06 grudnia 2023 r.

Małgorzata Mokwa

Grupa laboratoryjna numer 3

METODA BISEKCJI WYZNACZANIA ZER FUNKCJI

$$f(x) = a_0 + a_1|T_1| + a_2|T_2| + \cdots + a_n|T_n|$$

gdzie T_0, …, T_n są wielomianami Czebyszewa pierwszego rodzaju

1. Opis Metody

Do obliczenia miejsc zerowych funkcji powyżej, musimy zaznajomić się z dwoma ważnymi zagadnieniami:

- Wielomiany Czebyszewa pierwszego rodzaju

  Są to wielomiany spełniające warunek rekurencyjny

  $$T_0(x) = 1, \qquad T_1(x) = x, \qquad T_k(x) = 2xT_{k-1}(x) - T_{k-2}(x), \qquad k = 2,3,\ldots$$

  Tych zależności będziemy używać w obliczaniu wartości funkcji w punkcie

- Metoda bisekcji

  Jedna z wielu metod wyznaczania miejsca zerowego funkcji. Metoda ta działa na określonym przedziale [a, b] takim, że f(a) * f(b) < 0. Dodatkowo funkcja musi być ciągła na tym przedziale - mamy wtedy pewność, że znajdziemy miejsce zerowe. Wynika to z twierdzenia, które mówi, że:

  Niech [a, b] ⊂ R oraz niech f : [a, b] → R będzie funkcją ciągłą taką, że f(a)f(b) < 0. Wtedy istnieje co najmniej jedno α, (a < α < b) takie, że f(α) = 0.

  W metodzie bisekcji dzielimy przedział zawierający miejsce zerowe na pół, aż do uzyskania pożądanej dokładności. Przyjmujemy, na przykład, dokładność E = 10^-10 lub jeszcze większą, co gwarantuje bardziej precyzyjne przybliżenie miejsca zerowego. W moim programie, choć metoda bisekcji pierwotnie służy do wyznaczenia jednego miejsca zerowego, dążyć będę do znalezienia więcej niż jednego, jeśli takowe istnieją.

2. Opis działania programu

Program został podzielony na podzadania:

1. **Funkcja Czebyszew:** Funkcja Czebyszew służy do obliczania wartości wektora **x** dla kolejnych wielomianów, a konkretniej ich wartości bezwzględnej. Do tego celu wykorzystuje zależność rekurencyjną dla wielomianów Czebyszewa. W wyniku działania tej funkcji powstaje macierz o length(x) wierszach oraz n kolumnach, gdzie n to maksymalny stopień wielomianu, który jest nam potrzebny.
2. **Funkcja Wartość Funkcji:** Funkcja Wartość funkcji służy do obliczenia wartości funkcji w punkcie. Korzysta z wyznaczonych wcześniej wielomianów Czebyszewa. Każdy obliczony wielomian mnożymy przez odpowiedni element wektora alfa tak, aby dostać i zwrócić ostateczny wynik.
3. **Funkcja Bisekcja:** Służy do wyznaczania zera funkcji.
   Procedura bisekcji jest używana do odnalezienia miejsc zerowych funkcji. Na początku sprawdzamy, czy spełnione są warunki początkowe (zdefiniowane w sekcji 'Opis Metody'). Następnie w pętli dzielimy przedział na pół tak długo, aż jego długość będzie mniejsza lub równa ε, gdzie ε to nasz błąd przybliżenia.
4. **Funkcja Bisekcja2:** Usprawniona wersja bisekcji.
   W tej usprawnionej wersji metody bisekcji dążymy do identyfikacji wszystkich miejsc zerowych dla danej funkcji w określonym przedziale. Realizujemy to poprzez podział przedziału na mniejsze części i zastosowanie metody bisekcji dla każdego z tych podprzedziałów. Dla dostatecznie dużego przedziału oraz odpowiednio dużego parametru k (określającego, na ile podprzedziałów podzielić główny przedział), jesteśmy w stanie znaleźć wszystkie miejsca zerowe funkcji.

