Obliczenia naukowe

Sprawozdanie | Lista 3

Illia Azler | 239537

zad.1

Celem zadania było napisać funkcję obliczającą ilorazy różnicowe.
Dla realizacji pomysłu algorytmu, wykorzystałem pseudokod z książki:

[1] D. Kincaid, W. Cheney, Analiza numeryczna, WNT, 2005.
Iloraz różnicowy str.314

```
for i = 0 to n do
    d_i ← f(x_i)
end do
for j = 1 to n do
    for i = n to j step −1 do
        d_i ← (d_i − d_{i−1})/(x_i − x_{i−j})
    end do
end do
```

Dane:

x – wektor długosci n + 1 zawierajacy wezły $x_0 , \dots , x_n$

$$x[1] = x_0 , \dots , x[n+1] = x_n$$

f – wektor długosci n + 1 zawierajacy wartosci interpolowanej funkcji w wezłach f(x_0), . . . , f(x_n)

Wyniki:

f_x — wektor długości n+1 zawierający obliczone ilorazy różnicowe

Teoria:

Podejście do implementacji funkcji obliczającej ilorazy różnicowe może być zoptymalizowane poprzez wykorzystanie jednowymiarowej tablicy zamiast macierzy dwuwymiarowej. Skupiamy się wyłącznie na danych z pierwszego wiersza tej tablicy, a ponieważ każda kolumna zależy tylko od poprzedniej, proponujemy efektywne rozwiązanie. Inicjalnie przechowujemy wartości pierwszej kolumny w jednowymiarowej tablicy, korzystając z dostępnych danych funkcji w odpowiadających węzłach. W kolejnych krokach wpisujemy odpowiednie wartości z kolejnych kolumn na ostatnie miejsca w tablicy. W efekcie otrzymujemy tablicę zawierającą jedynie wartości ilorazów różnicowych z pierwszego wiersza, co może przynieść bardziej wydajne rozwiązanie.

Idea algorytnu polega na tym, że w danym momencie pamiętamy tylko jedną kolumnę macierzy trójkątnej reprezentującej wszystkie ilorazy różnicowe:

$$
\begin{array}{ccccccccc}
f[x_0] & f[x_0,x_1] & \cdots & f[x_0,\ldots,x_{m-1}] & f[x_0,\ldots,x_m] & \cdots & f[x_0,\ldots,x_{n-1}] & f[x_0,\ldots,x_n] \\
f[x_1] & f[x_1,x_2] & \cdots & f[x_1,\ldots,x_m] & f[x_1,\ldots,x_{m+1}] & \cdots & f[x_1,\ldots,x_n] & \\
\vdots & \vdots & & \vdots & \vdots & \iddots & & \\
f[x_k] & f[x_k,x_{k+1}] & \cdots & f[x_k,\ldots,x_{k+m-1}] & f[x_k,\ldots,x_{k+m}] & & & \\
f[x_{k+1}] & f[x_{k+1},x_{k+2}] & \cdots & f[x_{k+1},\ldots,x_{k+m}] & & & & \\
\vdots & & \iddots & & & & & \\
f[x_{n-1}] & f[x_{n-1},x_n] & & & & & & \\
f[x_n] & & & & & & &
\end{array}
$$

zad.2

Celem zadania była implementacja funkcji obliczającej wartość wielomianu interpolacyjnego stopnia n w postaci Newtona $N_{\mathrm{n}}(x)$ w punkcie $x = t$ za pomoca uogólnionego algorytmu Hornera, w czasie $O(n)$.

$$N_n(x) = \sum_{i=0}^{n} f[x_0, x_1, \ldots, x_i] \prod_{j=0}^{i-1} (x - x_j)$$

Dla realizacji pomysłu algorytmu, wykorzystałem pseudokod z książki:

[6] J. i M. Jankowscy, Przegląd metod i algorytmów numerycznych, cz. 1, WNT, 1988

Newton str.42

```
w := b[n];
for i := n-1 step -1 until 0 do
w := w×(x-x[i])+b[i];
```

Dane:

x – wektor długosci n + 1 zawierajacy wezły $x_0, \ldots, x_n$

$x[1] = x_0, \ldots, x[n+1] = x_n$

f_{x} – wektor długosci n + 1 zawierajacy ilorazy różnicowe

f_{x} [1] = f[x_0],

f_{x} [2] = f[x_0, x_1],..., fx[n]=f[$x_0, \ldots, x_{n-1}$], fx[n+1]=f[$x_0, \ldots, x_n$]

t – punkt, w którym nalezy obliczyc wartosc wielomianu

Wyniki:
nt – wartosc wielomianu w punkcie t.

zad.3

Celem zadania była implementacja funkcji obliczającej współczynniki $a_0, \dots, a_n$ postaci naturalnej wielomianu interpolacyjnego dla zadanych współczynników $d_0 = f[x_0]$, $d_1 = f[x_0, x_1]$, ... , $d_n = f[x_0, \dots, x_n]$ tego wielomianu w postaci Newtona oraz węzłów $x_0, \dots, x_n$. Funkcja miała działać w czasie $O(n^2)$.

