Grupa lab.3	Data wykonania 08.03.2024	Inżynieria Obliczeniowa 2023/2024
Temat ćwiczenia **Interpolacja Newtona**		
Imię i nazwisko Karolina Kurowska		Ocena i uwagi

Wstęp

Interpolacja jest metodą matematyczną wykorzystywaną do przybliżania wartości pomiędzy znanymi punktami danych. Jej głównym celem jest znalezienie funkcji, która najtrafniej oddaje charakter danych, umożliwiając oszacowanie wartości w obszarach między istniejącymi punktami danych. Metoda interpolacji wielomianowej Newtona jest jednym z narzędzi wykorzystywanych w praktyce do tego celu.

Przebieg ćwiczenia

Początek zadania wyglądał identycznie jak podczas ostatnich zajęć, czyli wpisanie danych z plików tekstowych do macierzy interpol. Aby obliczyć interpolację wielomianową Newtona należy zaimplementować wzór.:

$$W_n(x) = \sum_{k=0}^{n} b_k p_k(x),$$

Wielomian interpolacyjny Newtona ma postać ogólną, która opiera się na ilorazach różnicowych funkcji dla węzłów interpolacji. Wygląda to jak na obrazkach poniżej. Dla lepszego zobrazowania wzoru w funkcji main automatycznie dopisuje wartość wielomianu dla elementu pierwszego, gdyż jak wiemy wartość pierwszego elementu zawsze wynosi p_0=1 i b_0=f(x_0).

```
double polynomial;
double sum = 0.0;
cout << "0 Wspolczynnik wielomianu Newtona: " << interpol[0][1] << endl;
sum = newton (quantity, interpol, point);
polynomial = interpol[0][1] + sum;
cout << "Wartosc funkcji w danym punkcie " << polynomial << endl;
```

```
double newton (int quantity, double ** table, double point) {
    double sum = 0.0;

    for (int k = 1; k < quantity; k++) {
        sum += (count_b(k, table) * count_p(k, point, table));
    }

    return sum;
}
```

Kolejne potrzebne obliczenia elementów b_k i p_k gwarantuje implementacja kolejnych funkcji kolejno *count_b* i *count_p*. Funkcja count_b implementuje wzór na współczynniki wielomianu Newtona:

$$b_k = \sum_{i=0}^{k} \frac{f(x_i)}{\prod_{j=0,\, j\neq i}^{k} (x_i - x_j)},$$

W celu zwiększenia czytelności kodu mianownik tego wzoru został obliczony w osobnej funkcji *count_multi.*

```
double count_b (int quantity, double ** table) {
    double result = 0.0;
    for (int i = 0; i <= quantity; i++) {
        result += (table[i][1] / count_multi(quantity, i, table));
    }
    cout << quantity << " Wspolczynnik wielomianu Newtona: " << result <<endl;
    return result;
}
double count_multi (int quantity, int i, double ** table) {
    double result = 1;
    for (int j = 0; j <= quantity; j++) {
        if (j == i) {
            continue;
        }
        result *= (table[i][0] - table[j][0]);
    }
    return result;
}
```

Funkcja *count_p* opisuje wzór:

$$p_k = \prod_{i=0}^{k-1} (x - x_i).$$

```
double count_p (int quantity, double point, double** table) {
    double result = 1;
    for (int i = 0; i <= (quantity - 1); i++) {
        result *= (point - table[i][0]);
    }
    return result;
}
```

Wykorzystując stworzony przeze mnie kod otrzymaliśmy oczekiwane wyniki:
dla pliku *MN-2-p1.txt*