3. Przykłady

Oznaczenia:
alfa – wektor współczynników przy kolejnych wielomianach Czebyszewa
a, b – początek i koniec przedziału
Eps – błąd przybliżenia
Iter – ilość wykonanych iteracji, zanim funkcja znalazła miejsce zerowe

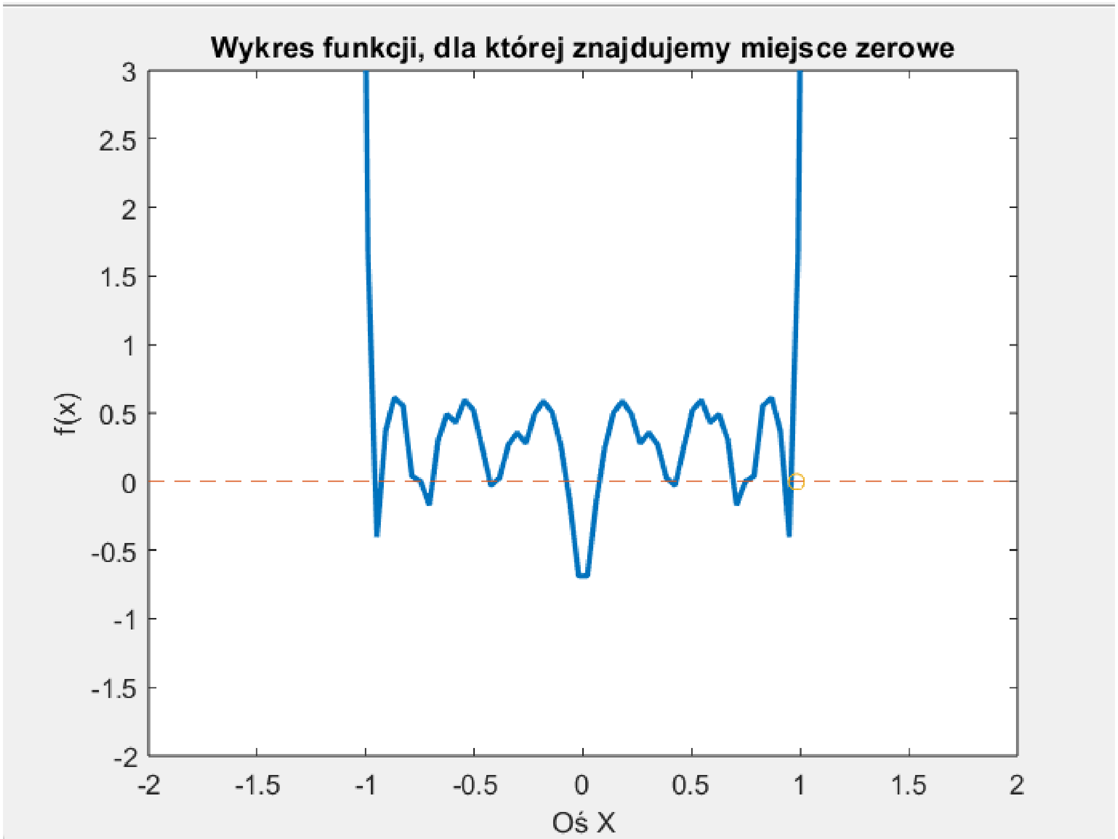
- Przykład numer 1: wyznaczanie miejsc zerowych, gdy są one blisko siebie

```
% -------- wybór funkcji, przedziału i obliczenie jednego miejsca zerowego
% oraz liczby iteracji:
alfa = [-4, 1, 1, 1, 1, 1, 1, 1];
a = 0;
b = 3;
eps = 10^(-5);
[msc_zerowe, iter] = bisekcja(a, b, alfa, eps);
%miejsce zerowe:
msc_zerowe  %%% 0.0740
%liczba iteracji:
iter %%%    18
```

```
bisekcja2(a, b, alfa, eps, 10); %jedno miejscewy
wszystkie_miejsca_zerowe =bisekcja2(a, b, alfa, eps, 50) % wszystkie siedem miejsc zerowych
bisekcja2(a, b, alfa, eps, 60); %pięć miejsc
disp('Wszystkie miejsca zerowe dla przykładu numer 1: ')
wszystkie_miejsca_zerowe
```

```
wszystkie_miejsca_zerowe =

    0.0740    0.4141    0.4394    0.6951    0.7854    0.9270    0.9755
```