Dla realizacji pomysłu algorytmu, wykorzystałem pseudokod z książki:

[6] J. i M. Jankowscy, Przegląd metod i algorytmów numerycznych, cz. 1, WNT, 1988

Postać naturalna str.43

```
a[n] := b[n];
for i := n−1 step −1 until 0 do
begin
   xi := x[i];
   a[i] := b[i];
   for k := i step 1 until n−1 do
   a[k] := a[k]−xi × a[k+1]
end;
```

Dane:
x – wektor długosci n + 1 zawierajacy wezły $x_0, \dots, x_n$

$x[1] = x_0, \dots, x[n+1] = x_n$

f_x – wektor długosci n + 1 zawierajacy ilorazy róznicowe

f_x [1]=f[x_0],

f_x [2]=f[x_0, x_1],..., fx[n]=f[$x_0, \dots, x_{n-1}$], fx[n+1]=f[$x_0, \dots, x_n$]

Wyniki:
a – wektor długosci n + 1 zawierajacy obliczone współczynniki postaci naturalnej
a [1] = a_0,
a [2] = a_1,..., a[n]= a_{n-1}, a[n+1]= a_n.

zad.4

Celem zadania była implementacja funkcji, która zainterpeluje zadaną funkcję f(x) w przedziale [a, b] za pomocą wielomianu interpolacyjnego stopnia n w postaci Newtona. Następnie narysuje wielomian interpolacyjny i interpolowaną funkcję. Do rysowania użyłem pakiet **Plots**.

W interpolacji użyłem węzłów równoodległych:

$x_k = a + kh$ dla $h = \frac{b-a}{n}$, k = 0,1, ... , n.

Prowadzone testy dla zadań 1-4:

Dla testów obrałem przykład z wykładu:

x	3	1	5	6
$f(x)$	1	-3	2	4

Korzystając z Twierdzenia 3 konstruujemy tablicę ilorazów różnicowych

$$\begin{array}{ccccc} 3 & 1 & 2 & -3/8 & 7/40 \\ 1 & -3 & 5/4 & 3/20 & \\ 5 & 2 & 2 & & \\ 6 & 4 & & & \end{array}$$

Źródło obrazku: Obliczenia Naukowe, prof. dr. hab. Paweł Zieliński, Wykład nr.6

Test dla zadania 1:

Wynik testu:

x0	x1	x2	x3
1.0	2.0	-0.375	0.17500000000000002

Wyniki są zgodne z macierzą.
Ostatni iloraz ma lekki błąd 2 * 10^{-17}, to błąd, którego nie da się uniknąć, ponieważ to wynika z błędu przy wykonaniu jakichkolwiek działań.

Test dla zadania 2:

Wynik testu:

x	5.0
f(x)	2.0

Czyli dla punktu 5.0 mamy wynik 2.0, co zgadza się.

Test dla zadania 3:

Wynik testu:

a4	a3	a2	a0
-8.75	7.525	-1.9500000000000002	0.17500000000000002

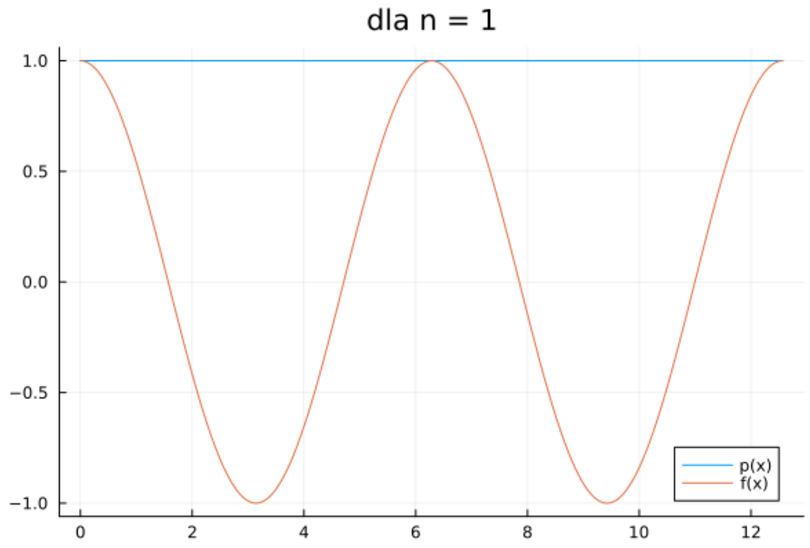
Wyniki są zgodne z macierzą.
Przy współczynnikach a2 i a1 ponownie występują nam lekkie błędy $2 * 10^{-17}$, jak powiedziano wcześniej to błąd, którego nie da się uniknąć, ponieważ to wynika z błędu przy wykonaniu jakichkolwiek działań.

