Obliczenia naukowe lista 3

Stanisław Tomkowiak

27 listopada 2024

Zadanie 1

Opis zadania

Napisać funkcję rozwiązującą równanie f (x) = 0 metodą bisekcji.

```
function mbisekcji(f, a::Float64, b::Float64, delta::Float64, epsilon::Float64)
```

Dane wejściowe

- `f` – funkcja f (x) zadana jako anonimowa funkcja (ang. anonymous function)
- `a,b` – końce przedziału początkowego
- `delta,epsilon` – dokładności obliczeń

Dane wyjściowe

- `(r,v,it,err)` – czwórka, gdzie
  - `r` – przybliżenie pierwiastka równania f (x) = 0
  - `v` – wartość f (r)
  - `it` – liczba wykonanych iteracji
  - `err` – sygnalizacja błędu
    - `0` - brak błędu
    - `1` - funkcja nie zmienia znaku w przedziale [a,b]

Opis metody

Aby metoda bisekcji była skuteczna do wyznaczenia zera funkcji w zadanym przedziale, muszą być spełnione dwa warunki:

- funkcja `f(x)` musi być ciągła w tym przedziale
- funkcja `f(x)` musi zmienić znak w zadanym przedziale to znaczy $f(a) * f(b) < 0$.

Pierwiastek obliczamy za pomocą iteracji. W każdej kolejnej iteracji wyznaczamy zero funkcji obliczając $r = \frac{a+b}{2}$. Jeżeli $|b - a| < \delta$ lub $|f(r)| < \epsilon$, to kończymy iterowanie i zwracamy wynik r. W przypadku, gdy warunek nie jest spełniony, wybieramy przedział $[a, r]$ lub $[r, b]$ w zależności, w którym przedziale funkcja zmienia znak, i powtarzamy proces. W ten sposób zawężamy przedział do osiągnięcia odpowiedniej dokładności.

Pseudokod

Metoda bisekcji (`mbisekcji`)

```
function MBISEKCJI(f, a, b, δ, ϵ)
    if b < a then
        Zamień a i b
    end if
    u ← f(a)
    v ← f(b)
    if sign(u) = sign(v) then
        return (0.0, 0.0, 0, 1)
    end if
    k ← 0
    c ← 0
    w ← 0
    e ← b − a
    for k ← 0 to 10^10 do
        e ← e/2
        c ← a + e
        w ← f(c)
        if |e| < δ or |w| < ϵ then
            return (c, w, k, 0)
        end if
        if sign(w) ≠ sign(u) then
            b ← c
            v ← w
        else
            a ← c
            u ← w
        end if
    end for
    return (c, w, k, 0)
end function
```

Zadanie 2

Opis zadania

Napisać funkcję rozwiązującą równanie f (x) = 0 metodą Newtona.

```
function mstycznych(f,pf,x0::Float64, delta::Float64, epsilon::Float64, maxit::Int)
```

Dane wejściowe

- `f, pf` – funkcją f (x) oraz pochodną f (x) zadane jako anonimowe funkcje
- `x0` – przybliżenie początkowe
- `delta,epsilon` – dokładności obliczeń
- `maxit` – maksymalna dopuszczalna liczba iteracji

Dane wyjściowe

- `(r,v,it,err)` – czwórka, gdzie
  - `r` – przybliżenie pierwiastka równania f (x) = 0
  - `v` – wartość f (r)
  - `it` – liczba wykonanych iteracji

- `err` – sygnalizacja błędu
  - `0` - metoda zbieżna
  - `1` - nie osiągnięto wymaganej dokładności w maxit iteracji
  - `2` - pochodna bliska zeru

Opis metody

Metoda Newtona (stycznych) posiada dodatkowy element wejściowy: pochodną funkcji $f(x)$. W tym sposobie obliczenia zer funkcji zaczynamy od x_0 i konstruujemy w tym punkcie styczną do funkcji $f(x_0)$. Kolejnym przybliżeniem jest x_1, czyli x-owa współrzędna przecięcia stycznej z osią X. W ogólności $x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$. Iterujemy tym sposobem do czasu aż $|f(x_n)| < \epsilon$ lub $|x_n| - x_{n-1} < \delta$. Dodatkowy warunek `maxit` pojawia się, ponieważ metoda Newtona nie musi być zbieżna. Gdy wybierzemy x_0 zbyt daleko od faktycznego pierwiastka, możemy nie wyznaczyć prawidłowego pierwiastka. Z tego powodu ustawiamy maksymalną liczbę iteracji.

