

Obliczenia naukowe Lista 3

Radosław Wojtczak

Numer indeksu: 254607

Spis treści

0.1	Wstęp do zadań 1-3	1
1	Zadanie 1.	3
1.1	Wstęp	3
1.2	Opis algorytmu	3
2	Zadanie 2.	5
2.1	Wstęp	5
2.2	Opis algorytmu	5
3	Zadanie 3.	7
3.1	Wstęp	7
3.2	Opis algorytmu	7
4	Zadanie 4.	9
4.1	Krótki opis problemu	9
4.2	Rozwiązanie	9
4.3	Wyniki oraz ich interpretacje	9
4.4	Wnioski	10
5	Zadanie 5.	11
5.1	Krótki opis problemu	11
5.2	Rozwiązanie	11
5.3	Wyniki oraz ich interpretacje	11
5.4	Wnioski	12
6	Zadanie 6.	13
6.1	Krótki opis problemu	13
6.2	Rozwiązanie	13
6.3	Wyniki oraz ich interpretacje	14
6.4	Wnioski	15

0.1 Wstęp do zadań 1-3

Główny celem zadań 1-3 było zaimplementowanie podanych przez Pana Profesora funkcji mających na celu szukanie odpowiedniego $r \in R$ dla którego zadana funkcja $f : R \rightarrow R$ przyjmuje wartość 0 ($f(r) = 0$). Ze względu na wykonywanie obliczeń w skończonych arytmetykach (warunkowanych faktem,

iż pamięć w komputerach jest skończona) otrzymane wyniki są na ogół przybliżone, jednak dokładność obliczeń może być z góry określona i dobiera się ją zależnie od potrzeb. (co oznacza, że nie sprawdzamy tak jak w modelu, czy coś jest równe 0, tylko czy otrzymany wynik jest dostatecznie bliski zera. Pojęcie "bliskość" zależy od ustalonych dokładności). Głównym obiektem badań będą funkcje nieliniowe, gdyż wyszukiwanie zer w funkcjach liniowych jest zadaniem trywialnym.

Wszystkie implementacje metod znajdują się w pliku *header.jl* w module *Functions*. Dodatkowo plik *tests.jl* odpowiada za testowanie zaimplementowanych funkcji.

Zadanie 1.

1.1 Wstęp

Pierwszą z funkcji jest funkcja, której działanie opiera się na metodzie bisekcji (metodzie połowienia), której ideą jest dzielenie początkowego przedziału na coraz to mniejsze, w celu znalezienia odpowiedniego przybliżenia miejsca zerowego funkcji.

1.2 Opis algorytmu

Swoje działanie ów metoda opiera na *twierdzeniu Darboux* (które poznaliśmy na analizie matematycznej w trakcie pierwszego roku studiów). Mówiąc kolokwialnie, jeśli funkcja na końcach wybranego przez nas przedziału ma przeciwne znaki i jest ciągła, to znaczy, że w którymś momencie "musiała zmienić znak", to znaczy znaleźć się w punkcie $y = 0$, czyli w miejscu zerowym. Od razu zauważamy, że możliwość użycia tej metody jest warunkowana przez ciągłość funkcji, jak i wybranie przedziału na którego końcach funkcja przyjmuje wartości o różnych znakach. Następnie dzielimy przedział na pół (**UWAGA:** środek przedziału nie jest obliczany w sposób $\frac{(a+b)}{2}$ ze względu na to, iż w pewnych przypadkach możemy paradoksalnie otrzymać liczbę spoza rozpatrywanego przedziału. Z tego względu środek obliczany jest w następujący sposób: $a + \frac{(b-a)}{2}$) i sprawdzamy w której części dalej zachodzi warunek różnych znaków na końcach. Dokonujemy ów podziału do momentu, aż wyznaczony środek podstawiony pod funkcję f zwraca wynik równy 0 (oczywiście, ze względu na ograniczenia sprzętowe nie sprawdzamy, czy otrzymana wartość to dokładnie zero- tutaj wprowadzamy takie pojęcia jak *delta* czy *epsilon*, które informują nas z jaką dokładnością poszukujemy zadanego pierwiastka funkcji). Warunkiem zakończenia pracy algorytmu jest znalezienie takiej wartości r dla której $f(r) = 0$ (z dokładnością do *epsilon*) lub niemożliwość podziału zadanego przedziału na mniejsze (co warunkuje parametr *delta*).