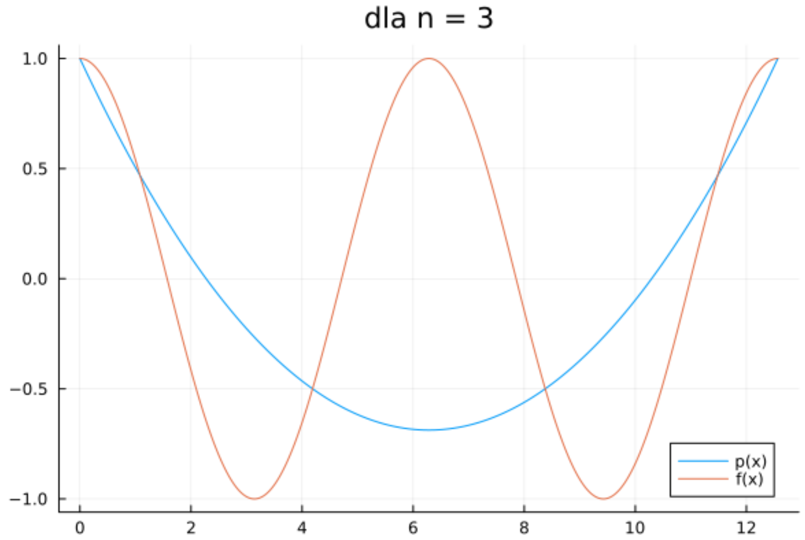
Obliczone współczynniki przy pomocy Wolfram Alpha dla sprawdzenia:

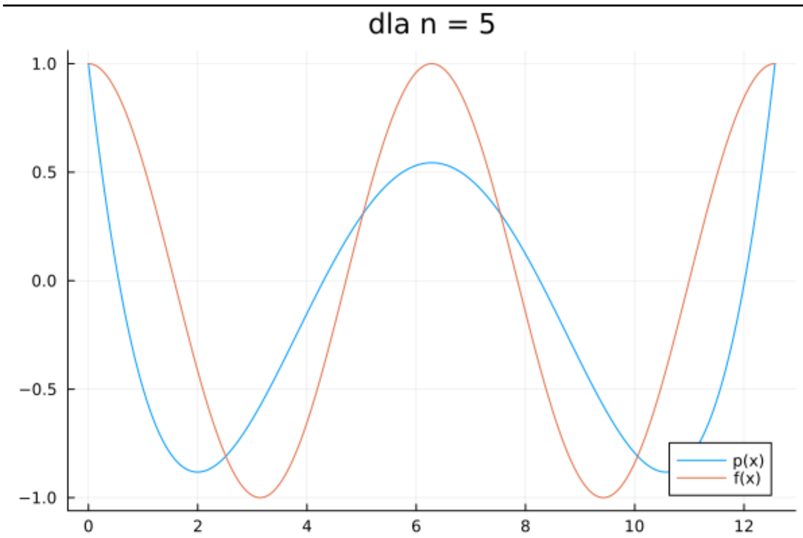
$$w_0\, x = \frac{7\,x^3}{40} - \frac{39\,x^2}{20} + \frac{301\,x}{40} - \frac{35}{4}$$

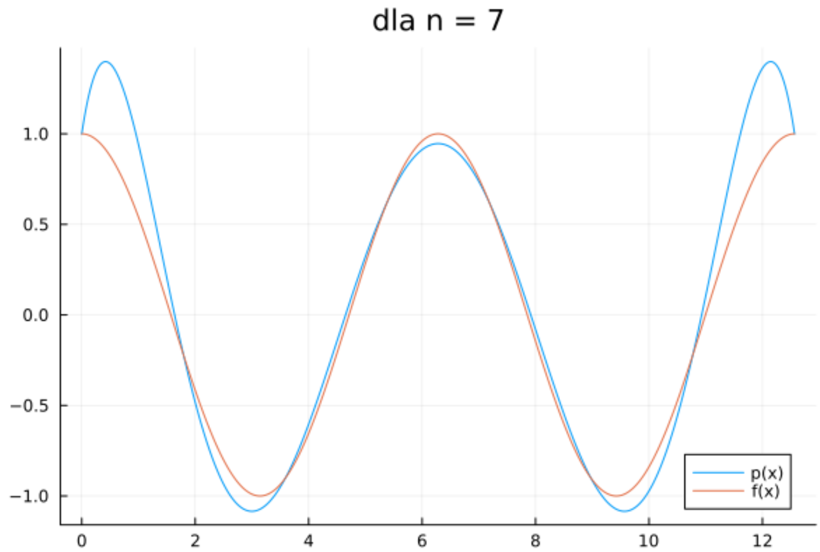
Test dla zadania 4:

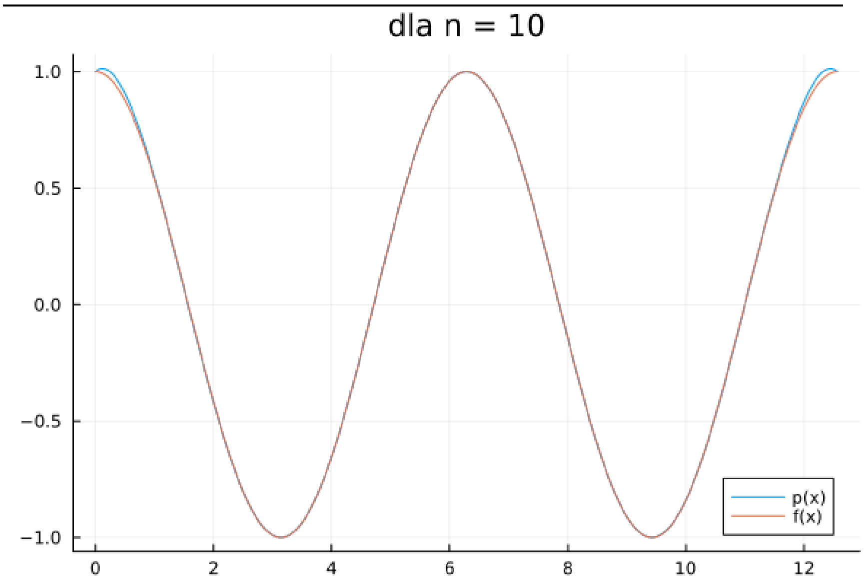
Dla testu wybrałem funkcję **cos(x).**
p(x) - wielomian, f(x) - funkcja.
Dla funkcji cos(x) przy stopniu n=1, f(x) jest zwykłą cosinusoidą, a p(x) jest ciągła prosta w 1. Krzywa p(x) jest różna od f(x), ale z każdym stopniem o jeden większy zaczynając od n=1 powoli zbiega się do f(x), najwyraźniej zaczynają się nakładać przy stopniu n=10.
Wykresy dla stopni: **n = 1, 3, 5, 7, 10, 15:**

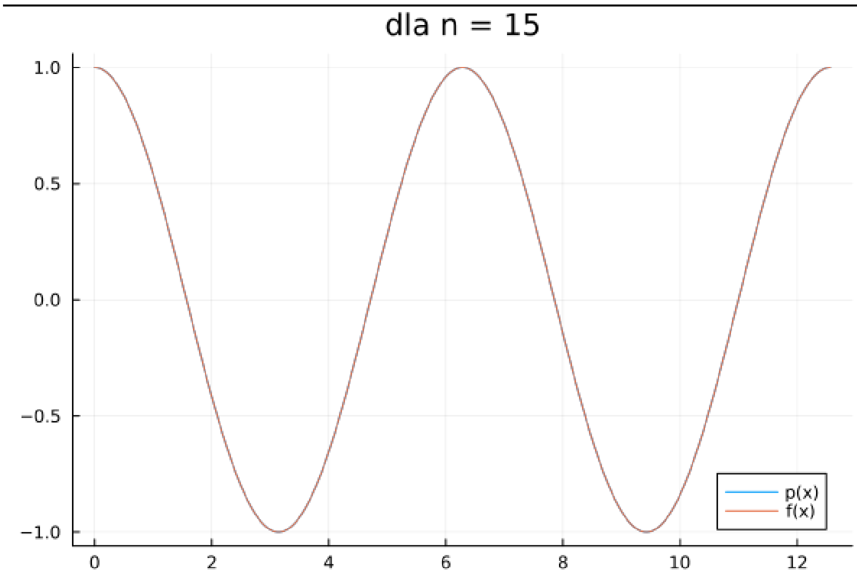

dla n = 1
1.0
0.5
0.0
−0.5
−1.0
0
2
4
6
8
10
12
p(x)
f(x)


dla n = 3
1.0
0.5
0.0
−0.5
−1.0
0
2
4
6
8
10
12
p(x)
f(x)


dla n = 5
1.0
0.5
0.0
−0.5
−1.0
0
2
4
6
8
10
12
p(x)
f(x)







Wniosek:

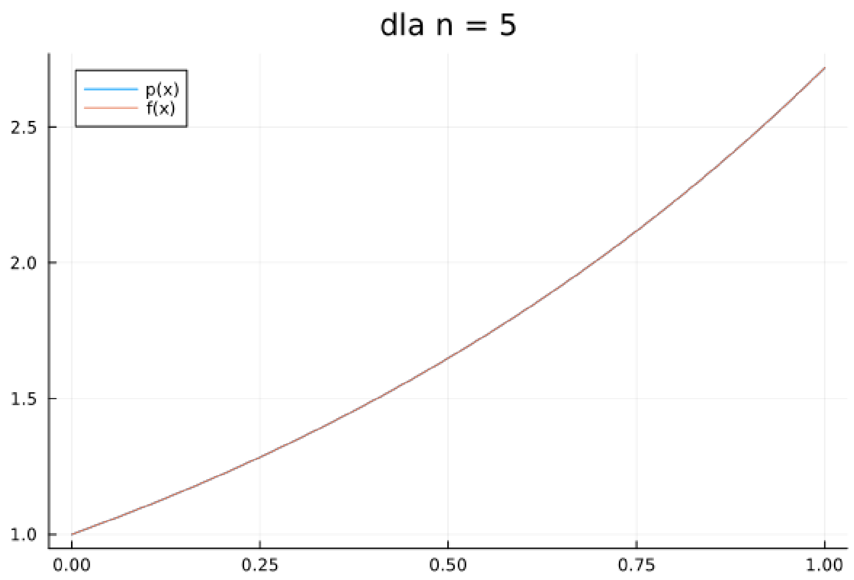
Przy stosowaniu interpolacji wielomianowej na równoodległych węzłach, zwłaszcza przy zwiększaniu stopnia wielomianu, może wystąpić efekt Rungego. Efekt ten wykazuje się poprzez oscylacje wielomianu interpolacyjnego wokół funkcji na krańcach przedziału interpolacji. W praktyce, dla wysokich stopni interpolacji, zjawisko to może prowadzić do znaczących odchyleń wielomianu interpolacyjnego od interpolowanej funkcji, zwłaszcza w obszarach blisko krańców przedziału.