Pseudokod

Metoda stycznych (`mstycznych`)

```
function MSTYCZNYCH(f, pf, x0, δ, ϵ, maxit)
    v ← f(x0)
    if |v| < ϵ then
        return (x0, v, 0, 0)
    end if
    k ← 0
    x1 ← x0
    for k ← 1 to maxit do
        if |pf(x0)| < ϵ then
            return (0.0, 0.0, 0, 2)
        end if
        x1 ← x0 − v / pf(x0)
        v ← f(x1)
        if |x1 − x0| < δ or |v| < ϵ then
            return (x1, v, k, 0)
        end if
        x0 ← x1
    end for
    return (x1, v, k, 1)
end function
```

Zadanie 3

Opis zadania

Napisać funkcję rozwiązującą równanie f (x) = 0 metodą siecznych.

`function msiecznych(f, x0::Float64, x1::Float64, delta::Float64, epsilon::Float64, maxit::Int)`

Dane wejściowe

- `f, pf` – funkcją f (x) oraz pochodną f (x) zadane jako anonimowe funkcje
- `x0,x1` – przybliżenia początkowe
- `delta,epsilon` – dokładności obliczeń
- `maxit` – maksymalna dopuszczalna liczba iteracji

Dane wyjściowe

- `(r,v,it,err)` – czwórka, gdzie
  - `r` – przybliżenie pierwiastka równania f (x) = 0,
  - `v` – wartość f (r)
  - `it` – liczba wykonanych iteracji
  - `err` – sygnalizacja błędu
    - `0` - metoda zbieżna
    - `1` - nie osiągnięto wymaganej dokładności w maxit iteracji

Opis metody

Metoda siecznych zaczyna się od wybrania dwóch przybliżeń x_0 i x_1. x_2 budujemy za pomocą siecznej przecinającej funkcję $f(x)$ w punktach $(x_0, f(x_0))$ i $(x_1, f(x_1))$, z osią X. W ogólności:

$$x_{n+1} = x_n - \frac{x_n - x_{n-1}}{f(x_n) - f(x_{n-1}} f(x_n).$$

Iterujemy tym sposobem do czasu, aż $|f(x_n)| < \epsilon$ lub $|x_n| - x_{n-1} < \delta$. Dodatkowy warunek `maxit` pojawia się ponieważ metoda siecznych nie musi być zbieżna. Z tego powodu ustawiamy maksymalną liczbę iteracji.

Pseudokod

Metoda siecznych (`msiecznych`)

function MSIECZNYCH($f, x_0, x_1, \delta, \epsilon,$ maxit)
    $f_a \leftarrow f(x_0)$
    $f_b \leftarrow f(x_1)$
    $k \leftarrow 0$
    for $k \leftarrow 1$ **to** maxit **do**
        if $|f_a| > |f_b|$ **then**
            Zamień x_0 z x_1
            Zamień f_a z f_b
        end if
        $s \leftarrow \frac{x_1 - x_0}{f_b - f_a}$
        $x_1 \leftarrow x_0$
        $f_b \leftarrow f_a$
        $x_0 \leftarrow x_0 - f_a \cdot s$
        $f_a \leftarrow f(x_0)$
        if $|x_1 - x_0| < \delta$ **or** $|f_a| < \epsilon$ **then**
            return $(x_0, f_a, k, 0)$
        end if
    end for
    return $(x_0, f_a, k, 1)$
end function

Zadanie 4

Opis zadania

W celu wyznaczenia pierwiastka równania $\sin x - \left(\frac{1}{2}x\right)^2 = 0$ zastosować wcześniej zaprogramowane metody:

- bisekcji z przedziałem początkowym $[1.5; 2]$ i $\delta = \frac{1}{2}10^{-5}$, $\epsilon = \frac{1}{2}10^{-5}$
- Newtona z przybliżeniem początkowym $x_0 = 1.5$ i $\delta = \frac{1}{2}10^{-5}$, $\epsilon = \frac{1}{2}10^{-5}$
- siecznych z przybliżeniami początkowym $x0 = 1$, $x1 = 2$ i $\delta = \frac{1}{2}10^{-5}$, $\epsilon = \frac{1}{2}10^{-5}$

Rozwiązanie

Uruchamiamy odpowiednie funkcje z modułu `ZeroFinding` z zadanymi parametrami.