Funkcja znalazła jedno miejsce zerowe. Można spróbować znaleźć pozostałe na tym przedziale:

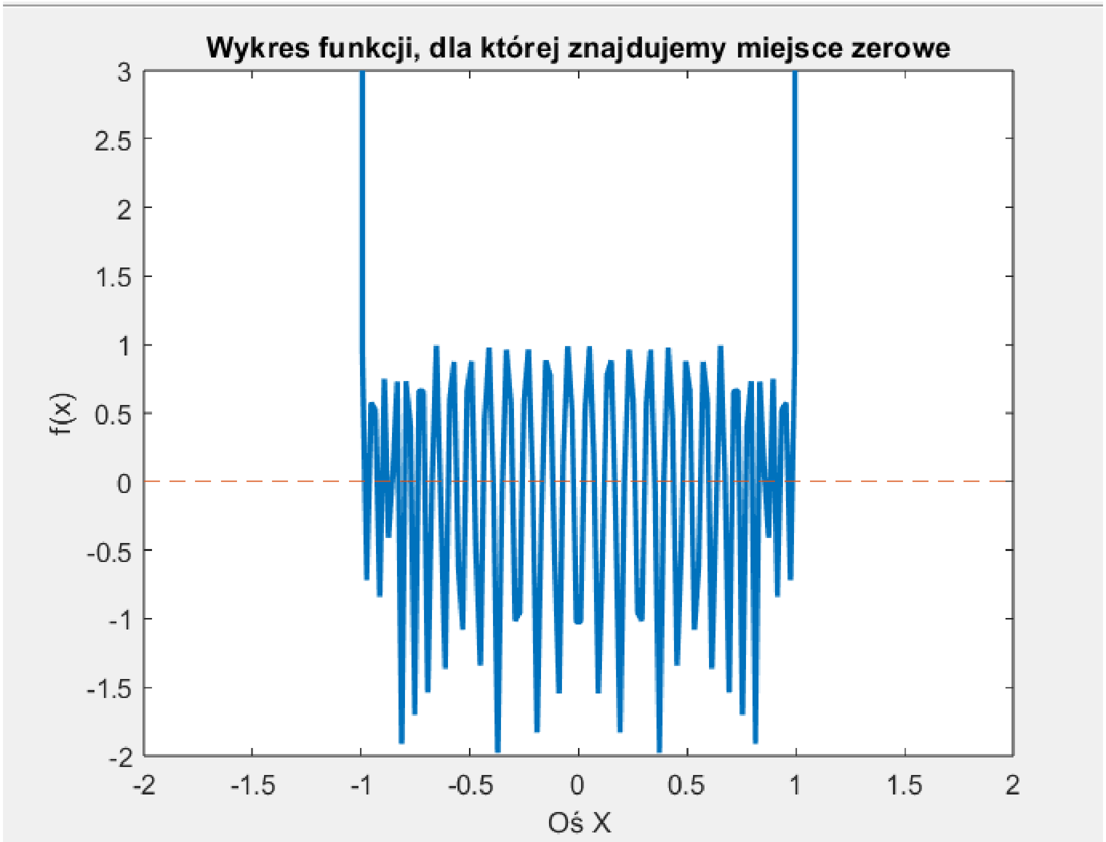
Pierwszy przykład pokazuje, że nie zawsze duża wartość k sprawia, że zostanie wyznaczone więcej miejsc zerowych. Zależy to tylko od wartości funkcji na krańcach przedziału. Dlatego funkcja bisekcja2 znalazła o dwa miejsca zerowe więcej dla k = 50 niż dla k = 60.

To co rzuca się w oczy patrząc na powyższy wykres (i każdy następny) jest fakt, że wszystkie rozpatrywane funkcje są parzyste. Dlatego każde wyznaczone miejsce zerowe możemy przemnożyć przez (-1) i dostać kolejne.