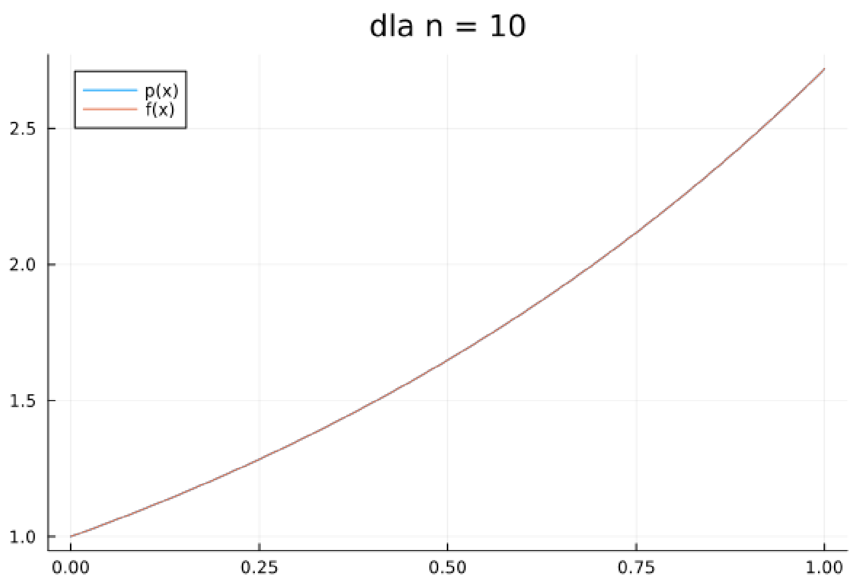
Wybór odpowiednich węzłów interpolacyjnych może być kluczowy dla uzyskania stabilnych i dokładnych wyników interpolacji, szczególnie dla funkcji o skomplikowanych wzorcach oscylacyjnych, takich jak cos(x).

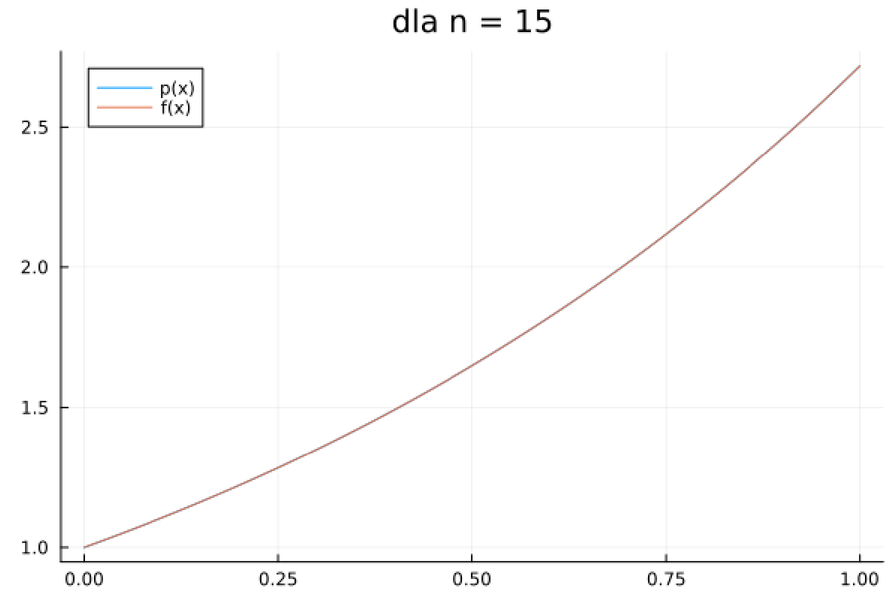
zad.5

Funkcja(a):