Wyniki i interpretacja

Metoda	**Przybliżenie** (x)	**Wartość funkcji** ($f(x)$)	**Iteracje** (k)
Bisekcji	1.9337539672851562	-2.702768×10^{-7}	16
Stycznych	1.933753779789742	-2.242331×10^{-8}	4
Siecznych	1.933753644474301	1.564525×10^{-7}	4

Tabela 1: Porównanie wyników metod Bisekcji, Stycznych i Siecznych

Wszystkie metody poprawnie zwróciły wynik funkcji $f(x) = \sin x - \left(\frac{1}{2}x\right)^2$ z podaną dokładnością. Możemy zauważyć, że metody stycznych i siecznych są zauważalnie szybsze niż bisekcji. Dodatkowo wyniki udowadniają współczynnik zbieżności α. Dla metody bisekcji jest on równy $\alpha = 1$, dla stycznych $\alpha = 2$, a dla siecznych $\alpha = \frac{1+\sqrt{5}}{2} \approx 1.618...$

Wnioski

Dzięki współczynnikowi $\alpha > 1$ metody siecznych i Newtona są zauważalnie szybsze od metody bisekcji. Jednak ich wadą jest to, że potrzebujemy do nich dodatkowych danych oraz nie zawsze są zbieżne. Dodatkowym minusem metody Newtona jest konieczność liczenia $f'(x)$.

Zadanie 5

Opis zadania

Metodą bisekcji znaleźć wartości zmiennej x, dla której przecinają się wykresy funkcji $y = 3x$ i $y = e^x$. Wymagana dokładności obliczeń: $\delta = 10^{-4}$, $\epsilon = 10^{-4}$.

Rozwiązanie zadania

Zadanie polega na znalezieniu punktu przecięcia dwóch funkcji, zatem jako pierwsze możemy zapisać: $3x = e^x$. Po odpowiednim przekształceniu otrzymujemy $3x - e^x = 0$, zatem funkcja, którą podstawimy do metody bisekcji znajdowania zera, wygląda następująco $f(x) = 3x - e^x$. Do naszej funkcji $f(x)$ używamy funkcji `mbisekcji` z modułu `ZeroFinding` z parametrami podanymi w zadaniu. Po analizie podanych funkcji wynika, że przecięcie znajduje się pomiędzy 0 i 1 oraz 1 i 2. Uruchamiamy zatem funkcję dwa razy, z wartościami $a = 0$ i $b = 1$ oraz $a = 1$ i $b = 2$.

Wyniki

Miejsce przecięcia	**Przybliżenie** (x)	**Wartość funkcji** ($f(x)$)	**Iteracje** (k)	**Przedział**
1	0.619140625	9.0663×10^{-5}	9	$[0, 1]$
2	1.5120849609375	7.6186×10^{-5}	13	$[1, 2]$

Tabela 2: Wyniki metody Bisekcji dla dwóch miejsc zerowych

Wnioski

Aby znaleźć prawidłowy punkt przecięcia dwóch funkcji metodą bisekcji, konieczna jest analiza ich przebiegu zmienności.

Zadanie 6

Opis zadania

Znaleźć miejsce zerowe funkcji $f_1(x) = e^{1-x} - 1$ oraz $f_2(x) = xe^{-x}$ za pomocą metod bisekcji, Newtona i siecznych. Wymagane dokładności obliczeń: $\delta = 10^{-5}$, $\epsilon = 10^{-5}$.

Sprawdzić co stanie, gdy w metodzie Newtona dla f_1 wybierzemy $x_0 \in (1, \infty]$ a dla f_2 wybierzemy $x_0 > 1$. Co się stanie jak wybierzemy $x_0 = 1$ dla f_2?

