ГУАП

КАФЕДРА № 42

ОТЧЕТ
ЗАЩИЩЕН С ОЦЕНКОЙ ____________________

ПРЕПОДАВАТЕЛЬ

канд. техн. наук, доцент		А.В. Аграновский
должность, уч. степень, звание	подпись, дата	инициалы, фамилия

ОТЧЕТ О ЛАБОРАТОРНОЙ РАБОТЕ №5

Интерполяционная кривая Catmull-Rom

по курсу: КОМПЬЮТЕРНАЯ ГРАФИКА

РАБОТУ ВЫПОЛНИЛ

СТУДЕНТ ГР. №	4329		Д.С. Шаповалова
		подпись, дата	инициалы, фамилия

Санкт-Петербург 2024

Содержание

1. Цель работы:

Изучение интерполяционной кривой Catmull-Rom, построение интерполяционной кривой Catmull-Rom с помощью математического пакета и/или языка программирования высокого уровня.

2. Задание:

Написать программу на языке программирования высокого уровня или с помощью математического пакета, которая выполняет построение интерполяционной кривой Catmull-Rom и вычисляет ошибку восстановления. На форме должен находиться график, таблица с координатами опорных точек, а также три кнопки. При нажатии на кнопку 1 – выполнить построение графика гармонических колебаний, опорных точек и таблицы с координатами базовых точек. Кнопка 2 – построение интерполяционной кривой Catmull-Rom на основе гармонических колебаний. Кнопка 3 – построение интерполяционной кривой Catmull-Rom на основе полинома.

Для этого необходимо:

1. Построить график гармонических колебаний.

2. На периоде гармонических колебаний взять N точек, где N равно 4 плюс номер студента в группе.

3. По опорным точкам из пункта 2 построить кривую Catmull-Rom (на том же графике, что и в пункте 1).

4. Рассчитать ошибку восстановления гармонических колебаний кривой Catmull-Rom.

5. Уменьшить число точек на периоде в 2 раза и повторить пункты 1-4.

6. Увеличить число точек на периоде в 2 раза и повторить пункты 1-4.

7. Построить кривую Catmull-Rom на основе полинома N-го порядка (где N берется из пункта 2) и рассчитать ошибку.

Вариант 4329 - $f(x) = 1 - e^{-x^5}$, x = [0; 1,5]

3. Теоретические сведения:

Сплайн — это двумерные геометрические объекты, которые совершенно самостоятельны и могут служить основой для построения более сложных трехмерных тел. Внешне сплайны представляют собой разнообразные линии, форма линии определяется типом вершин, через которые она проходит. Сплайнами могут быть как простейшие геометрические фигуры: прямоугольники, звезды, эллипсы и пр., так и сложные ломаные или кривые, а также контуры текстовых символов.

Интерполяция — это способ нахождения промежуточных значений величины по имеющемуся дискретному набору известных значений. Интерполяция использует значения некоторой функции, заданные в ряде точек, чтобы предсказать значения функции между ними. Перечисленные ниже методы предназначены для создания ряда с более высокой частотой наблюдений на основе ряда с низкой частотой. Например, вычислить ряд с квартальной динамикой на основе ряда годовых данных.

Сплайн Catmull-Rom — это тип кривой, используемой в компьютерной графике и анимации для создания гладких и непрерывных линий, проходящих через заданные контрольные точки. Эта кривая является частным случаем кубических сплайнов и обладает рядом интересных свойств:

- Прохождение через контрольные точки: Кривая Catmull-Rom проходит через все заданные контрольные точки, что делает её особенно полезной в ситуациях, где важно точно следовать заданным путям.
- Параметризация: Обычно кривая параметризуется, что позволяет легко настраивать её форму и создавать плавные движения.
- Гладкость: Catmull-Rom сплайны обеспечивают гладкие переходы между контрольными точками, что делает их идеальными для анимации и моделирования.
- Простота использования: Для построения такой кривой требуется всего четыре контрольные точки (две из которых являются соседями для текущей точки).

Для создания кривой Catmull-Rom требуется набор из четырёх точек P0, P1, P2 и P3, которые задают сегмент кривой между центральными точками P1 и P2. При построении кривой используется параметр t, который варьируется от 0 до 1 и позволяет получить промежуточные значения между P1 и P2. Уравнение для каждой точки на сегменте задается следующим образом (1):

$$R(t) = {}^{1}\!/_{2}\left(-t(1-t)^2 P_0 + (2 - 5t^2 + 3t^3)P_1 + t(1 + 4t - 3t^2)P_2 - t^2(1-t)P_3\right),$$
$$0 \le t \le 1. \quad (1)$$

где P_0, P_1, P_2, P_3 — координаты точек, определяющих сегмент;

Этот метод применим к любому участку кривой. Он позволяет объединить несколько участков в единый сплайн, который проходит через все опорные точки.

Если требуется создать замкнутую кривую, можно добавить копии начальной и конечной точек в начало и конец набора опорных точек. Чтобы создать замкнутый сплайн, нужно добавить в набор опорных точек копии начальной и конечной точек. Это позволяет сделать плавный переход в начале и конце кривой.