- Przykład numer 2: Wyznaczanie miejsc zerowych, gdy są one blisko oraz weźmiemy duży przedział początkowy

```
%wyznaczamy na dużym przedziale miejsce zerowe
alfa = [-2,0,0,0,0,0,0,0, 0, 0, 0 ,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,3];
a = 0;
b = 20;
eps = 10^(-10);

[msc_zerowe, iter] = bisekcja(a, b, alfa, eps) ;
msc_zerowe %%%     0.7700
iter %%%    37
```



Przykład ten pokazuje, jak trudno jest wyznaczyć wszystkie miejsca zerowe metodą bisekcji. Biorąc k = 40 jesteśmy w stanie wyznaczyć tylko jedno miejsce zerowe, ponieważ w większości podprzedziałów nie ma miejsc zerowych. Biorąc natomiast bardzo duże k = 10000, funkcja wyznacza 33 miejsca zerowe. Zliczając wszystkie zera na rozważanym przedziale okazuje się, że miejsc zerowych jest tylko 27.

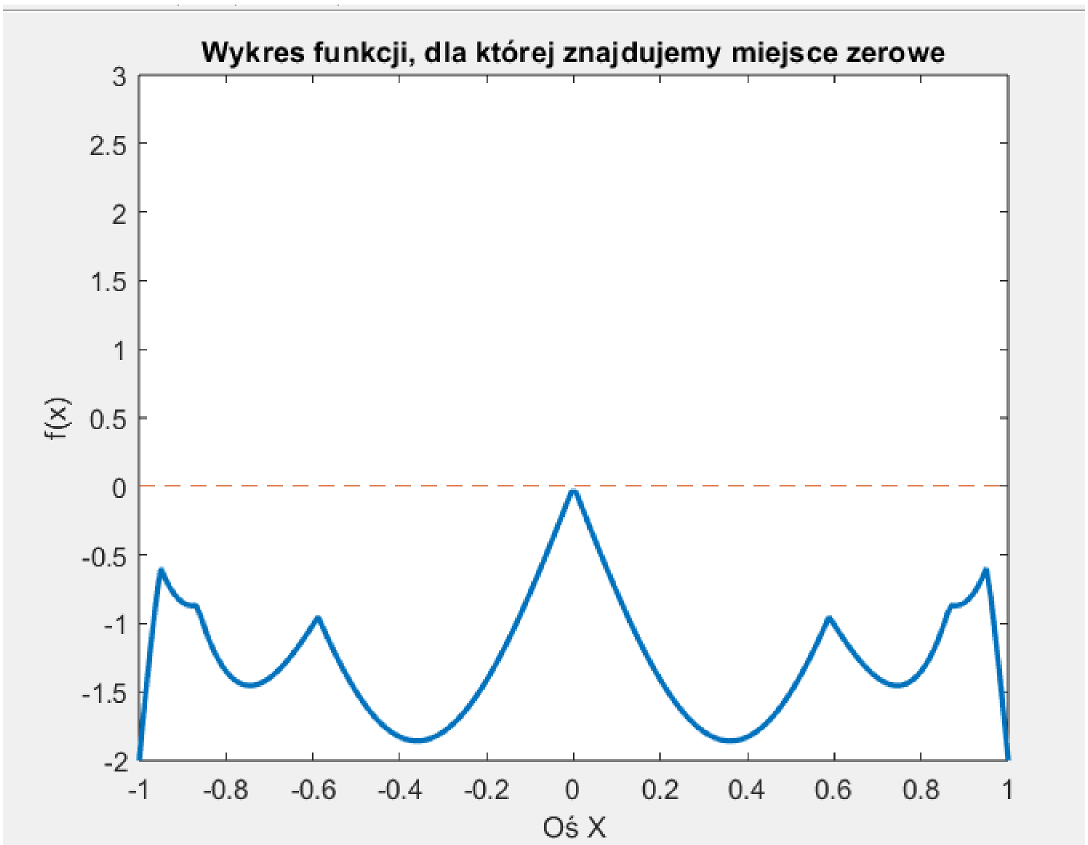
```
%spróbujmy znaleźć więcej miejsc
bisekcja2(a, b, alfa, eps, 40)  %wyznacza tylko jedno miejsce, nawet dla tak dużego k!!!
bisekcja2(a, b, alfa, eps, 10000); % 33 miejsca zerowe
%bisekcja2(a, b, alfa, eps, 400) % 15 miejsc zerowych
```

- Przykład numer 3: Kiedy metoda nie działa?