$f(x) = e^{x}, dla\ n = 5, 10, 15.$

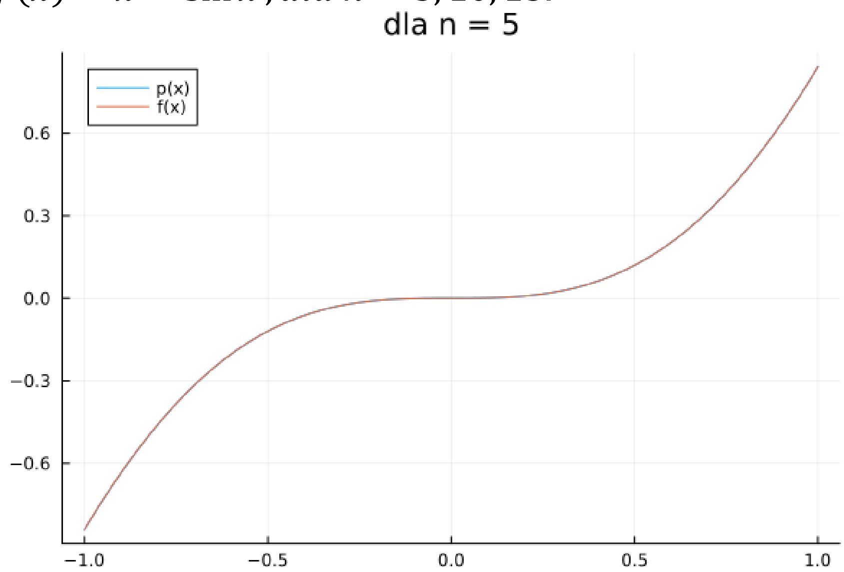


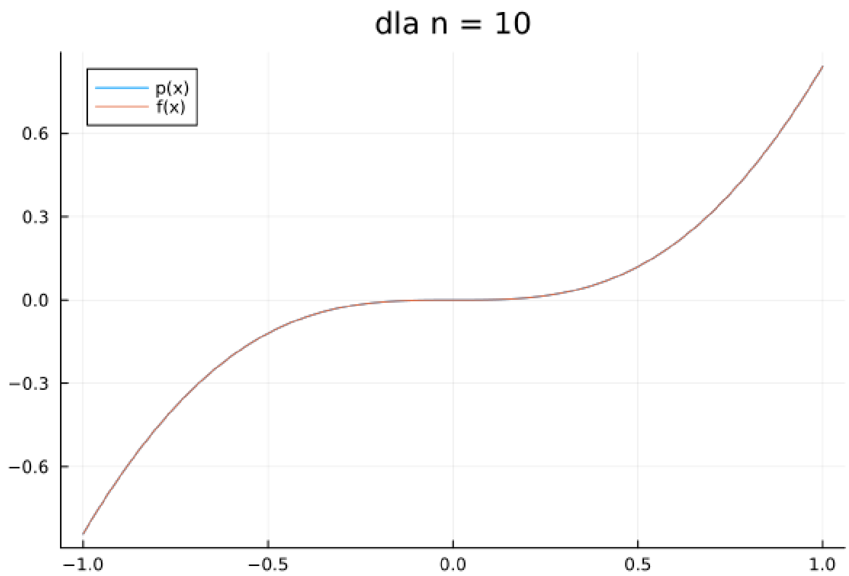


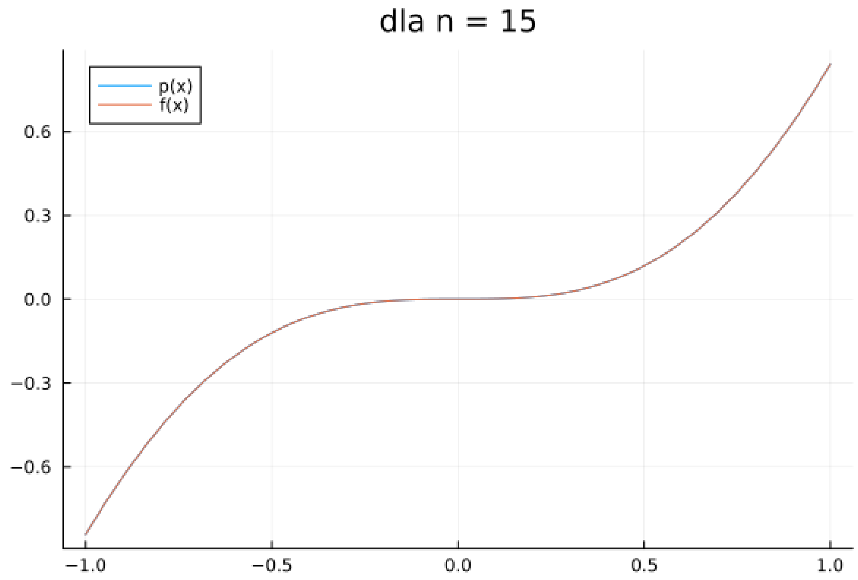


Funkcja (b):

$f(x) = x^2 * \sin x \, , dla \; n = 5, 10, 15.$







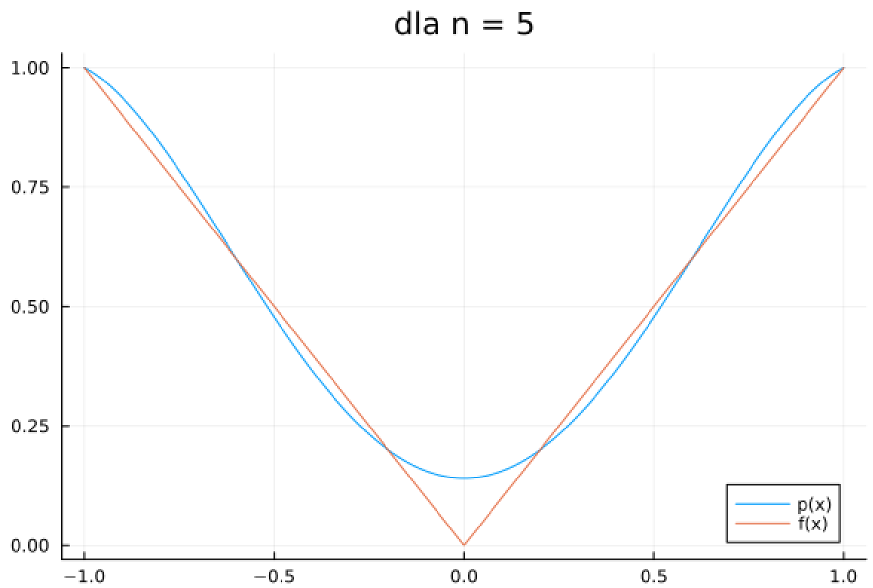
Wniosek:

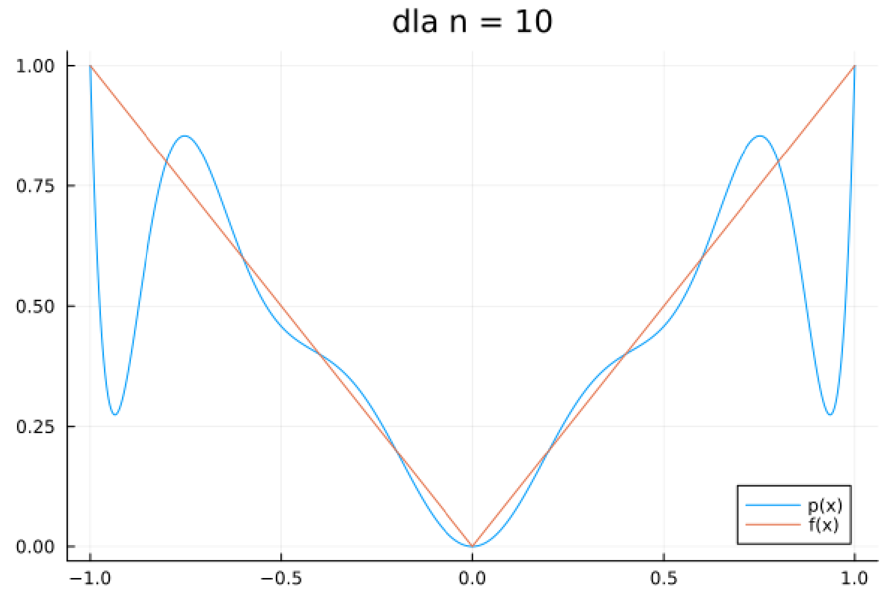
Wyniki wykresów zgadzają się z przewidywaniami - oryginalna funkcja i jej interpolacja praktycznie nakładają się na siebie, co sugeruje, że interpolacja doskonale odwzorowuje pierwotną funkcję, w odróżnieniu od badanej wcześniej funkcji cos(x) To zjawisko występuje z powodu ciągłości i gładkości analizowanych funkcji (ich wszystkie pochodne są również ciągłe). W rezultacie przybliżanie tych funkcji za pomocą wielomianu interpolacyjnego jest niezwykle skuteczne.

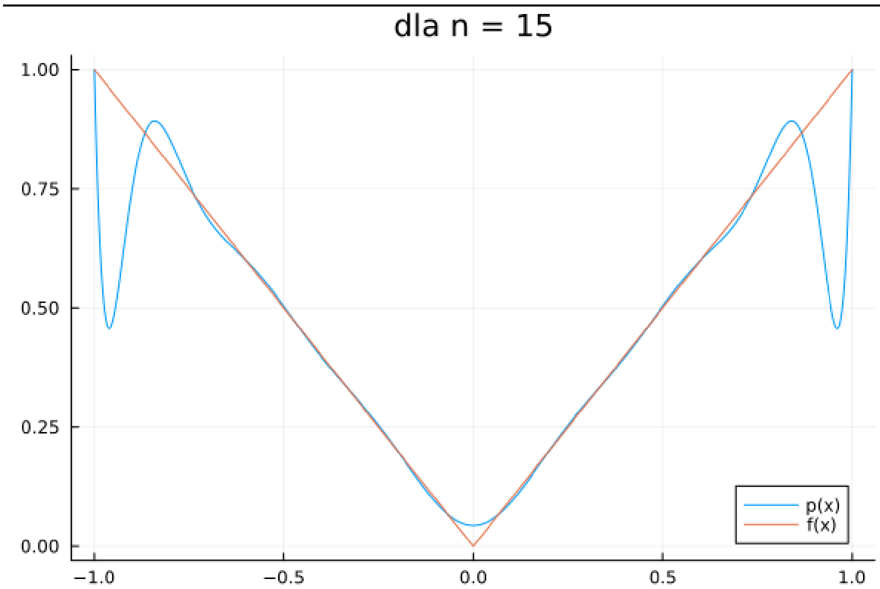
zad.6

Funkcja (a):

