

Министерство высшего образования и науки Российской Федерации
Национальный научно-исследовательский университет ИТМО
Факультет программной инженерии и компьютерной техники

Лабораторная работа №5
по дисциплине
«Вычислительная математика»

Вариант 11

Работу выполнил:
Макеев Роман Ильич

Группа Р3208

Преподаватель:
Машина Екатерина Алексеевна

Санкт-Петербург

2024

Цель работы:

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

Вычислительная часть:

x_i	0.25	0.3	0.35	0.4	0.45	0.5	0.55
y_i	1.2557	2.1764	3.1218	4.0482	5.9875	6.9195	7.8359

$$X_1 = 0.255 \quad X_2 = 0.405$$

Таблица конечных разностей

y_i	Δy_i	$\Delta^2 y_i$	$\Delta^3 y_i$	$\Delta^4 y_i$	$\Delta^5 y_i$	$\Delta^6 y_i$
1.2557	0.9207	0.0247	-0.0437	1.0756	-4.1277	10.1917
2.1764	0.9454	-0.019	1.0319	-3.0521	6.064	
3.1218	0.9264	1.0129	-2.0202	3.0119		
4.0482	1.9393	-1.0073	0.9917			
5.9875	0.932	-0.0156				
6.9195	0.9164					
7.8359						

X_1 лежит в левой половине => используем первую формулу Ньютона

$$t = \frac{x - x_0}{h} = \frac{0.255 - 0.25}{0.05} = 0.1$$

$$\begin{aligned} N_6(x) &= y_0 + t\Delta y_0 + \frac{t(t-1)}{2!}\Delta^2 y_0 + \dots + \frac{t(t-1)(t-2)(t-3)(t-4)(t-5)}{6!}\Delta^6 y_0 = \\ &= 1.2557 + 0.09207 - 0.045 \cdot 0.0247 - 0.0285 \cdot 0.0437 - \\ &- 0.0207 \cdot 1.0756 - 0.016 \cdot 4.1277 - 0.01312 \cdot 10.1917 = 1.2334 \end{aligned}$$

X_2 лежит в правой половине => используем первую формулу Гаусса

$$t = \frac{x - a}{h} = \frac{0.405 - 0.4}{0.05} = 0.1$$

$$\begin{aligned} P_6(x) &= y_0 + t\Delta y_{-1} + \frac{t(t+1)}{2!}\Delta^2 y_{-1} + \dots + \frac{t(t+1)(t+2)(t+3)(t-1)(t-2)}{6!}\Delta^6 y_{-3} = \\ &= 4.0482 + 0.09264 + 0.0557 - 0.017 + \\ &+ 0.0264 - 0.0136 + 0.0173 = 4.2096 \end{aligned}$$

Листинг программы:

Расчет значения функции в точке Лагранжем

```
def lagrange_polynom(data: PointTable, x: float) -> float:
    result: float = 0

    n: int = len(data)
    for i in range(n):
        l: float = data[i].y
        for j in range(n):
            if i == j:
                continue

            l *= (x - data[j].x)
            l /= (data[i].x - data[j].x)
        result += l

    return result
```

Расчет разделенных разностей

```
def fill_nuton_diffs(data: PointTable) -> None:
    for i in range(len(data)):
        curr_diffs: list[float] = []
        for j in range(len(data) - i):
            if i == 0:
                curr_diffs.append(data[j].y)
            else:
                curr_diffs.append((nuton_diffs[-1][j+1] - nuton_diffs[-1][j])
/ (data[j+i].x - data[j].x))

        nuton_diffs.append(curr_diffs)
```

Расчет конечных разностей

```
def fill_nuton_diffs2(data: PointTable) -> None:
    for i in range(len(data)):
        curr_diffs: list[float] = []
        for j in range(len(data) - i):
            if i == 0:
                curr_diffs.append(data[j].y)
            else:
                curr_diffs.append(nuton_diffs2[-1][j+1] -
nuton_diffs2[-1][j])

        nuton_diffs2.append(curr_diffs)
```