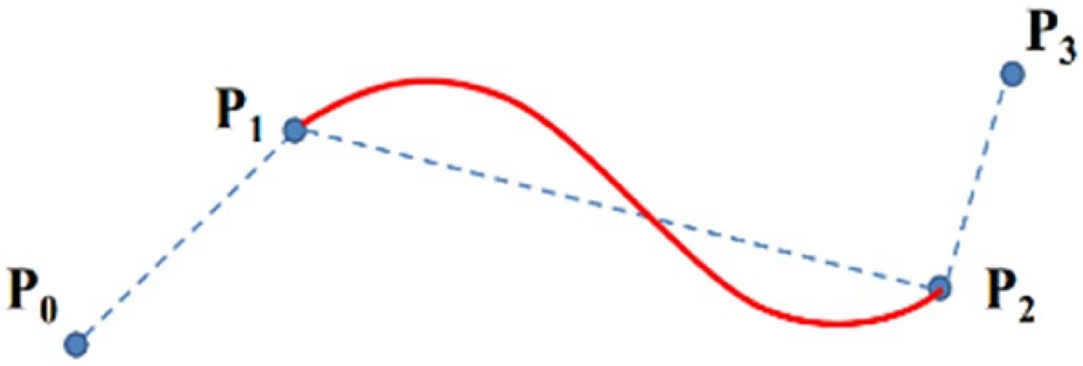

Рисунок 1 – Сплайновая кривая Catmull-Rom

Для оценки точности построенной кривой в работе применяют метод средней абсолютной ошибки (MAE). Он рассчитывается как среднее значение абсолютных отклонений между значениями исходной функции и значениями аппроксимации.

Ошибка восстановления считается по формуле (2):

$$\frac{1}{N}\sum_{i=1}^{N}\left|y_{\text{истинное}}(x_i) - y_{\text{апрокс}}(x_i)\right|, \tag{2}$$

где $y_{\text{истинное}}$ — значения исходной функции;

$y_{\text{аппрокс}}$ — значения, полученные с помощью Catmull-Rom сплайна.

Чтобы построить функцию, которая будет похожа на исходные данные, используют полиномиальную аппроксимацию. Потом с помощью этой функции строят Catmull-Rom сплайн. Так можно понять, насколько точно работает Catmull-Rom сплайн, если использовать не настоящие значения функции, а приближённые.

