

Wydział WIMiIP	Imię i Nazwisko Mateusz Witkowski	Rok II	Grupa 4
Kierunek IS	Temat Interpolacja Lagrange'a	Prowadzący dr hab. inż. Hojny Marcin, prof. AGH	
Data ćwiczenia 05.03.20	Data oddania 18.03.20	Data zaliczenia	Ocena

1. Cel ćwiczenia:

Celem ćwiczenia było napisanie implementacji metody numerycznej - interpolacji Lagrange'a pozwalającej na znalezienie wartości funkcji w dowolnym punkcie.

2. Wprowadzenie do Interpolacji Lagrange'a

Interpolacja Lagrange'a (lub też interpolacja wielomianowa) jest to metoda numeryczna wykorzystywana do przybliżania przebiegu funkcji przy pomocy tak zwanego "wielomianu Lagrange'a" stopnia n . Wybieranych jest $n+1$ punktów zwanych węzłami interpolacji, należących do dziedziny danej funkcji. W punktach tych funkcja(wielomian) przyjmuje wartości takie same jak przybliżana funkcja.

Wielomian wylicza się przy pomocy wzoru:

$$L(x) = \sum_{i=1}^n y_i l_i(x)$$

Gdzie:

x – argument dla którego chcemy znaleźć wartość

y_i - wartość funkcji odpowiadająca x_i

Wzór na współczynniki l_i :

$$l_i(x) = \prod_{0 < j \leq n \wedge j \neq i} \frac{x - x_j}{x_i - x_j}$$

3. Kod źródłowy programu

```
#include <iostream>
#include <vector>
#include <fstream>
#include <sstream>
#include <cstdlib>
#include <string>
using namespace std;

double lagrange(int liczba_punktow, double* X, double* Y, double szukana) { //funkcja realizująca Interpolację Lagrange'a.
    double wspolczynnik_l = 0; //zmienna przechowująca wartość pojedynczego iloczynu Y[i]*L[i](x)
    double wynik = 0;
    for (int i = 0; i < liczba_punktow; i++) {
        wspolczynnik_l = Y[i]; //zapisanie wartości kolejnych Y[i](x)
        for (int j = 0; j < liczba_punktow; j++)
        {
            if (j != i) {
                wspolczynnik_l *= (szukana - X[j]) / (X[i] - X[j]); //obliczenie współczynnika L[i](x)
            }
        }
        wynik += wspolczynnik_l;
    }
    return wynik;
}

int pobranie_z_pliku(vector <double>* data, string fileName) { //Funkcja pobierająca dane z pliku tekstowego i zapisujące je do wektora.
    string linia;
    fstream plik;

    plik.open(fileName, ios::in);
    if (plik.good() == true)
    {
        while (!plik.eof())
        {
            getline(plik, linia, ','); //zapisuje słowa oddzielane przecinkami
            double d = atof(linia.c_str());
            data->push_back(d); //zapis słowa parsowanego na typ double do wektora
        }
        plik.close();
    }
    if (data->size() % 2 == 1) { //zabezpieczenie w razie źle wypełnionego pliku tekstowego
        cout << "błędne dane w pliku" << endl;
        return -1;
    }
    return data->size(); //funkcja zwraca wielkość wektora potrzebna do stworzenia tablic przechowujących X i Y
}
```

```

int main() {
    vector < double > dane;
    string nazwaPliku = "punkty.txt";
    int wielkosc_wektora = pobranie_z_pliku(&dane, nazwaPliku);
    if (wielkosc_wektora == -1) //program zostanie przerwany jesli plik tekstowy był błędny
    {
        return -1;
    }
    int liczba_punktow = wielkosc_wektora / 2; //wielkosc wektora podzielona przez 2 powinna nam dać ilość punktów zapisanych w pliku
    double* X = new double[liczba_punktow]; //tablica współrzędnych x
    double* Y = new double[liczba_punktow]; //tablica współrzędnych y
    int j = 0;
    for (size_t i = 0; i < dane.size(); i++) //wypełnianie tablic
    {
        if (i % 2 == 0) {
            X[j] = dane[i];
            cout << "f(" << X[j] << ") = ";
        }
        else {
            Y[j] = dane[i];
            cout << Y[j] << endl;
            j++;
        }
    }
    double szukana;
    cout << "Podaj wartosc szukanego x: ";
    cin >> szukana;
    cout << endl;
    cout << "Wartosc dla podanego x to: " << lagrange(liczba_punktow, X, Y, szukana) << endl;
    return 0;
}

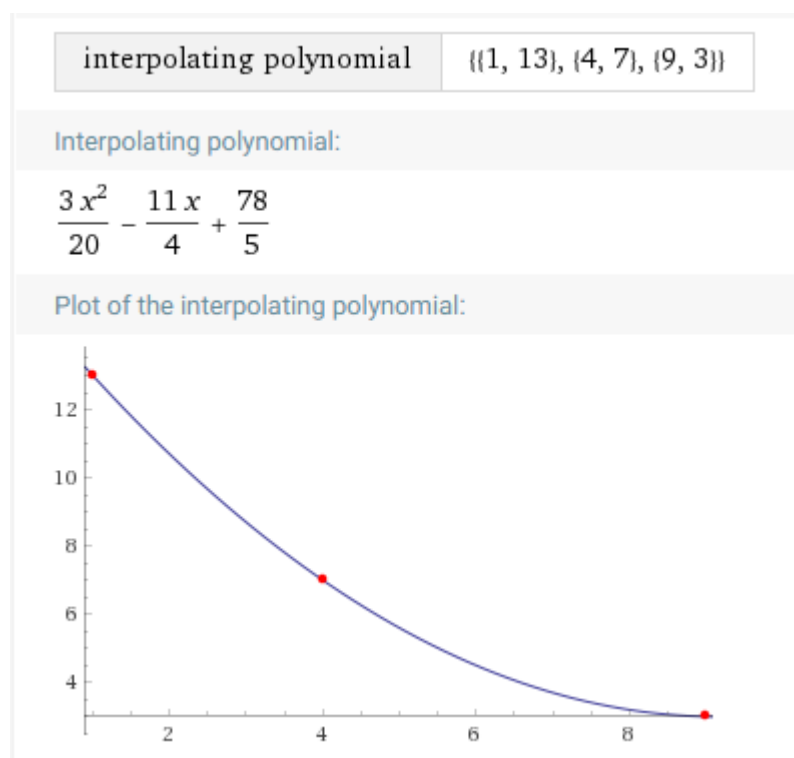
```