Ньютон с разделенными разностями

```
def nuton_polinom1(data: PointTable, x: float) -> float:
    if len(nuton_diffs) == 0:
        fill_nuton_diffs(data)

    result: float = nuton_diffs[0][0]

    for i in range(1, len(data)):
        n: float = nuton_diffs[i][0]
        for j in range(i):
            n *= (x - data[j].x)
        result += n

    return result
```

Ньютон с конечными разностями

```
def nuton_polinom2(data: PointTable, x: float) -> float:
    h: float = (data[-1].x - data[0].x) / (len(data) - 1)
    for i in range(1, len(data)):
        if round(data[i].x - data[i - 1].x, 5) != round(h, 5):
            return None

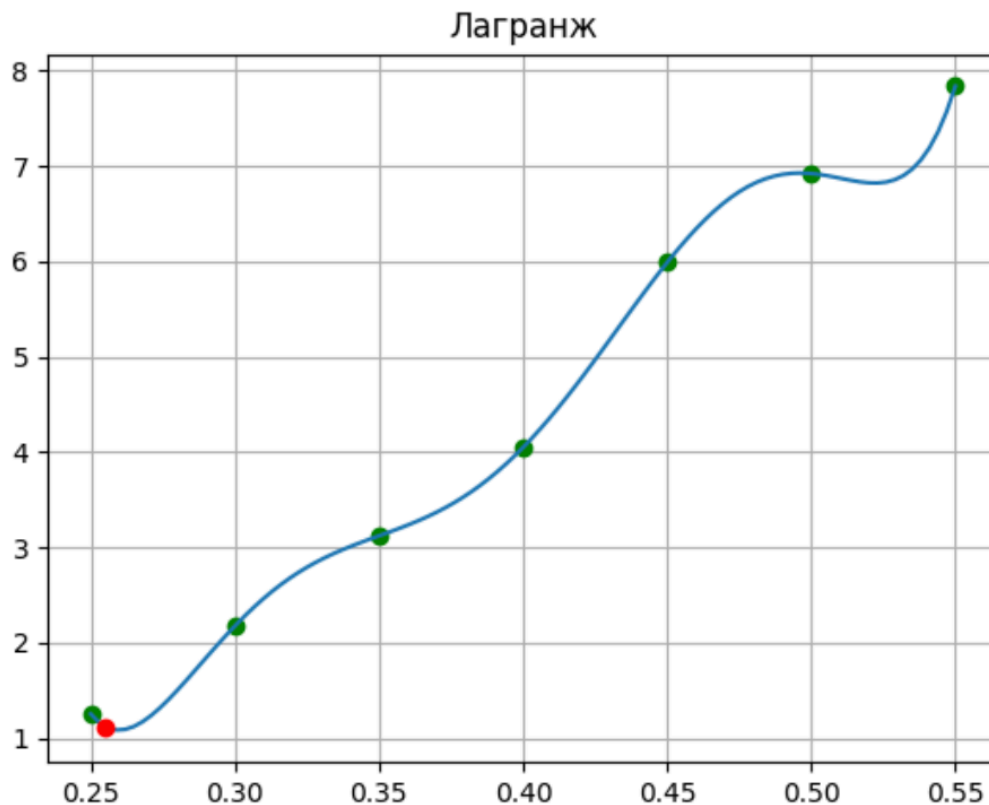
    if len(nuton_diffs2) == 0:
        fill_nuton_diffs2(data)

    result: float = nuton_diffs2[0][0]
    factorials: list[int] = [1, 1]
    for i in range(1, len(data)):
        factorials.append(i * factorials[-1])
        n: float = nuton_diffs2[i][0] / (factorials[-1] * h ** i)
        for j in range(i):
            n *= (x - data[j].x)

        result += n

    return result
```

Результаты работы программы:



```
1: Console
2: File
3: Function
Choose input type: [1/2/3] -> 2
Type filename ->
Input x -> 0.255
Ответ методом "Лагранж": 1.1225
Ответ методом "Ньютон с разделенными разностями": 1.1225
Ответ методом "Ньютон с конечными разностями": 1.1225
```

Выводы:

В ходе работы была написана программа, интерполирующая функцию по заданным точкам методами Лагранжа и Ньютона для нахождения значения функции в точке отличной от заданных