4. Листинг с кодом программы:

```
import numpy as np
import matplotlib.pyplot as plt
from scipy.interpolate import PchipInterpolator
import tkinter as tk
from tkinter import ttk
from matplotlib.backends.backend_tkagg import FigureCanvasTkAgg

N = 42  # Количество точек (4+17)
Z = 10 # Количество точек после запятой для ошибки восстановления
x_range = np.linspace(0, 1.5, 500)  # Диапазон значений x для построения
графиков

def f(x):
    return 1 - np.exp(-x ** 5)

def calculate_error(true_vals, approx_vals):
    return np.mean(np.abs(true_vals - approx_vals))  # Среднее абсолютное
отклонение

# Функция для замкнутого Catmull-Rom сплайна
def catmull_rom_spline(x_points, y_points):
    x_points = np.concatenate(([x_points[0]], x_points, [x_points[-1]]))
    y_points = np.concatenate(([y_points[0]], y_points, [y_points[-1]]))
    spline_x = []
    spline_y = []
    for i in range(1, len(x_points) - 2):
        P0, P1, P2, P3 = x_points[i - 1], x_points[i], x_points[i + 1],
x_points[i + 2]
        Q0, Q1, Q2, Q3 = y_points[i - 1], y_points[i], y_points[i + 1],
y_points[i + 2]

        t_values = np.linspace(0, 1, 100)
        for t in t_values:
            xt = 0.5 * ((2 * P1) + (-P0 + P2) * t + (2 * P0 - 5 * P1 + 4 * P2
- P3) * t ** 2 + (-P0 + 3 * P1 - 3 * P2 + P3) * t ** 3)
            yt = 0.5 * ((2 * Q1) + (-Q0 + Q2) * t + (2 * Q0 - 5 * Q1 + 4 * Q2
- Q3) * t ** 2 + (-Q0 + 3 * Q1 - 3 * Q2 + Q3) * t ** 3)
            spline_x.append(xt)
            spline_y.append(yt)
    return np.array(spline_x), np.array(spline_y)

# Функция для построения графика гармонических колебаний
def plot_f(points_count):
    fig.clear()
    x_points = np.linspace(0, 1.5, points_count)
    y_points = f(x_points)
    y_true = f(x_range)
    ax = fig.add_subplot(111)
    ax.plot(x_range, y_true, label="Гармонические колебания", color='cyan',
linewidth=1.5)
    ax.scatter(x_points, y_points, color='red')
    ax.legend()
    ax.set_title("Гармонические колебания")
    ax.set_xlabel("x")
    ax.set_ylabel("y")
    for i in table.get_children():
        table.delete(i)
    for x, y in zip(x_points, y_points):
        table.insert("", "end", values=(f"{x:.3f}", f"{y:.3f}"))
    canvas.draw()
```

```
# Функции для построения Catmull-Rom сплайна
def plot_catmull_rom_spline(points_count):
    fig.clear()
    x_points = np.linspace(0, 1.5, points_count)
    y_points = f(x_points)
    spline_x, spline_y = catmull_rom_spline(x_points, y_points)
    y_true = f(spline_x)
    error = calculate_error(y_true, spline_y)
    ax = fig.add_subplot(111)
    ax.plot(spline_x, spline_y, label="Catmull-Rom на основе функции", color
= 'orange', linestyle='--')
    ax.scatter(x_points, y_points, color='blue')
    ax.legend()
    ax.set_title(f"Catmull-Rom на основе функции\nОшибка восстановления:
{error:.{Z}f}")
    ax.set_xlabel("x")
    ax.set_ylabel("y")
    canvas.draw()

def plot_catmull_rom_spline_polynomial(points_count):
    fig.clear()
    x_points = np.linspace(0, 1.5, points_count)
    coefficients = np.polyfit(x_points, f(x_points), points_count - 1)
    polynomial = np.poly1d(coefficients)
    y_points = polynomial(x_points)
    spline = PchipInterpolator(x_points, y_points)
    y_spline = spline(x_range)
    y_true = f(x_range)
    error = calculate_error(y_true, y_spline)
    ax = fig.add_subplot(111)
    ax.plot(x_range, y_spline, label="Catmull-Rom на основе полинома",color =
'green', linestyle='--')
    ax.scatter(x_points, y_points, color='violet')
    ax.legend()
    ax.set_title(f"Catmull-Rom на основе полинома\nОшибка восстановления:
{error:.{Z}f}")
    ax.set_xlabel("x")
    ax.set_ylabel("y")
    canvas.draw()

# Создание интерфейса
root = tk.Tk()
root.title("Интерполяционная кривая Catmull-Rom")
root.geometry("800x800")

# График
fig = plt.Figure(figsize=(6, 4), dpi=100)
canvas = FigureCanvasTkAgg(fig, master=root)
canvas.get_tk_widget().pack(side=tk.TOP, fill=tk.BOTH, expand=1, pady=(0, 5))

# Таблица
table_frame = tk.Frame(root)
table_frame.pack(side=tk.TOP, fill=tk.BOTH, padx=10, pady=(0, 10))

table = ttk.Treeview(table_frame, columns=("X", "Y"), show='headings')
table.heading("X", text="X")
table.heading("Y", text="Y")
table.pack()

def button1_action():
    plot_f(points_count=N)

def button2_action():
```

```
    plot_catmull_rom_spline(points_count=N)

def button3_action():
    plot_catmull_rom_spline_polynomial(points_count=N)

# Панель с кнопками
button_frame = tk.Frame(root)
button_frame.pack(side=tk.BOTTOM, fill=tk.X, padx=10, pady=(0, 10))

# Кнопки
btn1 = tk.Button(button_frame, text="1 - e^(-x^5)", command=button1_action)
btn1.pack(side=tk.LEFT, padx=(0, 5))

btn2 = tk.Button(button_frame, text="Catmull-Rom на основе функции",
command=button2_action)
btn2.pack(side=tk.LEFT, padx=(0, 5))

btn3 = tk.Button(button_frame, text="Catmull-Rom на основе полинома",
command=button3_action)
btn3.pack(side=tk.LEFT)

# Запуск интерфейса
root.mainloop()
```

5. Скриншоты, иллюстрирующие результаты работы программы:

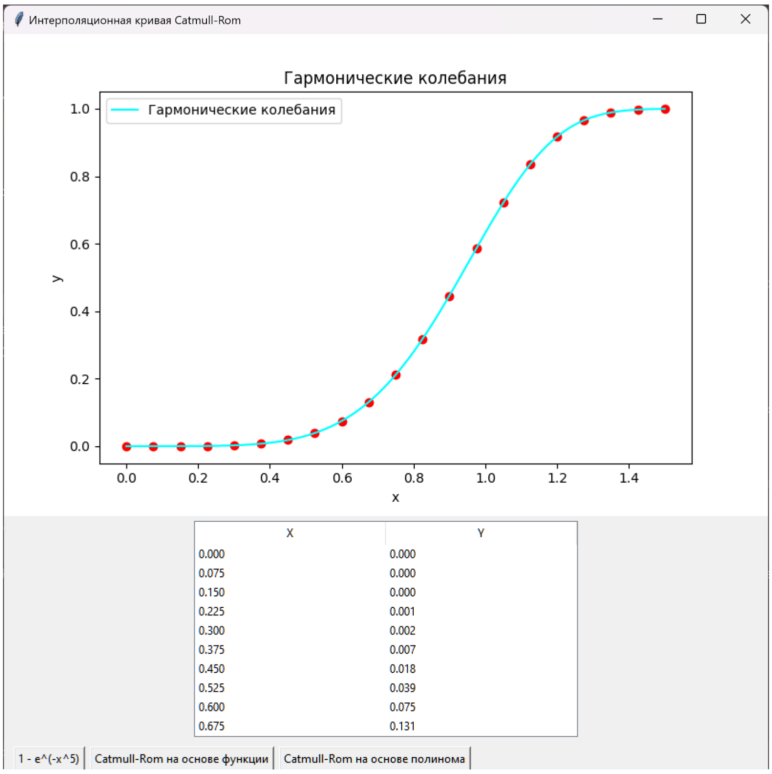

Рисунок 2.1 – Построение графика гармонических колебаний, 21 опорная точка

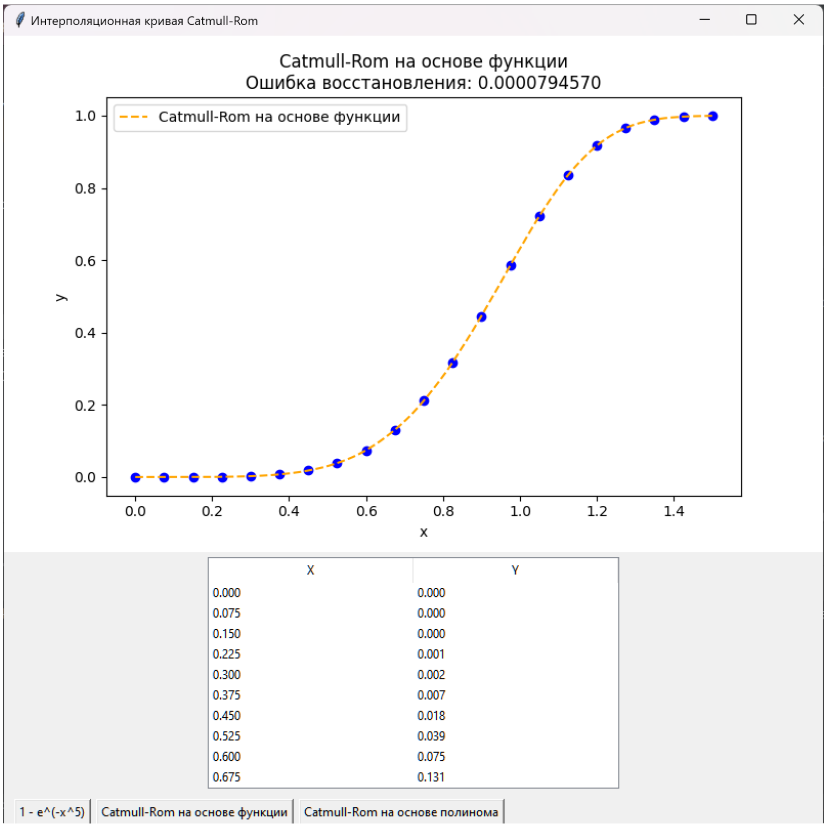

Рисунок 2.2 – График кривой Catmull-Rom на основе функции, 21 опорная точка

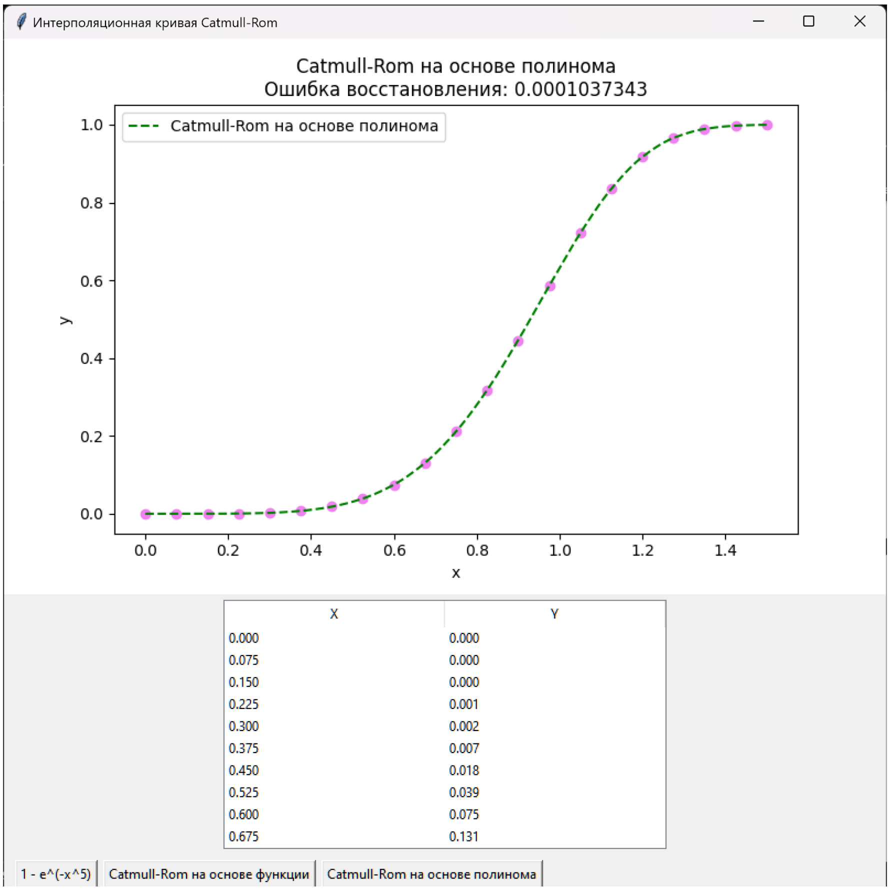