4. Testy poprawności działania algorytmu

Testy wykonano przy pomocy programu WolframAlpha. Do programu wprowadzono losowo wybrane punkty w celu uzyskanie wykresu oraz wzoru funkcji. Dzięki czemu można było łatwo podstawić szukane wartości i porównać wyniki z Wolframa i mojego programu.

Szukana wartość dla $x = 17$

Wolfram:



Wynik po podstawieniu 12.2

Mój program:

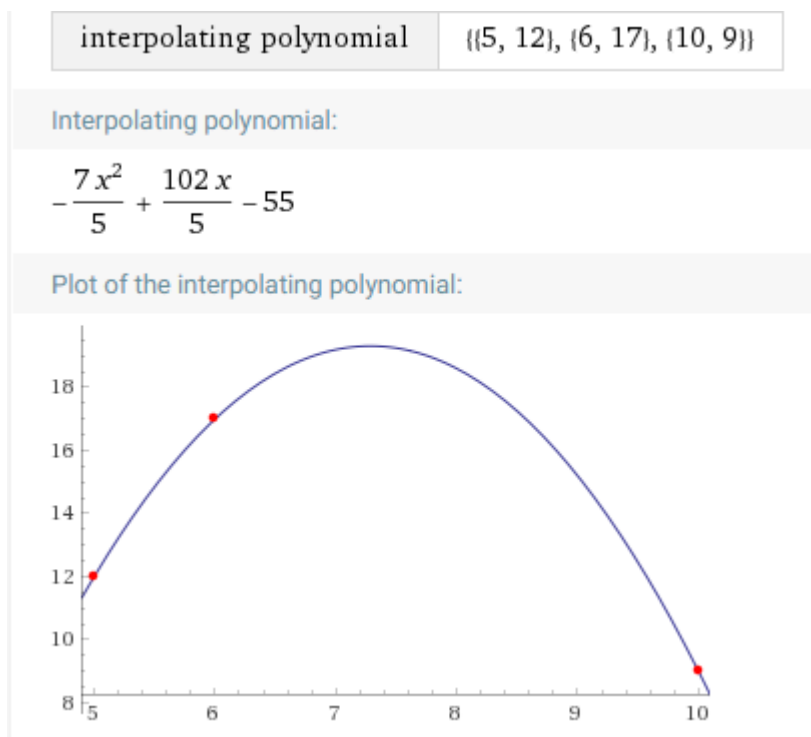
```
punkty.txt  x  Źródło.cpp
1  1, 13,
2  4, 7,
3  9, 3
```

Konsola debugowania programu Microsoft Visual Studio

```
f(1) = 13
f(4) = 7
f(9) = 3
Podaj wartosc szukanego x: 17
Wartosc dla podanego x to: 12.2
```

Szukana wartość dla $x = 5.5$

Wolfram:



Wynik po podstawieniu 14.85

Mój program:

```
punkty.txt  x  Źródło.cpp
1  5, 12,
2  6, 17,
3  10, 9
```

Konsola debugowania programu Microsoft Visual Studio

```
f(5) = 12  
f(6) = 17  
f(10) = 9  
Podaj wartosc szukanego x: 5.5  
  
Wartosc dla podanego x to: 14.85
```

5.Wnioski

Interpolacja Lagrange'a jest jedną z najprostszych metod interpolacyjnych, umożliwia znalezienie jednoznacznego rozwiązania, gdy posiadamy wiedzę o należących do funkcji punktach i ich wartościach. Większa ilość znanych punktów zapewnia większą dokładność. Program nie pokaże nam przybliżonej postaci, jedynie wartość dla konkretnego x 'a. Testy potwierdziły poprawność programu zarówno względem wyników z Wolframa jak i tych obliczonych w tradycyjny sposób.