МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РФ

Федеральное государственное автономное
образовательное учреждение высшего образования
«Национальный исследовательский университет ИТМО»

ФАКУЛЬТЕТ ПРОГРАММНОЙ ИНЖЕНЕРИИ И КОМПЬЮТЕРНОЙ ТЕХНИКИ

ЛАБОРАТОРНАЯ РАБОТА №5

по дисциплине

‘ВЫЧИСЛИТЕЛЬНАЯ МАТЕМАТИКА’

Вариант №8

Выполнила:
Конаныхина Антонина
P3215
Преподаватель:
Малышева Татьяна
Алексеевна

Санкт-Петербург, 2022

Цель работы:

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

- многочлен Лагранжа;
- многочлен Ньютона;
- многочлен Гаусса.

Задание:

<u>Обязательное задание (до 80 баллов)</u>

1. Вычислительная реализация задачи:
   1.1Используя первую или вторую интерполяционную формулу Ньютона, первую или вторую интерполяционную формулу Гаусса вычислить значения функции при данных значениях аргумента (для значения X_1 и X_2, см. табл. 1 - 4).
   1.2 Построить таблицу конечных разностей.
   1.3 Подробные вычисления привести в отчете.

2. Программная реализация задачи:
   2.1. Исходные данные задаются в виде: а) набора данных (таблицы x,y), б) на основе выбранной функции (например, $\sin x$).
   2.2. Вычислить приближенное значение функции для заданного значения аргумента, введенного с клавиатуры, указанными методами (см. табл.5).
   2.3. Построить графики заданной функции с отмеченными узлами интерполяции и интерполяционного многочлена Ньютона/Гаусса (разными цветами).

<u>Необязательное задание (до 20 баллов)</u>

Реализовать в программе вычисление значения функции для заданного значения аргумента, введенного с клавиатуры, используя схемы Стирлинга и Бесселя.

Рабочие формулы используемых методов

Формула для полинома Лагранжа:

$$Ln(x) = \sum_{i=0}^{n} y_i \frac{(x-x_0)(x-x_1)\dots(x-x_{i-1})(x-x_{i+1})\dots(x-x_n)}{(x_i-x_0)(x_i-x_1)\dots(x_i-x_{i-1})(x_i-x_{i+1})\dots(x_i-x_n)}$$

Формула Ньютона для неравноостоящих узлов:

$$N_n(x) = f(x_0) + \sum_{k=1}^{n} f(x_0, x_1, \dots, x_k) \prod_{j=0}^{k-1} (x - x_j)$$,

где $$f(x_i, x_{i+1}, \dots, x_{i+k}) = \frac{f(\,x_{i+1}, \dots, x_{i+k}) - f(x_i, x_{i+k-1})}{x_{i+k} - x_i}$$ - разделённые разности.

Вторая интерполяционная формула Ньютона для интерполирования назад для равноотстоящих узлов:

$$\boldsymbol{N_n(x) = y_n + t\Delta y_{n-1} + \frac{t(t+1)}{2!}\Delta^2 y_{n-2} + \dots + \frac{t(t+1)\dots(t+n-1)}{n!}\Delta^n y_0}$$

Где $t = \frac{x-x_0}{h}$ и $\Delta^k y_i = \Delta^{k-1} y_{i+1} - \Delta^{k-1} y_i$

Формула интерполяционного полинома Гаусса для x > a:

$$\begin{aligned} P_n(x) = {} & y_0 + t\Delta y_0 + \frac{t(t-1)}{2!}\Delta^2 y_{-1} + \frac{(t+1)t(t-1)}{3!}\Delta^3 y_{-1} \\ & + \frac{(t+1)t(t-1)(t-2)}{4!}\Delta^4 y_{-2} \\ & + \frac{(t+2)(t+1)t(t-1)(t-2)}{5!}\Delta^5 y_{-2} \dots \\ & + \frac{(t+n-1)\dots(t-n+1)}{(2n-1)!}\Delta^{2n-1} y_{-(n-1)} \\ & + \frac{(t+n-1)\dots(t-n)}{(2n)!}\Delta^{2n} y_{-n} \end{aligned}$$