$f(x) = |x|, w\ przedziale\ [-1;\ 1]\ , dla\ n = 5, 10, 15.$

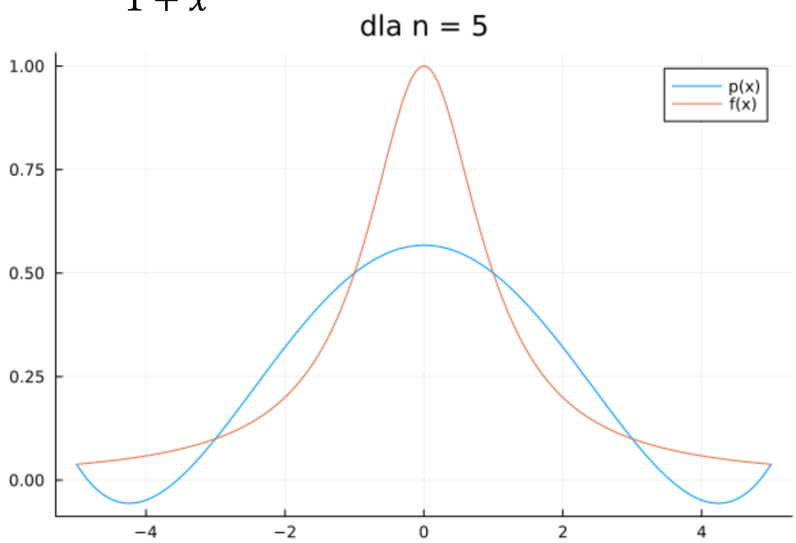


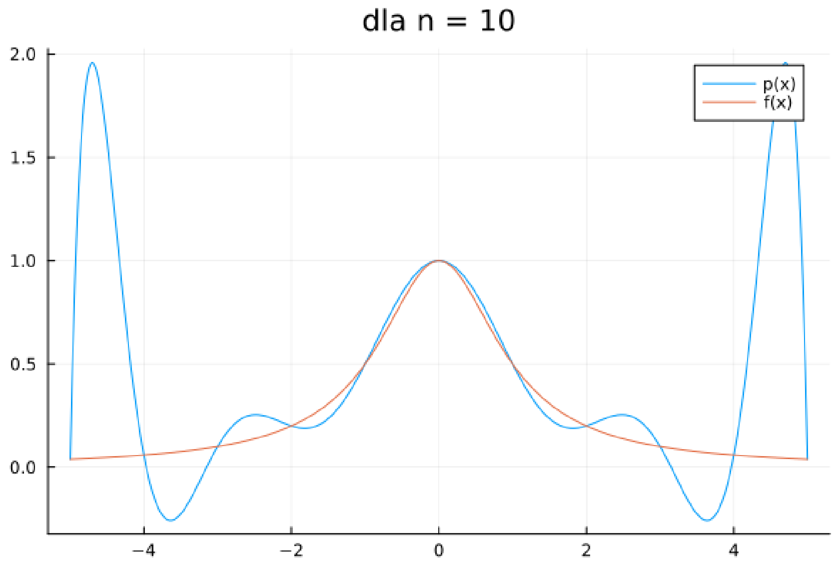


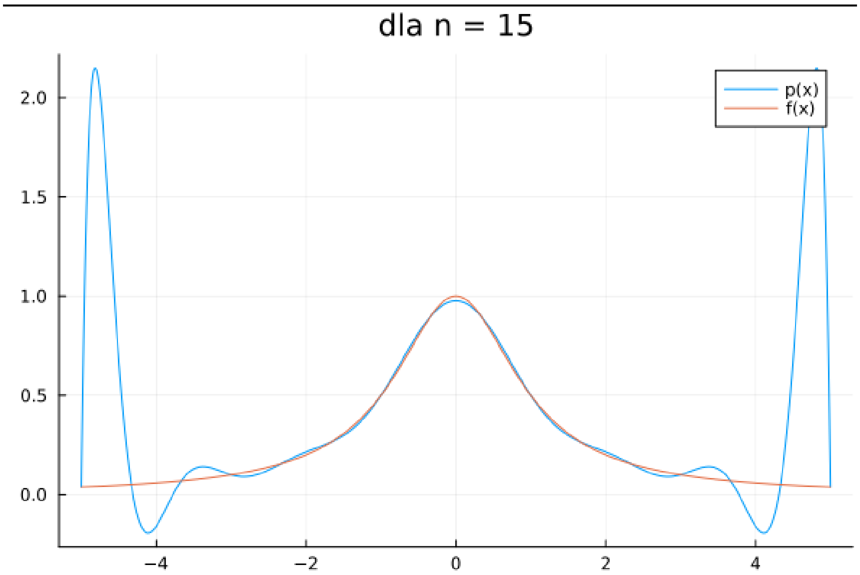


Funkcja (b):

$$f(x) = \frac{1}{1+x^2}\ , w\ przedziale\ [-5;\ 5]\ dla\ n = 5, 10, 15.$$







Wnioski:

W obecnej sytuacji obserwujemy znaczne rozbieżności między wielomianem interpolacyjnym a funkcją wejściową. Pojawia się tu zjawisko Rungego (podobnie jak przy badanej funkcji $cos(x)$), szczególnie widoczne w przypadku funkcji niespójnych ($f(x) = |x|$) lub przy interpolacji funkcji dużego stopnia wielomianu i równoodległych węzłów ($f(x) = \frac{1}{1+x^2}$). Warto zauważyć, że duże odchylenia występują głównie na krawędziach określonego przedziału. Aby zwiększyć precyzję, zaleca się zwiększenie liczby węzłów w tych obszarach, gdzie pojawiają się trudności. Jednym z potencjalnych rozwiązań jest użycie węzłów Czebyszewa, które są uzyskiwane z miejsc zerowych wielomianów Czebyszewa i posiadają większą gęstość na końcach przedziału.