Рисунок 2.3 – График кривой на основе полинома, 21 опорная точка

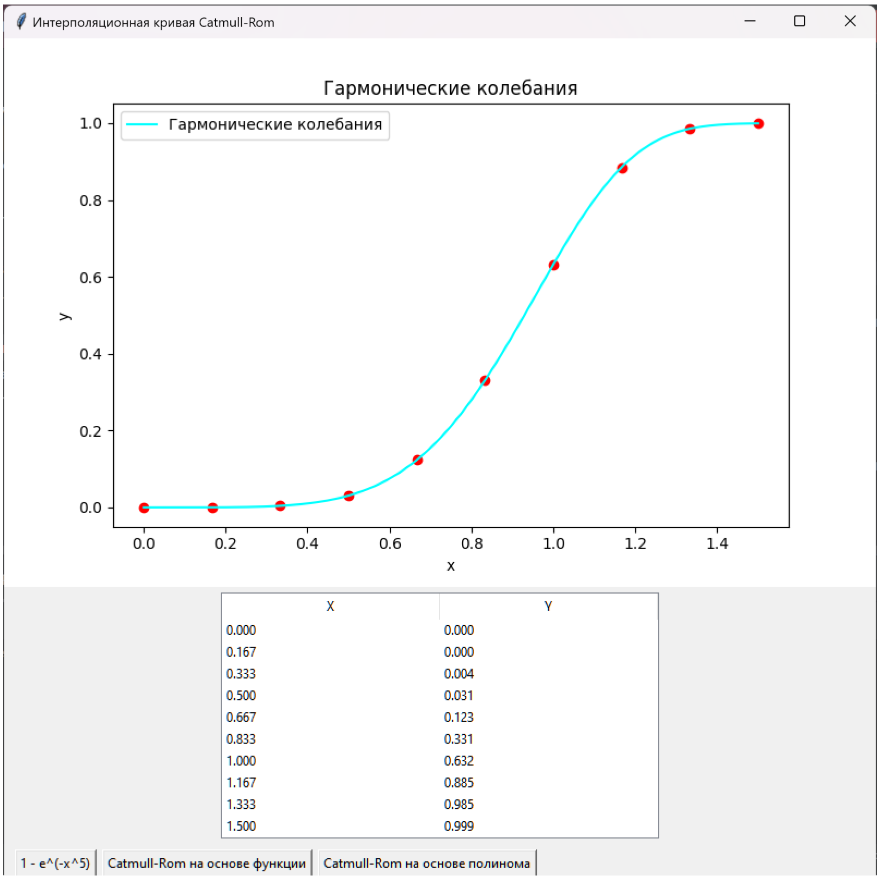

Рисунок 2.4 – Построение графика гармонических колебаний, 10 опорных точек

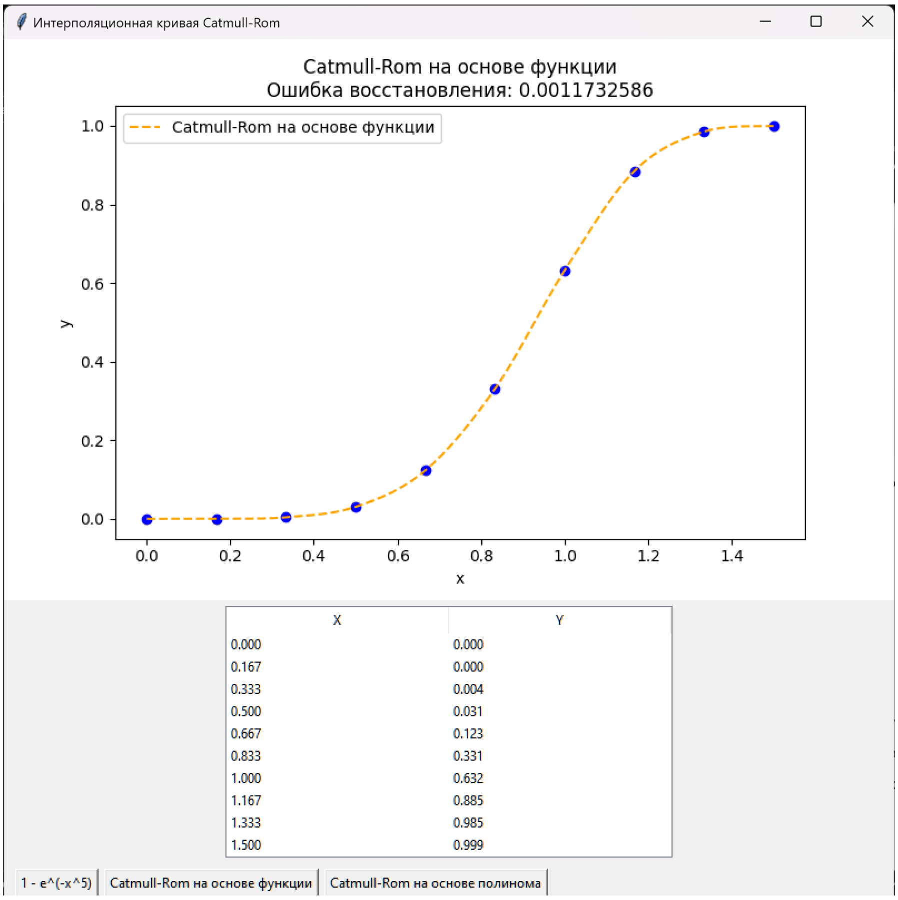

