 416 с.
15. Постников М. М. Лекции по геометрии. Семестр II: Линейная алгебра // М.: ЛЕНАНД, 2017. – 400 с.
16. Ильин В. А., Поздняк Э. Г. Линейная алгебра // М.: «Наука», главная редакция физико-математической литературы, 1974. – 206 с.: ил.
17. Ильин В. А., Поздняк Э. Г. Основы математического анализа: В 2-х ч. Часть I // М.: Наука, Физматлит, 1998. – 616 с.
18. Ильин В. А., Поздняк Э. Г. Основы математического анализа: В 2-х ч. Часть II // М.: Наука, Физматлит, 1998. – 448 с.
19. Свешников А. Г., Тихонов А. Н. Теория функций комплексной переменной // М.: Наука, Физматлит, 1999. – 320 с.
20. Шилдт Г. Полный справочник по C++, 4-е изд. //М.: изд. дом «Вильямс», 2009. – 800 с.: ил.
21. Шилдт Г. Полный справочник по C, 4-е изд. //М.: изд. дом «Вильямс», 2009. – 704 с. : ил.

22. Джосаттис Н. М. Стандартная библиотека C++: справочное руководство, 2-е изд. // М.: изд. дом «Вильямс», 2017. – 1136 с.: ил.
23. Лафоре Р. Объектно-ориентированное программирование в C++. Классика Computer Science // СПб.: Питер, 2012. – 928 с.: ил.