Формула интерполяционного полинома Стирлинга:

$$\begin{aligned} P_n(x) = {} & y_0 + t\frac{\Delta y_{-1} + \Delta y_0}{2!} + \frac{t^2}{2}\Delta^2 y_{-1} + \frac{t(t^2-1^2)}{3!} \cdot \frac{\Delta^3 y_{-2} + \Delta^3 y_{-1}}{2} \\ & + \frac{t^2(t^2-1^2)}{4!}\Delta^4 y_{-2} + \dots + \frac{t(t^2-1^2)(t^2-2^2)\dots(t^2-(n-1)^2)}{(2n)!} \\ & \cdot \frac{\Delta^{2n-1} y_{-n} + \Delta^{2n-1} y_{-(n-1)}}{2} \\ & + \frac{t^2(t^2-1^2)(t^2-2^2)\dots(t^2-(n-1)^2)}{(2n)!}\Delta^{2n} y_{-n} \end{aligned}$$

где $t = \frac{x-x_0}{h}$ и $\Delta^k y_i = \Delta^{k-1} y_{i+1} - \Delta^{k-1} y_i$

Формула интерполяционного полинома Бесселя:

$$\begin{aligned} P_n(x) = {} & \frac{y_0 + y_{-1}}{2}{}_0 + \left(t - \frac{1}{2}\right)\Delta y_0 + \frac{t(t-1)}{2!}\frac{\Delta^2 y_{-1} + \Delta^2 y_0}{2} + \frac{\left(t-\frac{1}{2}\right)t(t-1)}{3!}\Delta^3 y_{-1} \\ & + \frac{t(t-1)(t+1)(t-2)}{4!}\frac{\Delta^4 y_{-2} + \Delta^2 y_{-1}}{2} + \frac{\left(t-\frac{1}{2}\right)t(t-1)(t+1)(t-2)}{5!}\Delta^5 y_{-2} \\ & + \frac{t(t-1)(t+1)(t-2)(t+2)(t+3)}{6!}\frac{\Delta^6 y_{-3} + \Delta^6 y_{-2}}{2} + \dots \\ & + \frac{t(t-1)(t+1)(t-2)(t+2)\dots(t-n)(t+n-1)}{(2n)!} \cdot \frac{\Delta^{2n} y_{-n} + \Delta^{2n} y_{-(n-1)}}{2} \\ & + \frac{\left(t-\frac{1}{2}\right)t(t-1)(t+1)(t-2)(t+2)\dots(t-n)(t+n-1)}{(2n+1)!}\Delta^{2n+1} y_{-n} \end{aligned}$$

где $t = \frac{x-x_0}{h}$ и $\Delta^k y_i = \Delta^{k-1} y_{i+1} - \Delta^{k-1} y_i$

Вычислительная реализация задачи:

Метод Ньютона

x	y	Δy	Δ^2y	Δ^3y	Δ^4y	Δ^5y	Δ^6y
1,05	0,1213	1,0103	0,004	-0,0077	0,0014	0,0391	-0,1478
1,15	1,1316	1,0143	-0,0037	-0,0063	0,0405	-0,1087	
1,25	2,1459	1,0106	-0,01	0,0342	-0,0682		
1,35	3,1565	1,0006	0,0242	-0,034			
1,45	4,1571	1,0248	-0,0098				
1,55	5,1819	1,015					
1,65	6,1969						

$$h = 0{,}1$$

$$t = \frac{x - x_0}{0{,}1}$$

Посчитаем для x = 1,562

$$t = \frac{1{,}562 - 1{,}05}{0{,}1} = 5{,}12$$

$$\begin{aligned}
y(1{,}562) = 0{,}1213 + 5{,}12 * 1{,}0103 + \frac{5{,}12 * (5{,}12 - 1)}{2!} 0{,}004 \\
+ \frac{5{,}12 * (5{,}12 - 1) * (5{,}12 - 2)}{3!} (-0{,}0077) \\
+ \frac{5{,}12(5{,}12 - 1)(5{,}12 - 2)(5{,}12 - 3)}{4!} (0{,}0014) \\
+ \frac{5{,}12(5{,}12 - 1)(5{,}12 - 2)(5{,}12 - 3)(5{,}12 - 4)}{5!} 0{,}0391 \\
+ \frac{5{,}12(5{,}12 - 1)(5{,}12 - 2)(5{,}12 - 3)(5{,}12 - 4)(5{,}12 - 5)}{6!} (-0{,}1478) = \\
= 5{,}30697
\end{aligned}$$