Rozwiązanie zadania

Dla podanej funkcji f_1 analizujemy jak dobrać startowe wartości. Mamy:

- Metoda bisekcji $a = 0$, $b = 3$
- Metoda Newtona $x_0 = 0.5$
- Metoda siecznych $x_0 = 0$, $x_1 = 2$

Metoda	**Przybliżenie (x)**	**Wartość funkcji ($f(x)$)**	**Iteracje (k)**
Bisekcja	0.99993896484375	6.1037×10^{-5}	14
Styczne	0.9999850223207645	1.4978×10^{-5}	3
Sieczne	1.0000017597132702	-1.7597×10^{-6}	6

Tabela 3: Porównanie wyników metod Bisekcji, Stycznych i Siecznych

Dla podanej funkcji f_2 analizujemy jak dobrać startowe wartości. Mamy:

- Metoda bisekcji $a = -0.5$, $b = 1.0$
- Metoda Newtona $x_0 = 0.5$
- Metoda siecznych $x_0 = -0.5$, $x_1 = 1.0$

Metoda	**Przybliżenie (x)**	**Wartość funkcji ($f(x)$)**	**Iteracje (k)**
Bisekcja	$-7.62939453125 \times 10^{-6}$	$-7.629452739132958 \times 10^{-6}$	16
Styczne	$-3.0642493416461764 \times 10^{-7}$	$-3.0642502806087233 \times 10^{-7}$	5
Sieczne	$7.367119728946578 \times 10^{-6}$	$7.3670654546934 \times 10^{-6}$	10

Tabela 4: Porównanie wyników metod Bisekcji, Stycznych i Siecznych

Wyniki są poprawne zarówno dla $f_1(x)$ i $f_2(x)$. Zgadzają się z podaną dokładnością.

Przeanalizujmy teraz drugą część zadania. Najpierw przedstawmy wyniki dla $x_0 \in (1, \infty)$:

Przybliżenie (x)	**Wartość funkcji ($f(x)$)**	**Iteracje (k)**	**Kod zakończenia**	**wartość x_0**
0.9999447477307815	5.5254×10^{-5}	3	0	1.5
0.9999999710783241	2.8922×10^{-8}	9	0	3.0
0.9999996427095682	3.5729×10^{-7}	54	0	5.0
-457.6416330443619	1.5330×10^{199}	0	1	7.5
NaN	NaN	0	1	10.0
NaN	NaN	0	1	20.0

Tabela 5: Wyniki metody stycznych dla różnych przypadków x_0

Dla $x_1 \in (1, \infty)$ ilość iteracji w metodzie Newtona rośnie bardzo szybko. Wynika to z tego, że pochodna $f_1'(x) = -e^{1-x}$ bardzo szybko osiąga wartości bliskie 0. Już dla $x_0 = 5.0$ ilość iteracji jest równa 54, a dla $x_0 = 7.5$ kod zakończenia zwraca 1.

Przeanalizujmy teraz $f_2(x)$ dla $x_0 > 1$:

Przybliżenie (x)	**Wartość funkcji ($f(x)$)**	**Iteracje (k)**	**Kod zakończenia**	**wartość x_0**
14.787436802837927	5.5949×10^{-6}	10	0	1.5
14.787436802837927	5.5949×10^{-6}	10	0	3.0
15.19428398343915	3.8272×10^{-6}	9	0	5.0
14.172988816578348	9.9131×10^{-6}	6	0	7.5
14.636807965014	6.4382×10^{-6}	6	0	8.0
100.0	3.7201×10^{-42}	0	0	100.0
500.0	3.5623×10^{-215}	0	0	500.0

Tabela 6: Wyniki metody stycznych dla różnych przypadków x_0

Dla $x_0 > 1$ algorytm zwraca błędny wynik równy około 14.7. Wynika to z tego, że funkcja od wartości $x = 1$, dąży w nieskończoności do 0. Jedyne miejsce zerowe dla funkcji $f_2(x)$ jest równe $x = 0$. Funkcja zwraca wartości $x \approx 14.7$ ponieważ dla tych wartości x spełniony jest warunek $\epsilon < 10^{-5}$.

Ostatni przypadek dla funkcji $f_2(x)$ to przypadek $x_0 = 1$, zwraca on kod zakończenia 2. Znaczy to, że algorytm kończy działanie od razu. Dzieje się tak, ponieważ dla wartości $x_0 = 1$ pochodna funkcji $f_2(x)$ jest równa 0.

Wnioski

Dla metod wyszukiwania zer funkcji należy najpierw przeprowadzić analizę przypadków. W przeciwnym razie możemy dobrać dane, aby metoda zwracała błędny wynik jako prawidłowy.