Pierwszym powodem jest po prostu brak miejsc zerowych. W rozważanych funkcjach wartości wielomianów Czebyszewa są pod wartością bezwzględną, więc są nieujemne. Dlatego jeśli wektor alfa jest w całości dodatni albo ujemny to możemy być pewni, że funkcja jest nieujemna / niedodatnia.

Miejsca zerowego nie wyznaczymy też, gdy funkcja jest nieujemna / niedodatnia. Mimo że może mieć miejsce zerowe, to nigdy nie znajdziemy takich a i b, że f(a) * f(b) < 0.

```
alfa2 = [0, 0, 0, -1, 0, -1]; %funkcja posiada miejsce zerowe, ale nigdy nie zostanie wyznaczone
```



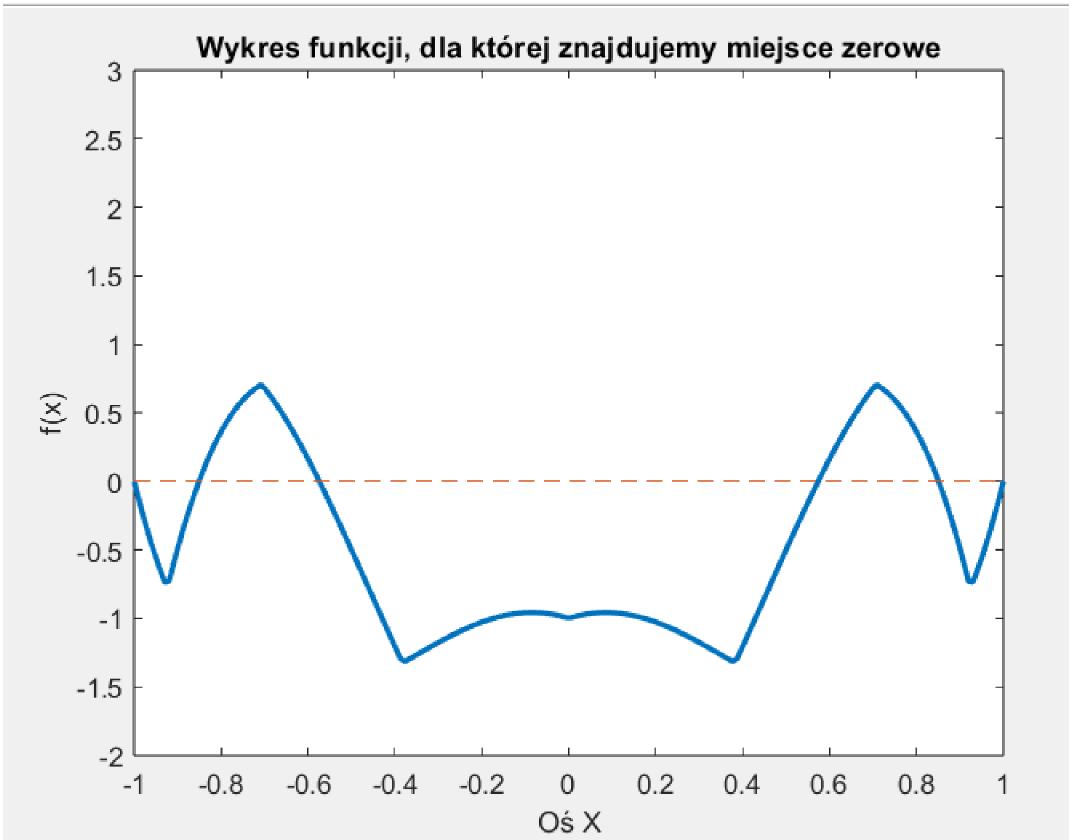
- Przykład numer 4: Miejsce zerowe znajduje się w środku przedziału

  Jest to najbardziej pożądany przypadek, funkcja nie musi zmniejszać przedziału, wynik mamy od razu.

```
alfa = [-1, 1, -1, 0, 1];
a = 0;
b = 2;
eps = 10^(-10);
[miejsce, iter] = bisekcja(a, b, alfa, eps);
% nie trzeba iterować, funckja spełnia założenia pierwszego if-a i pętla zostaje przerwana

%iteracje liczba:
iter  %%% 0

%miejsce zerowe:
miejsce %%% 1
```



Dodatkowo, funkcja bisekcja2 dla k = 6 wyznacza wszystkie miejsca zerowe na przedziale.