Рисунок 2.5 – График кривой Catmull-Rom на основе функции, 10 опорных точек

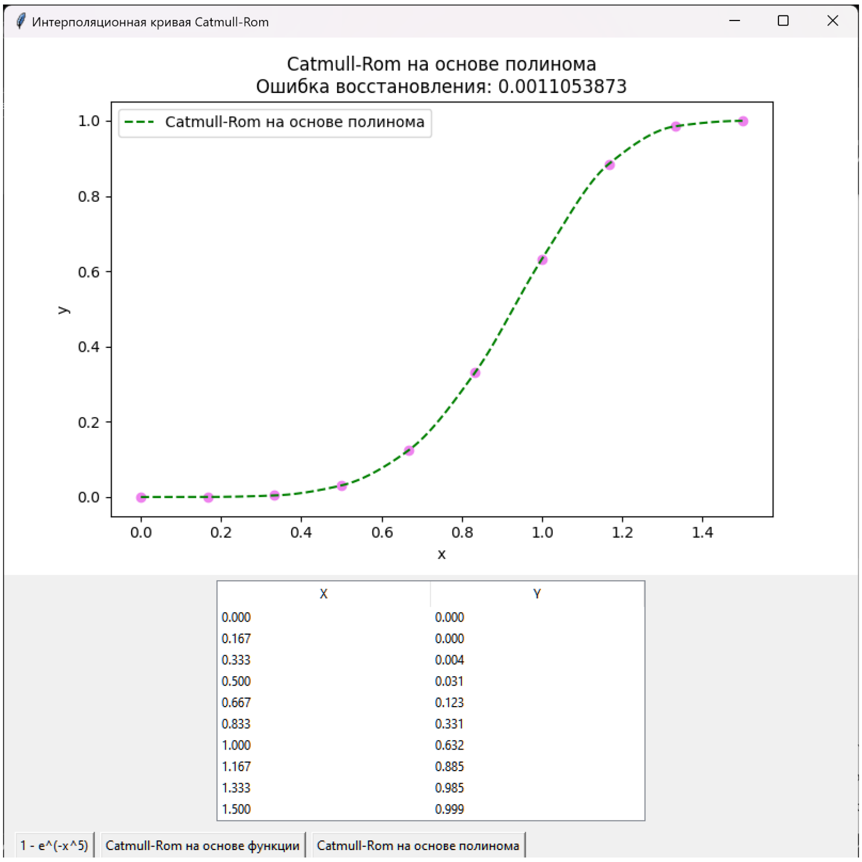

Рисунок 2.6 – График кривой на основе полинома, 10 опорных точек

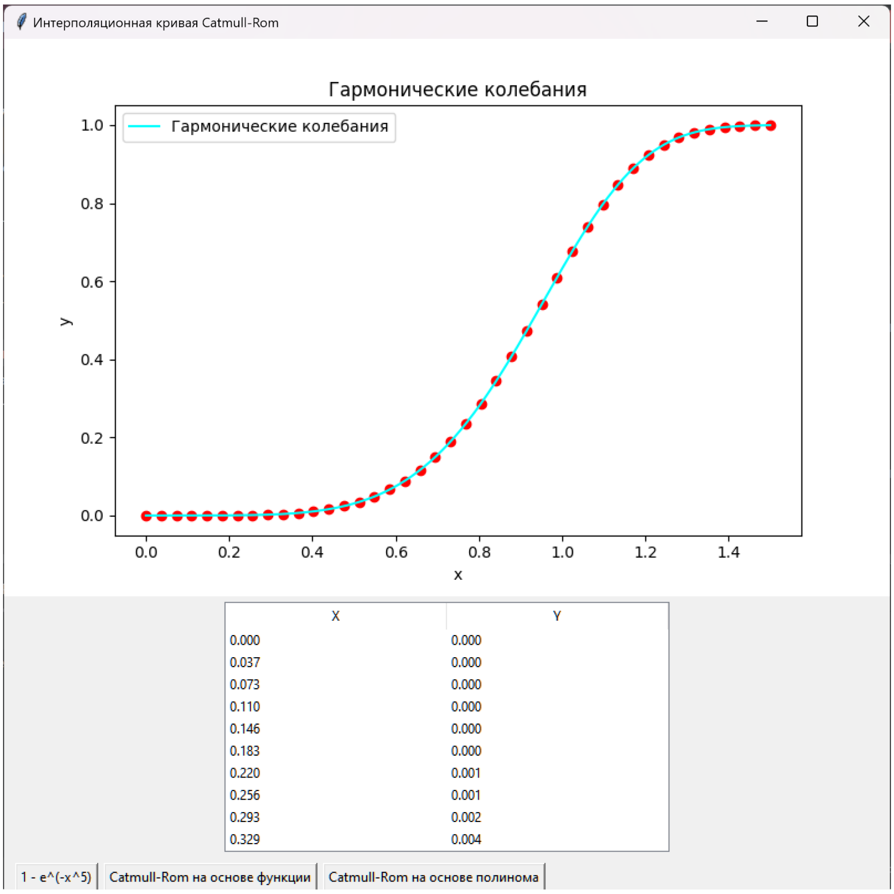

Рисунок 2.7 – Построение графика гармонических колебаний, 42 опорные точки