Podany punkt	Wartość
2.5	6.25
3.5	12.25

```
Ilosc podanych wezlow: 5
wezel 1->1
wezel 2->4
wezel 3->9
wezel 4->16
wezel 5->25
Podaj punkt w ktorym obliczymy wartosc wielomianu: 2.5
0 Wspolczynnik wielomianu Newtona: 1
1 Wspolczynnik wielomianu Newtona: 3
2 Wspolczynnik wielomianu Newtona: 1
3 Wspolczynnik wielomianu Newtona: -4.44089e-16
4 Wspolczynnik wielomianu Newtona: 2.22045e-16
Wartosc funkcji w danym punkcie 6.25

C:\Users\kkuro\source\repos\lab02_met_num\x64\Debug\lab02_met_num.exe (proces 32236) zakończono z kodem 0.
Naciśnij dowolny klawisz, aby zamknąć to okno...
```

```
Ilosc podanych wezlow: 5
wezel 1->1
wezel 2->4
wezel 3->9
wezel 4->16
wezel 5->25
Podaj punkt w ktorym obliczymy wartosc wielomianu: 3.5
0 Wspolczynnik wielomianu Newtona: 1
1 Wspolczynnik wielomianu Newtona: 3
2 Wspolczynnik wielomianu Newtona: 1
3 Wspolczynnik wielomianu Newtona: -4.44089e-16
4 Wspolczynnik wielomianu Newtona: 2.22045e-16
Wartosc funkcji w danym punkcie 12.25

C:\Users\kkuro\source\repos\lab02_met_num\x64\Debug\lab02_met_num.exe (proces 23012) zakończono z kodem 0
Naciśnij dowolny klawisz, aby zamknąć to okno...
```

dla pliku *MN-2-p2.txt*

Podany punkt	Wartość
-1	7.89617
2	-2.90084

```
Konsola debugowania progra

Ilosc podanych wezlow: 8
wezel -2->4
wezel 0->1
wezel 1->-3
wezel 4->0
wezel 5->1
wezel 7->7
wezel 10->10
wezel 12->8
Podaj punkt w ktorym obliczymy wartosc wielomianu: -1
0 Wspolczynnik wielomianu Newtona: 4
1 Wspolczynnik wielomianu Newtona: -1.5
2 Wspolczynnik wielomianu Newtona: -0.833333
3 Wspolczynnik wielomianu Newtona: 0.347222
4 Wspolczynnik wielomianu Newtona: -0.0853175
5 Wspolczynnik wielomianu Newtona: 0.0152116
6 Wspolczynnik wielomianu Newtona: -0.00196502
7 Wspolczynnik wielomianu Newtona: 0.000219566
Wartosc funkcji w danym punkcie 7.89617

C:\Users\kkuro\source\repos\lab02_met_num\x64\Debug\lab02_met_num.exe (proces 17964) zakończono z kodem 0.
Naciśnij dowolny klawisz, aby zamknąć to okno...
```
,

```
Ilosc podanych wezlow: 8
wezel -2->4
wezel 0->1
wezel 1->-3
wezel 4->0
wezel 5->1
wezel 7->7
wezel 10->10
wezel 12->8
Podaj punkt w ktorym obliczymy wartosc wielomianu: 2
0 Wspolczynnik wielomianu Newtona: 4
1 Wspolczynnik wielomianu Newtona: -1.5
2 Wspolczynnik wielomianu Newtona: -0.833333
3 Wspolczynnik wielomianu Newtona: 0.347222
4 Wspolczynnik wielomianu Newtona: -0.0853175
5 Wspolczynnik wielomianu Newtona: 0.0152116
6 Wspolczynnik wielomianu Newtona: -0.00196502
7 Wspolczynnik wielomianu Newtona: 0.000219566
Wartosc funkcji w danym punkcie -2.90048

C:\Users\kkuro\source\repos\lab02_met_num\x64\Debug\lab02_met_num.exe (proces 30264) zakończono z kodem 0.
Naciśnij dowolny klawisz, aby zamknąć to okno...
```

Wnioski

Interpolacja wielomianowa Newtona jest efektywną metodą przybliżania wartości funkcji pomiędzy znanymi punktami danych. Interpolacja stanowi potężne narzędzie w analizie danych i numerycznych obliczeniach, pozwalając na precyzyjne oszacowanie wartości funkcji w miejscach, gdzie dane nie są bezpośrednio dostępne. Jej zalety to:

- ➔ Łatwość implementacji,
- ➔ Możliwość dostosowania do nieregularnych odległości między węzłami.

Jednakże, metoda ta może prowadzić do dużych błędów interpolacji w przypadku funkcji gwałtownie zmieniających się w przedziale interpolacji.