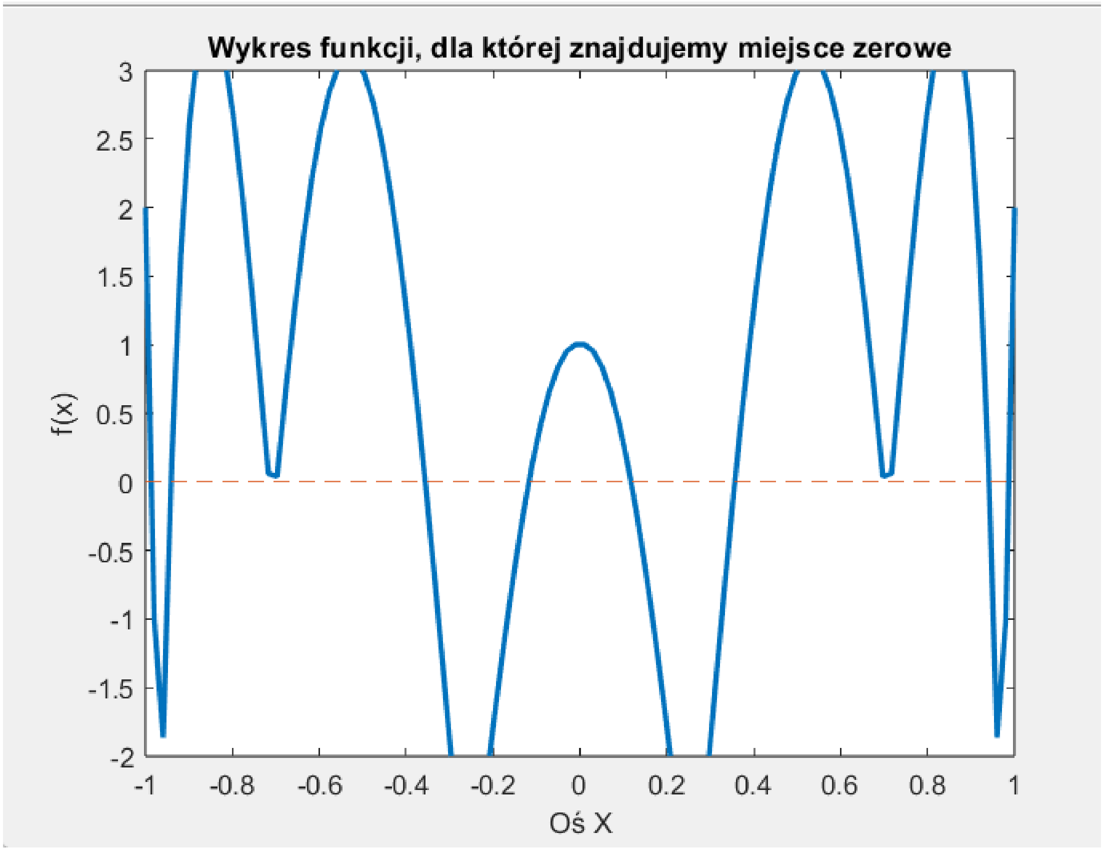
```
miejsca = bisekcja2(a, b, alfa, eps, 6)
```

```
miejsca =

    0.5739    0.8511    1.0000
```

- Przykład numer 5: Zwiększamy wartość parametru ‘Eps’

```
alfa = [-1, 1, -3, 0, 0, 0, 5];
a = -1;
b = 0.3;
eps = 10^(-3);
x = linspace(-1, 1, 100);
[miejsce, iter] = bisekcja(a, b, alfa, eps);
%iter   10
%miejsce    -0.3557
```




```
%blad bezwzgledny i wzgledny:
fun = @(x) wartosc_funkcji(alfa, x);
dokladny = fzero(fun, [a, b]);
dokladny %%%    -0.3562
blad_bezwzgledny = miejsce - dokladny;
```

Wartość Eps ma istotny wpływ na dokładność wyniku. Porównując wynik mojego algorytmu z funkcją wbudowaną w Matlaba wyznaczającą miejsca zerowe okazuje się, że błąd bezwzględny wynosi -0.3557 – (-0.3562) = 0.0005, co porównując z błędami bezwzględnymi z innych przykładów jest bardzo dużą niedokładnością.

