Національний університет «Львівська політехніка»
Інститут комп’ютерних наук та інформаційних технологій

ЗВІТ
Про виконання лабораторної роботи № 8
«Наближення дискретних (таблично заданих) функцій»
з дисципліни «Чисельні методи»

Лектор:
доцент кафедри ПЗ
Мельник Н.Б.

Виконав:
студ. групи ПЗ-15
Марущак А. С.

Прийняв:
асистент кафедри ПЗ
Гарматій Г.Ю
«___» ______ 2022 р.
∑ = ____ ___________

Львів – 2022

Тема роботи: Наближення дискретних (таблично заданих) функцій.

Мета роботи: ознайомитися з методом інтерполяції таблично заданих функцій.

Теоретичні відомості

Найпростіша задача інтерполяції полягає в тому, що на відрізку [a,b] задано (n +1) точок $x_0, x_1, \dots, x_n$, які називають вузлами інтерполяції, і значення деякої функції у цих точках

$$f(x_0) = y_0, \quad f(x_1) = y_1, \quad \dots, \quad f(x_n) = y_n.$$

Необхідно побудувати інтерполяційну функцію F(x) , яка приймає у вузлах інтерполяції ті самі значення, що й функція f (x) . Тобто треба знайти таку функцію F(x) , щоб

$$F(x_0) = y_0, \quad F(x_1) = y_1, \quad \dots, \quad F(x_n) = y_n.$$

Геометрично це означає, що треба знайти криву y = F(x) певного типу, яка проходить через задану систему точок.

Зауважимо, що через задану множину точок можна провести безліч гладких кривих. Тому задача інтерполяції є неоднозначною. Вона стає однозначною тоді, коли за інтерполяційну функцію вибрати поліном $P_n(x)$ n-го степеня, де степінь полінома на одиницю менший від к-ті вузлів інтерполяції, і такий, що виконуються умови

$$P_n(x_0) = y_0, \quad P_n(x_1) = y_1, \quad \dots, \quad P_n(x_n) = y_n.$$

Інтерполяційний поліном Лагранжа

Один з методів знаходження інтерполяційного полінома запропонував Лагранж. Основна ідея цього методу полягає в пошуку полінома, який в одному довільному вузлі інтерполяції приймає значення один, а в усіх інших вузлах – нуль.

Наближену функцію y = F(x) розглянемо у вигляді

$$F(x) = L_n(x) = \sum_{i=0}^{n} P_i(x) f(x_i),$$

де $P_i(x)$ - такий многочлен, що

$$P_i(x_j) = \begin{cases} 0, & i \neq j, \\ 1, & i = j, \end{cases} \qquad i, j = \overline{0, n}.$$

Оскільки точки $x_0, x_1, \dots, x_n$ є коренями полінома, то його можна записати у такому вигляді

$$P_i(x)=\frac{(x-x_0)(x-x_1)\dots(x-x_{i-1})(x-x_{i+1})\dots(x-x_n)}{(x_i-x_0)(x_i-x_1)\dots(x_i-x_{i-1})(x_i-x_{i+1})\dots(x_i-x_n)},$$

а наближена функція F(x) , яку називають інтерполяційним многочленом Лагранжа, матиме вигляд

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

Для запису інтерполяційного полінома Лагранжа зручно використовувати таблицю:

$\Pi_{n+1}(x)$					D_i	y_i
$x-x_0$	x_0-x_1	x_0-x_2	...	x_0-x_n	D_0	y_0
x_1-x_0	$x-x_1$	x_1-x_2	...	x_1-x_n	D_1	y_1
x_2-x_0	x_2-x_1	$x-x_2$	...	x_2-x_n	D_2	y_2
...	...	...	...	...	...	...
x_n-x_0	x_n-x_1	x_n-x_2	...	$x-x_n$	D_n	y_n

Тут D_i – добуток елементів і –го рядка, $\Pi_{n+1}(x)$ – добуток елементів головної діагоналі.

Тоді поліном Лагранжа можна записати у вигляді

$$L_n(x)=\Pi_{n+1}(x)\sum_{i=0}^{n}\frac{y_i}{D_i}.$$

У випадку рівновіддалених вузлів вираз набуде форми

$$L_n(x)=\frac{1}{n!}\Pi_{n+1}(t)\sum_{i=0}^{n}(-1)^{n-i}\frac{C_n^i}{t-i}y_i,$$

де t – крок інтерполяції.

Інтерполяційний поліном Ньютона

Інший спосіб розв’язування задачі інтерполяції запропонував Ньютон. Цей спосіб полягає в тому, що поліном $P_n(x)$ для загального випадку нерівновіддалених вузлів записують у вигляді

$$P_n(x)=f(x_0)+f(x_0,x_1)(x-x_0)+f(x_0,x_1,x_2)(x-x_0)(x-x_1)+$$
$$+\dots+f(x_0,x_1,\dots,x_n)(x-x_0)(x-x_1)\dots(x-x_{n-1}),$$

де

$f(x_0, x_1) = \frac{f(x_1)-f(x_0)}{x_1-x_0}$ – розділена різниця 1-го порядку,

$f(x_0, x_1, x_2) = \frac{f(x_1,x_2)-f(x_0,x_1)}{x_2-x_0}$ – розділена різниця 2-го порядку,

$f(x_0, x_1, \dots, x_n) = \frac{f(x_1,\dots,x_n)-f(x_0,\dots,x_{n-1})}{x_n-x_0}$ – розділена різниця n-го порядку.