Посчитаем для x = 1,362:

$$t = \frac{1{,}362 - 1{,}05}{0{,}1} = 3{,}12$$

$$
\begin{aligned}
y(1{,}362) = 0{,}1213 + 3{,}12 * 1{,}0103 + \frac{3{,}12(3{,}12-1)}{2!}0{,}004 \\
+ \frac{3{,}12(3{,}12-1)(3{,}12-2)}{3!}(-0{,}0077) \\
+ \frac{3{,}12(3{,}12-1)(3{,}12-2)(3{,}12-3)}{4!}(0{,}0014) \\
+ \frac{3{,}12(3{,}12-1)(3{,}12-2)(3{,}12-3)(3{,}12-4)}{5!}0{,}0391 \\
+ \frac{3{,}12(3{,}12-1)(3{,}12-2)(3{,}12-3)(3{,}12-4)(3{,}12-5)}{6!}(-0{,}1478) = \\
= 3{,}27665
\end{aligned}
$$

Метод Гаусса

i	x	y	Δy	Δ^2y	Δ^3y	Δ^4y	Δ^5y	Δ^6y
-3	1,05	0,1213	1,0103	0,004	-0,0077	0,0014	0,0391	-0,1478
-2	1,15	1,1316	1,0143	-0,0037	-0,0063	0,0405	-0,1087	
-1	1,25	2,1459	1,0106	-0,01	0,0342	-0,0682		
0	1,35	3,1565	1,0006	0,0242	-0,034			
1	1,45	4,1571	1,0248	-0,0098				
2	1,55	5,1819	1,015					
3	1,65	6,1969						

$h = 0{,}1$

$$t = \frac{x - x_0}{0{,}1}$$

Посчитаем для x = 1,562:

$$t = \frac{1{,}562 - 1{,}35}{0{,}1} = 2{,}12$$

$$
\begin{aligned}
y(1{,}562) = 3{,}1565 + 2{,}12 * 1{,}0006 + \frac{2{,}12(2{,}12-1)}{2!}(-0{,}01) \\
+ \frac{(2{,}12+1)2{,}12(2{,}12-1)}{3!}0{,}0342 \\
+ \frac{(2{,}12+1)2{,}12(2{,}12-1)(2{,}12-2)}{4!}0{,}0405 \\
+ \frac{(2{,}12+2)(2{,}12+1)2{,}12(2{,}12-1)(2{,}12-2)}{5!}(-0{,}1087) \\
+ \frac{(2{,}12+2)(2{,}12+1)2{,}12(2{,}12-1)(2{,}12-2)(2{,}12-3)}{6!}(-0{,}1478) \\
= 5{,}30697
\end{aligned}
$$

Посчитаем для x = 1,362

$$t = \frac{1{,}362 - 1{,}35}{0{,}1} = 0{,}12$$

$$\begin{aligned} y(1{,}362) = 3{,}1565 + 0{,}12 * 1{,}0006 + \frac{0{,}12(0{,}12-1)}{2!}(-0{,}01) \\ + \frac{(0{,}12+1)0{,}12(0{,}12-1)}{3!}0{,}0342 \\ + \frac{(0{,}12+1)0{,}12(0{,}12-1)(0{,}12-2)}{4!}0{,}0405 \\ + \frac{(0{,}12+2)(0{,}12+1)0{,}12(0{,}12-1)(0{,}12-2)}{5!}(-0{,}1087) \\ + \frac{(0{,}12+2)(0{,}12+1)0{,}12(0{,}12-1)(0{,}12-2)(0{,}12-3)}{6!}(-0{,}1478) = \\ = 3{,}27665 \end{aligned}$$