Kolejną ciekawą rzecz można zauważyć podczas wywołania funkcji bisekcja2(), która znajduje inne miejsca zerowe na danym przedziale. To jest wynik po wywołaniu:

```
wszystkie_miejsca =

   -0.9878   -0.9407   -0.7148   -0.6986   -0.3557   -0.1184    0.1172
```

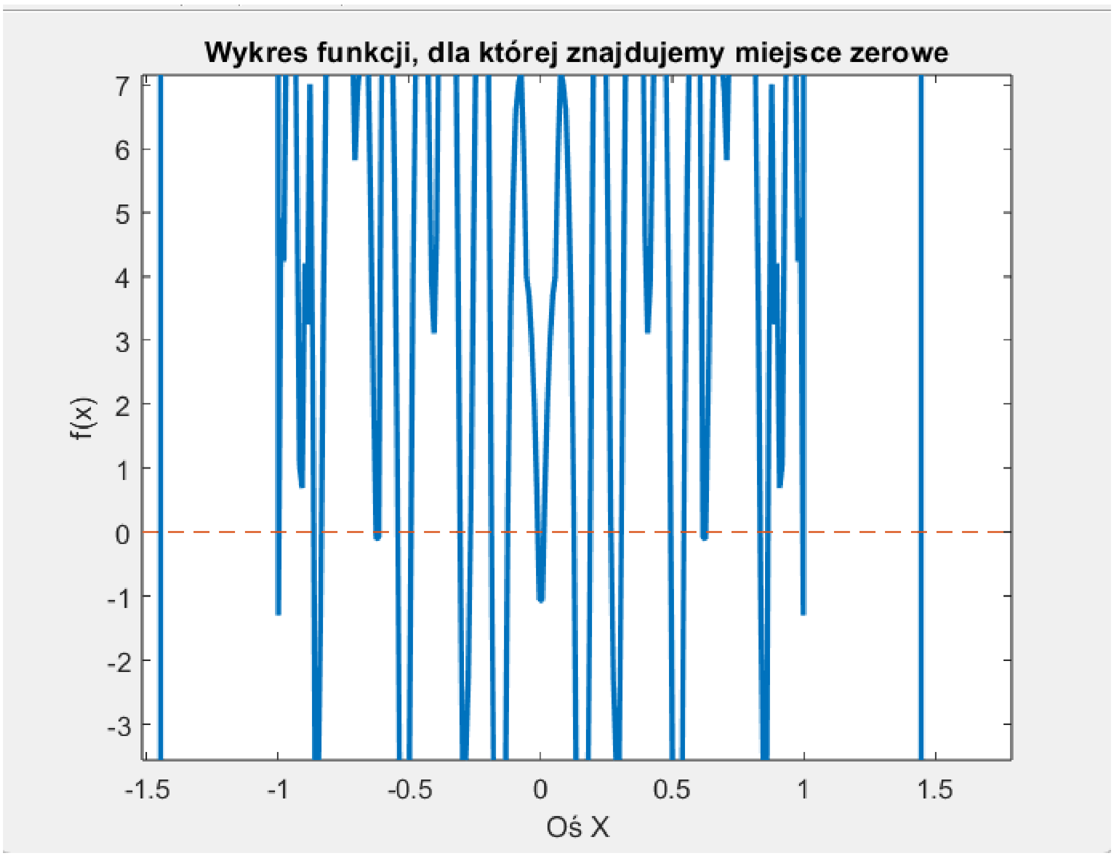
Porównując z wykresem na górze widzimy, że funkcja wyznaczyła wszystkie miejsca zerowe. Jednak oprócz tego wyznaczyła dwa, które nie istnieją. Przez dużą wartość epsilona mogłoby się wydawać, że w punktach (-0.7148,0) oraz (-0.6986,0) są miejsca zerowe, jednak w tych punktach funkcja jest po prostu bardzo bliska zeru. Na wykresie bardzo dobrze widać, że funkcja w otoczeniu tych punktów nawet nie dotyka osi X.