Для випадку рівновіддалених вузлів маємо вираз

$$P_n(x) = f(x_0) + \frac{\Delta f(x_0)}{1!h}(x-x_0) + \frac{\Delta^2 f(x_0)}{2!h^2}(x-x_0)(x-x_1) +$$

$$+ \dots + \frac{\Delta^n f(x_0)}{n!h^n}(x-x_0)(x-x_1)\dots(x-x_{n-1}).$$

де $\Delta f(x_0)$ – скінченна різниця першого порядку і обчислюється наступним чином

$$\Delta f(x_i) = f(x_{i+1}) - f(x_i).$$

Для обчислення скінченної різниці другого порядку використовуємо скінченні різниці першого порядку

$$\Delta^2 f(x_i) = \Delta(\Delta f(x_i)) = \Delta f(x_{i+1}) - \Delta f(x_i).$$

Аналогічно запишемо скінченну різницю n -го порядку

$$\Delta^n f(x_i) = \Delta(\Delta^{n-1} f(x_i)) = \Delta^{n-1} f(x_{i+1}) - \Delta^{n-1} f(x_i)$$

Індивідуальне завдання

1. Ознайомитися з теоретичним матеріалом.
2. Використовуючи інтерполяційні поліноми Лагранжа та Ньютона, обчислити значення табличної заданої функції у точці x_0 .

Варіант завдання

x	0,115	0,120	0,125	0,130	0,135	0,140	0,145	0,150	0,165	0,170
y	8,68	8,29	7,96	7,65	7,36	7,10	6,85	6,62	6,40	6,20

$x_0 = 0,122$

Хід роботи

Проведемо деякі обчилення вручну для кращого розуміння вивченого матеріалу.

По-перше, як бачимо, вузли не є рівновіддаленими, тому знаходити поліном будемо як для загального випадку.

По-друге, як бачимо, ми маємо 10 точок, тому інтерполяційні поліноми Лагранжа та Ньютона будуть 9-го степеня.

Поліном Лагранжа представлятиме собою суму 10 доданків, кожен з яких – добуток коефіцієнта Лагранжа на значення функції в певній точці. Наприклад, для першої точки цей доданок матиме вигляд

$$\frac{(x-0.12)(x-0.125)(x-0.13)(x-0.135)(x-0.14)(x-0.145)(x-0.150)(x-0.165)(x-0.170)}{(-0.005)\cdot(-0.01)\cdot(-0.015)\cdot(-0.02)\cdot(-0.025)\cdot(-0.03)\cdot(-0.035)\cdot(-0.05)\cdot(-0.055)}\cdot f(0.115)$$

Очевидно, що проводити таку к-ть обчислень вручну не є доцільним, тому решту обчислень проведемо програмно.

Для знаходження інтерполяційного поліному Ньютона нам необхідно знайти розділені різниці. Знову ж таки, знаходження всіх розділених різниць потребує довгих обчислень, тому покажемо обчислення лише деяких:

$$f(x_0,x_1)=\frac{f(x_1)-f(x_0)}{x_1-x_0}=\frac{8{,}29-8{,}68}{0.120-0.115}=-78$$

$$f(x_1,x_2)=\frac{f(x_2)-f(x_1)}{x_2-x_1}=\frac{7{,}96-8{,}29}{0{,}125-0{,}12}=-66$$

$$f(x_0,x_1,x_2)=\frac{f(x_1,x_2)-f(x_0,x_1)}{x_2-x_0}=\frac{-66+78}{0{,}125-0.115}=1200$$

Тоді перші доданки інтерполяційного поліному Ньютона запишемо наступним чином

$$P(x)=8{,}29-78(x-0.115)+1200(x-0.115)(x-0.12)+\ldots$$

Наступна мета – реалізувати подані методи програмно, використовуючи отримані знання, для того щоб завершити розрахунки.

Для легшої роботи створимо структуру даних Polynom для керування поліномами, в ній реалізуємо операції множення та додавання поліномів наступним чином:

```
public static Polynom operator+(Polynom a, Polynom b)
        {
            Polynom result = new Polynom(Math.Max(a.Coefs.Length, b.Coefs.Length));
            for(int i = 0; i < Math.Max(a.Coefs.Length, b.Coefs.Length); i++)
            {
                result.Coefs[i] = a.Coefs.ElementAtOrDefault(i) +
b.Coefs.ElementAtOrDefault(i);
            }
            return result;
        }
```

```
        public static Polynom operator*(Polynom a, Polynom b)
        {
            Polynom result = new Polynom(a.Coefs.Length + b.Coefs.Length - 1);

            for(int i = 0; i < a.Coefs.Length; i++)
            {
                for(int j = 0; j < b.Coefs.Length; j++)
                {
                    result.Coefs[i + j] += a.Coefs[i] * b.Coefs[j];
                }
            }

            return result;
        }
```