Листинг программы

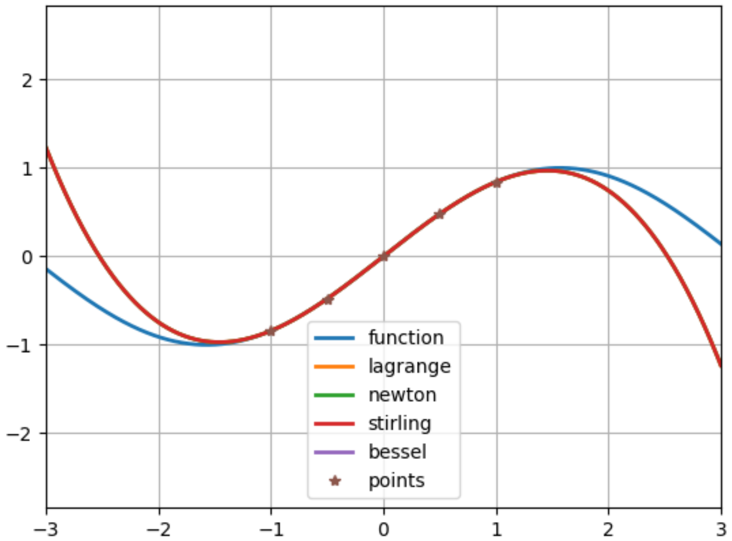
```
import math


def lagrange_method(x_arr, y_arr, x):
    l = 0
    for j in range(len(y_arr)):
        numerator = 1
        denominator = 1
        for i in range(len(x_arr)):
            if i != j:
                numerator = numerator * (x - x_arr[i])
                denominator = denominator * (x_arr[j] - x_arr[i])
        l = l + y_arr[j] * numerator / denominator
    return l


def get_points_from_func(f, a, b, step):
    x = []
    y = []
    x_now = a
    while x_now <= b:
        x.append(x_now)
        y.append(f(x_now))
        x_now += step
    return x, y

def newton_method(x_arr, y_arr, x):
    diff = [y_arr]
    for i in range(len(x_arr)):
        tmp_dif = []
        for j in range(len(x_arr) - i - 1):
            tmp_dif.append((diff[-1][j + 1] - diff[-1][j])/(x_arr[j + i + 1]
- x_arr[j]))
        diff.append(tmp_dif)
    mul = 1
    answer = y_arr[0]
    for i in range(len(x_arr)-1):
        mul *= (x - x_arr[i])
        answer += diff[i + 1][0] * mul
```

```
    return answer

#конечные разности
def final_differences(y_values):
    final_diff = [y_values]
    for i in range(len(y_values)-1):
        temp_fin_dif = []
        for j in range(len(final_diff[i]) - 1):
            diff = final_diff[i][j + 1] - final_diff[i][j]
            temp_fin_dif.append(diff)
        final_diff.append(temp_fin_dif)
    return final_diff


def create_factorial(n):
    result = []
    for i in range(n+1):
        fact = 1
        for j in range(1, i+1):
            fact *= j
        result.append(fact)
    return result


def stirling_method(x_arr, y_arr, x):
    if len(y_arr) % 2 == 0:
        return
    h = x_arr[1] - x_arr[0]
    #проверка, что все остальные разности одинаковые
    for i in range(len(y_arr)-1):
        if (x_arr[i+1] - x_arr[i]) != h:
            return
    fin_diff = final_differences(y_arr)
    fact = create_factorial(len(y_arr))
    mid = len(y_arr)//2
    t = (x - x_arr[mid]) / h
    result = y_arr[mid]

    for n in range(1, mid + 1):
        mul = 1
        for j in range(1, n):
            mul *= (t * t - j * j)
        result += 1 / fact[2 * n - 1] * t * mul * (fin_diff[2 * n - 1][-(n -
1) + mid] + fin_diff[2 * n - 1][-n + mid]) / 2
        result += 1 / fact[2 * n] * (t ** 2) * mul * (fin_diff[2 * n][-n +
mid])
    return result

def bessel_method(x_arr, y_arr, x):
    if len(y_arr) % 2 == 1:
        return
    h = x_arr[1] - x_arr[0]
    for i in range(len(y_arr)-1):
        if (x_arr[i+1] - x_arr[i]) != h:
            return
    fin_diff = final_differences(y_arr)
    fact = create_factorial(len(y_arr))
    mid = (len(y_arr) - 2) // 2
    t = (x - x_arr[mid]) / h
    result = 0

    for n in range(0, mid + 1):
        mul = 1
        for j in range(1, n + 1):
```

```
            mul *= (t - j)*(t + j - 1)
        result += (1 / fact[2 * n]) * mul * (fin_diff[2 * n][-n + mid] +
fin_diff[2 * n][-(n - 1) + mid]) / 2
        result += (1 / fact[2 * n + 1]) * (t - (1 / 2)) * mul * (fin_diff[2 *
n + 1][-n + mid])
    return result
```

Результаты выполнения программы:

```
Ведите источник точек. Для файла - 1, из функции - 2: 2
Введите границы интервала для набора точек
-1
1
Введите шаг:
0.5
Введите X:
-2
Методом Лагранжа:  -0.7439012304117165
Методом Ньютона:  -0.743901230411717
Методом Стирлинга:  -0.743901230411717
Методом Бесселя:  None
```



Вывод:

В результате выполнения данной лабораторной работы я познакомилась с различными методами интерполяции функции и реализовала их на языке программирования Python. Все реализованные методы выдавали одинаковые результаты, с небольшой погрешностью. Наименее точным оказался метод Ньютона, но, думаю, это связано с особенностями реализации и погрешностью вычислений.