Министерство науки и высшего образования Российской Федерации Федеральное государственное автономное образовательное учреждение высшего образования «Национальный исследовательский университет ИТМО»

Факультет Программной инженерии и компьютерной техники

**­­­­­Лабораторная работа №5**

**по дисциплине «Вычислительная математика»**

**«Интерполяция функции»**

Вариант №3

Группа: P3212

Выполнил: Балин А. А.

Проверила: Наумова Н. А.

# Цель работы

Решить задачу интерполяции, найти значения функции при заданных значениях аргумента, отличных от узловых точек.

# Вычислительная реализация

Таблица по варианту:





Построим таблицу конечных разностей:



Так как и мы находимся в первой половине интервала, выбираем первую интерполяционную формулу Ньютона. В качестве берем

Коэффициенты:



<https://www.desmos.com/calculator/pio3rv4b7q?lang=ru>

Причём

Посчитаем методом Гаусса. Так как ближайшее к : ,

Следовательно, пользуемся второй формулой и выбираем коэффициенты:



<https://www.desmos.com/calculator/vmt2flbfhs?lang=ru>

Для сравнения

# Программная часть

## Код

import math

class Interpolation:

    def \_\_init\_\_(self, x,y):

        self.x = x

        self.y = y

        self.n = len(x)

        assert len(set(x)) == self.n and self.n==len(y), "x values must be unique"

        self.endless = None

    def L(self):

        # return string with the Lagrange polynomial

        L = []

        for i in range(self.n):

            l = [f"({self.y[i]})"]

            for j in range(self.n):

                if i != j:

                    l.append(f"(x-{self.x[j]})/({self.x[i]-self.x[j]})")

            L.append("\*".join(l))

        return "+".join(L)

    def g(self,i):

        n = len(i)

        if n==1:

            return self.y[i[0]]

        if n==2:

            return (self.y[i[1]]-self.y[i[0]])/(self.x[i[1]]-self.x[i[0]])

        else:

            return (self.g(i[1:])-self.g(i[:-1]))/(self.x[i[-1]]-self.x[i[0]])

    def endless\_delta(self):

        dy = []

        dy.append(self.y)

        while(len(dy[-1])!=1):

            ddy = []

            for i in range(len(dy[-1])-1):

                ddy.append((dy[-1][i+1]-dy[-1][i]))

            dy.append(ddy)

        self.endless = dy

    def N\_not\_same(self):

        N = [str(self.g([0]))]

        for i in range(1,self.n):

            coef = self.g(list(range(i+1)))

            l = [f"({coef})"]

            for j in range(i):

                l.append(f"(x-{self.x[j]})")

            N.append("\*".join(l))

        return "+".join(N)

    def N\_same(self):

        assert len(set([self.x[i]-self.x[i-1] for i in range(1,self.n)])) == 1, "dx must be the same for all i"

        self.endless\_delta()

        dy = self.endless

        h = self.x[1]-self.x[0]

        N = [str(self.y[0])]

        for i in range(1,self.n):

            coef = dy[i][0]/(math.factorial(i)\*h\*\*i)

            l = [f"({coef})"]

            for j in range(i):

                l.append(f"(x-{self.x[j]})")

            N.append("\*".join(l))

        return "+".join(N)

Изображение выглядит как текст, снимок экрана, Мультимедийное программное обеспечение, программное обеспечение

Автоматически созданное описание

Рисунок 1. Пример выполнения программы.

Изображение выглядит как текст, снимок экрана, Мультимедийное программное обеспечение, Графическое программное обеспечение

Автоматически созданное описание

Рисунок 2. Пример выполнения программы.

Изображение выглядит как текст, снимок экрана, Мультимедийное программное обеспечение, Графическое программное обеспечение

Автоматически созданное описание

Рисунок 3. Пример выполнения программы.

## Репозиторий с исходниками

https://github.com/ta4ilka69/docs\_for\_labs/tree/main/Вычмат

# Вывод

В ходе реализации данной лабораторной работы я ознакомился с различными методами интерполяции.