І тоді отримання поліномів Лагранжа та Ньютона опишемо наступним чином:

```
        public static Polynom GetLangrange(KeyValuePair<decimal, decimal>[] xypairs)
        {
            Polynom res = new Polynom(xypairs.Length);

            for(int i = 0; i < xypairs.Length; i++)
            {

                Polynom L_i = 1; //Поліном-доданок
                for(int j = 0; j < xypairs.Length; j++)
                {
                    if (i == j) continue; // Пропускаємо, якщо номер доданку дорівнює
номеру точки
                    Polynom numerator = new Polynom(new decimal[] { -xypairs[j].Key, 1 });
                    decimal denominator = (xypairs[i].Key - xypairs[j].Key);
                    L_i *= numerator * (1M / denominator);
                }
                res += L_i * xypairs[i].Value;
            }
            //Повертаємо результат
            return res;
        }

        public static Polynom GetNewton(KeyValuePair<decimal, decimal>[] xypairs)
        {
            //Масив-таблиця для зберігання розділених різниць
            decimal[,] diff = new decimal[xypairs.Length, xypairs.Length + 1];

            for(int i = 0; i < xypairs.Length+1; i++)
            {
                for(int j = 0; j < xypairs.Length - i; j++)
                {
                    if (i == 0) diff[j, i] = xypairs[j].Value;
                    else
                    {
                        //Для обчислення різниці i-го порядку використовуємо різницю (i-1)-
го порядку
                        decimal num = (diff[j + 1, i - 1] - diff[j, i - 1]);
                        decimal denom = (xypairs[j + i].Key - xypairs[j].Key);
                        diff[j, i] = num / denom;
                    }
                }
            }

            Polynom result = new Polynom();


            for(int i = 0; i < xypairs.Length; i++)
            {
                Polynom P_i = diff[0, i]; //Поліном-доданок
                for (int j = 0; j < i; j++)
```

```
                P_i *= new Polynom(new decimal[] { -xypairs[j].Key, 1 });
            result += P_i;
        }
        //Повертаємо результат
        return result;
    }
```

Подамо 10 точок з варіанту завдання на вхід програми і подивимось на результат

<u>Результат</u> виконання програми:

```
Поліном Лагранжа:

325541125541125,54x^9
-410385762385762,39x^8
+229367234247234,25x^7
-74598167003367,00x^6
+15559174952861,95x^5
-2158278933720,54x^4
+199112360132,37x^3
-11780711450,94x^2
+405636586,20x
-6193015,94

Поліном Ньютона:

325541125541125,54x^9
-410385762385762,39x^8
+229367234247234,25x^7
-74598167003367,00x^6
+15559174952861,95x^5
-2158278933720,54x^4
+199112360132,37x^3
-11780711450,94x^2
+405636586,20x
-6193015,94

Перевіримо, чи значення поліномів в ключових точках співпадають з таблицею:

0,115 => Лагранж: 8,680 ; Ньютон: 8,680
0,120 => Лагранж: 8,290 ; Ньютон: 8,290
0,125 => Лагранж: 7,960 ; Ньютон: 7,960
0,130 => Лагранж: 7,650 ; Ньютон: 7,650
0,135 => Лагранж: 7,360 ; Ньютон: 7,360
0,140 => Лагранж: 7,100 ; Ньютон: 7,100
0,145 => Лагранж: 6,850 ; Ньютон: 6,850
0,150 => Лагранж: 6,620 ; Ньютон: 6,620
0,165 => Лагранж: 6,400 ; Ньютон: 6,400
0,170 => Лагранж: 6,200 ; Ньютон: 6,200


Значення поліномів в точці x0

0,122 => Лагранж: 8,152 ; Ньютон: 8,152
```

Рис 1.1 Результат виконання програми.

<u>Аналіз результатів</u>:

Як бачимо, поліноми, отримані обома методами, співпали. До того ж, значення поліномів в ключових точках збігаються з табличними. Це свідчить про те, що завдання з високою вірогідністю було виконано правильно. Також, можемо бачити, що x_0 лежить між 2-им та 3-ім значенням аргументів в таблиці. Враховуючи, що при $0.115 \leq x \leq 0.170$ функція монотонно спадає, те, що значення функції лежить в межах значень функції в 2-ій та 3-ій точках таблиці $(7.960 \leq 8.152 \leq 8.290)$ є очікуваним.

<u>Висновок</u>:

Виконуючи цю лабораторну роботу, ми ознайомилися з методом інтерполяції таблично заданих функцій.

За допомогою цих знань ми реалізували програму, що змогли представити таблично задану функцію

x	0,115	0,120	0,125	0,130	0,135	0,140	0,145	0,150	0,165	0,170
y	8,68	8,29	7,96	7,65	7,36	7,10	6,85	6,62	6,40	6,20

у вигляді поліному методами Ньютона та Лагранжа.

Також, ми знайшли значення цього поліному у точці $x_0 = 0.122$ і отримали, що $P(x_0) \approx 8.192$.