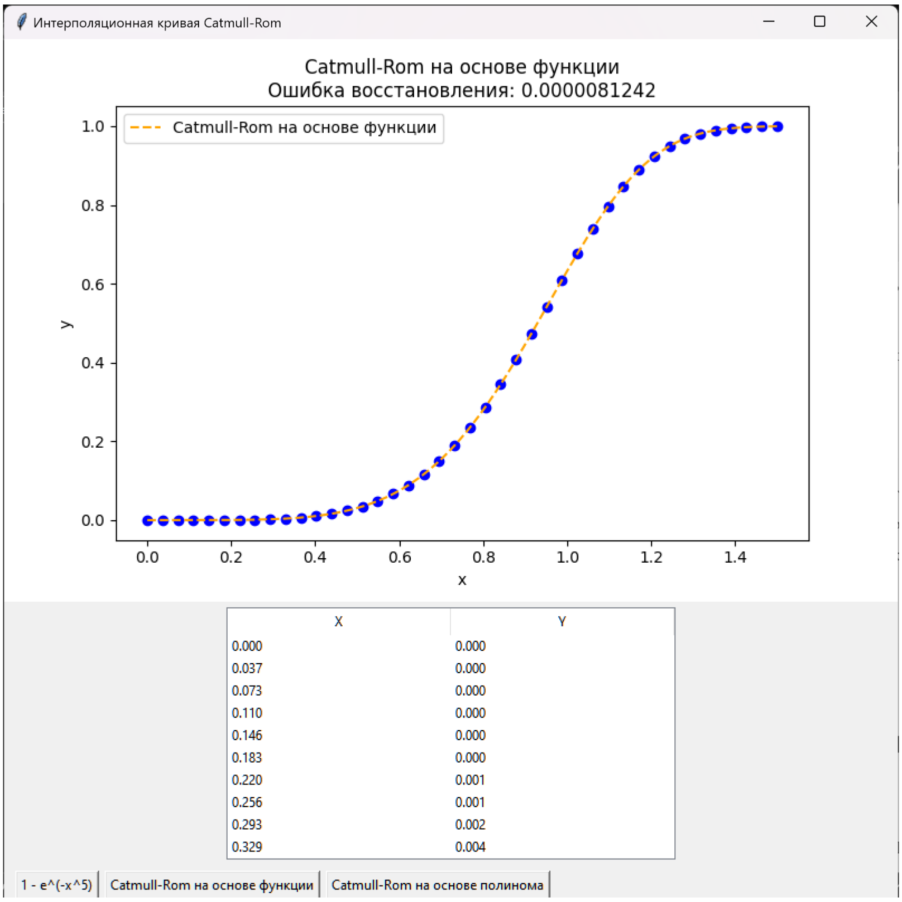

Рисунок 2.8 – График кривой Catmull-Rom на основе функции, 42 опорных точки

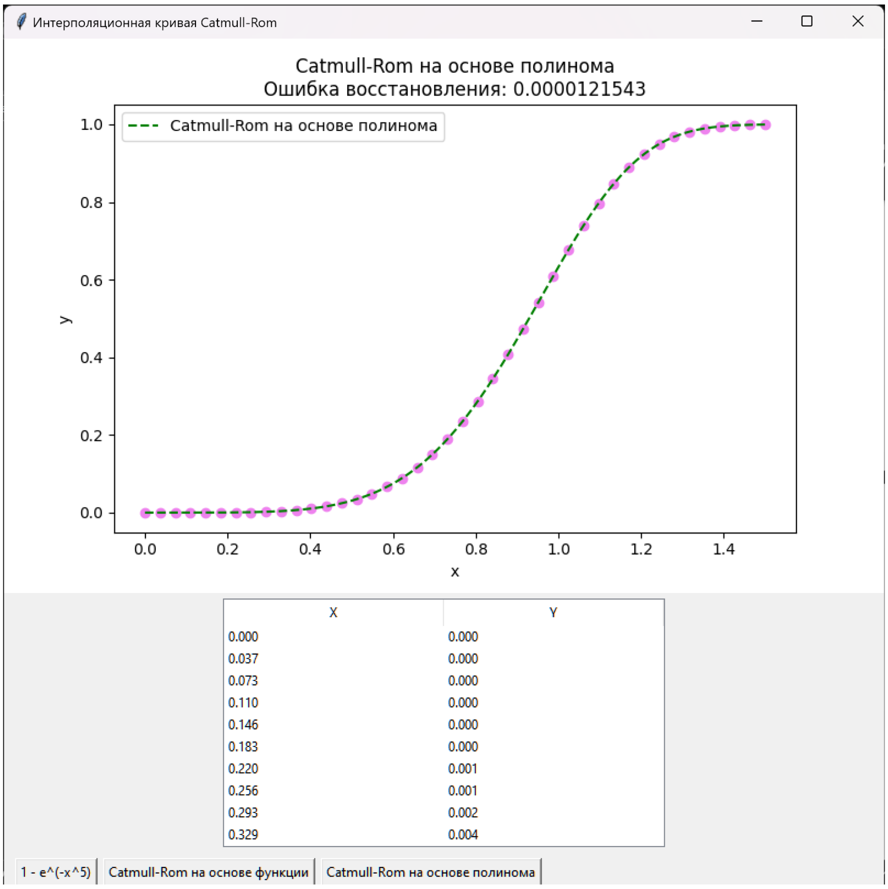

Рисунок 2.9 – График кривой на основе полинома, 42 опорных точки

6. Вывод:

В данной работе мы подробно рассмотрели, как создавать кривые, которые точно проходят через заданные точки, используя метод сплайнов. Мы также проанализировали способы построения сплайнов на основе исходной функции и её приближения многочленами.

Одним из ключевых элементов нашего исследования стала кривая Catmull-Rom, которая служит основой для создания плавных кривых, точно повторяющих заданные точки. Эта кривая нашла широкое применение в анимации, моделировании и других областях, где требуется безупречное прохождение через все необходимые точки.

Мы изучили два метода построения сплайнов: на основе исходной функции и её приближения многочленами, что позволило нам сравнить их и определить наиболее эффективный подход.

В ходе лабораторной было изучено, что кривая Catmull-Rom состоит из сегментов, что обеспечивает плавный переход от одной части к другой. Для построения сегмента кривой Catmull-Rom требуется четыре точки. Если добавить начальную и конечную точки, можно получить замкнутую кривую, что открывает новые горизонты для работы с замкнутыми траекториями.

Мы также освоили метод оценки точности приближения, используя среднюю абсолютную ошибку. Этот показатель позволяет понять, насколько точно кривая соответствует исходной функции, что имеет важное значение для достижения наилучших результатов.

Мы осознали, что полиномиальная аппроксимация может сделать кривую более гладкой, однако иногда она не всегда точно передаёт форму исходной функции. В зависимости от количества опорных точек, использование полиномиальной аппроксимации может помочь уменьшить ошибку, но если точек достаточно, то лучше использовать саму функцию для вычисления опорных точек, что обеспечит более точный и естественный результат.

Анализ ошибок восстановления:

Были проведены эксперименты с изменением количества опорных точек для построения кривой исходной функции и кривой Catmull-Rom с целью проанализировать точность и ошибки восстановления.

Для 21 точки ошибка восстановления для кривой Catmull-Rom на основе исходной функции составила 0,0000794570, в то время как для кривой на основе полинома 0,0001037343. Подобное число точек предоставляет улучшенную точность для обеих кривых и значительно снижает абсолютное отклонение. Разница между подходами не сильно заметная, так как чем больше точек, тем больше кривая Catmull-Rom на основе функции начинает более точно воспроизводить форму колебаний.

Для 10 точек ошибка восстановления для кривой Catmull-Rom на основе исходной функции составила 0,0011732586, в то время как для кривой на основе полинома 0,0011053873. Эти цифры говорят о том, что если точек немного, то полиномиальная аппроксимация может сгладить колебания и сделать меньше ошибку, чем интерполяция по точкам исходной функции. Но чем меньше точек, тем менее точно можно повторить форму исходной функции.

Для 42 точек ошибка восстановления для кривой Catmull-Rom на основе исходной функции составила 0,0000081242, а для кривой на основе полинома 0,0000121543. По сравнению с ошибкой восстановления для меньшего количества точек, подобные цифры являются мизерными. Значит для задач требующих высокой точности лучше использовать большое количество точек, а расчёты производить по исходной функции, хотя на меньшем количестве точек выигрышней смотрится точность для кривая на основе полинома.

В общем:

Исследование показало, что кривая Catmull-Rom — это хороший способ сделать линию плавной и точно пройти через заданные точки. Если точек мало, то лучше использовать полиномиальную аппроксимацию. Она сглаживает резкие изменения и уменьшает ошибки. Но если точек много, то лучше использовать саму функцию для вычисления значений, особенно на участках, где функция меняется.

В ходе работы мы узнали о методах интерполяции и аппроксимации функций, изучили кривую Catmull-Rom и научились оценивать ошибки восстановления.

Кривые Catmull-Rom полезны в компьютерной графике и моделировании, потому что они позволяют сделать линии плавными и точными. Это важно для создания анимации, построения траекторий и моделирования динамических систем.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Шикин Е.В., Плис Л.И. Кривые и поверхности на экране компьютера. Руководство по сплайнам для пользователей. М.: ДИАЛОГ-МИФИ, 1996. – 286 с.
2. Интерполяция – URL: https://help.fsight.ru/ru/mergedProjects/lib/03_transformations/uimodelling_interpolation.htm (дата обращения 10.11.2024)
3. Что такое сплайны – URL: http://cpu3d.com/lesson/chto-takoe-splayny/ (дата обращения 10.11.2024)
4. Что такое интерполяция и зачем она нужна? // Хабр. URL: https://habr.com/ru/articles/323442/ (дата обращения: 09.11.2024).