- Przykład numer 6: Bardzo duży wielomian Czebyszewa (dużego stopnia)

  Ostatni już przykład pokaże wykres i miejsca zerowe dla wielomianów Czebyszewa stopnia 27.

```
alfa = [-1, 2,-4, 0, 0, 3, -6, -6, 5, 2, 5, 0, -5, 2, 3, -4, -1, 0, 0, 5, 3, 4, -6, 7, -3, -1, 8, -3]
a = 0.6;
b = 10;
x = linspace(-2, 2, 400)
eps = 10^(-10);

[msc, iter] = bisekcja(a, b, alfa, eps);
%miejsce zerowe:
msc %%% 1.4464
%iteracje liczba:
iter %%%    36
```



Wielomian ten ma 7 miejsc zerowych na przedziale [0.6, 10]. Wyznaczenie ich jest możliwe dla np. k = 2000.

```
miejsca = bisekcja2(a, b, alfa, eps, 2000) %7 miejsc zerowych
```

```
miejsca =

    0.6164    0.6313    0.8349    0.8635    0.9085    0.9161    1.4464
```

4. Analiza, podsumowanie, wnioski

W tabelce poniżej zamieściłam najważniejsze informacje dla każdego przykładu:

- Miejsce zerowe wyznaczone za pomocą funkcji bisekcja()
- Ilość iteracji: ilość wykonanych podziałów przedziału zanim osiągnęliśmy oczekiwany wynik
- Błąd względny / bezwzględny: obliczone za pomocą porównania wyników własnych z funkcją fzero() wbudowaną w Matlaba

	miejsce zerowe	ilość iteracji	błąd względny	błąd bezwzgędny
Przykład 1	0.07398	18	2.306e-06	1.706e-07
Przykład 2	0.77004	37	-7.3161e-11	-5.6337e-11
Przykład 3	333	333	333	333
Przykład 4	1	0	0	0
Przykład 5	-0.35571	10	-0.0013873	0.00049415
Przykład 6	1.4464	36	4.154e-11	6.0085e-11

Wartość 333 jest wpisywana, gdy funkcja nie mogła obliczyć miejsca zerowego. Tak jest w przykładzie numer 3, gdzie rozpatrujemy niedodatnią funkcję.

Wnioski z tabelki:

- Największy błąd bezwzględny jest wtedy, gdy przyjmiemy duży epsilon jako warunek stopu. W przykładzie numer 5 różnica ta jest bardzo widoczna.
- Najlepsze wyniki uzyskujemy, gdy miejsce zerowe znajduje się w środku rozpatrywanego przedziału, wtedy wszystkie błędy wynoszą 0 oraz nie ma potrzeby iterowania i dzielenia przedziału. Jest to jednak bardzo rzadka sytuacja.

- Im większy przedział, tym więcej iteracji się wykonuje. Ilość iteracji jest prawie taka sama dla przykładu 2 i 6, mimo że w przykładzie numer 2 wybraliśmy dwa razy większy przedział. Przykład numer 6 jest jednak dużo bardziej skomplikowany.
- Funkcja bisekcja bardzo dobrze przybliża miejsce zerowe, różnice między wynikiem dokładnym i wynikiem obliczonym różnią się na dalekich pozycjach (oprócz przykładu numer 5, gdzie dokładność jest bardzo mała)

Podsumowanie:

Metoda bisekcji potrafi z bardzo dużą dokładność wyznaczyć miejsce zerowe na danym przedziale. Jednak w porównaniu z innymi metodami (na przykład z metodą Newtona, dla której wykładnik zbieżności wynosi 2) jest dosyć wolna – jej wykładnik zbieżności wynosi 1. Jak zauważyliśmy w przykładach, ma ona pewne ograniczenia

- Dla dużego epsilona może wyznaczać miejsca zerowe, których nie ma
- Dla funkcji niedodatnich nie wyznaczy miejsc zerowych

Nie jest ona również najlepszą metodą do wyznaczania wszystkich miejsc zerowych na przedziale. Jak zauważyliśmy wcześniej, zwiększanie parametru k nie gwarantuje wyznaczenia większej ilości miejsc zerowych. Nie zawsze też wiadomo ile miejsc zerowych ma dana funkcja, dlatego trudno może być zweryfikować, czy wyznaczyliśmy wszystkie zera tej funkcji.