Przykładowy wykres ilustrujący ideę algorytmu:

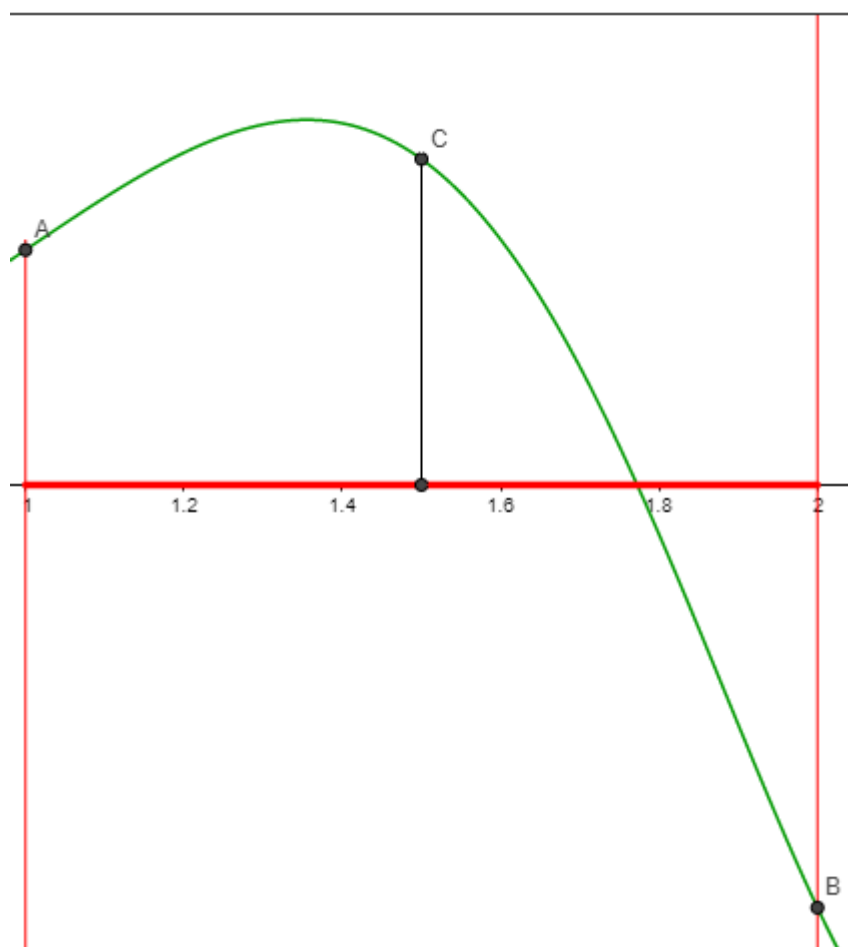


Figure 1.1: Pierwsza iteracja metody bisekcji dla $f(x) = \sin(x)$ w przedziale $[1, 2]$

Zadanie 2.

2.1 Wstęp

Drugą z funkcji jest funkcja, której działanie opiera się na metodzie stycznych (zwanej również metodą *Newtona*), której ideą jest wykorzystanie pojęcia stycznej w celu znalezienia odpowiedniego przybliżenia pierwiastka równania.

2.2 Opis algorytmu

Wybieramy punkt początkowy x_0 i konstruujemy styczną do funkcji w punkcie x_0 . W wyniku tej operacji ta styczna przecina oś OX w punkcie x_1 , który będzie następnym przybliżeniem miejsca zerowego. Następnie powtarzamy powyższy proces aż do momentu, gdy otrzymany wynik jest dostatecznie bliski zera (warunkowane parametrem *epsilon*), odległość między przybliżeniami jest zbyt bliska (warunkowane parametrem *delta*) lub liczba iteracji wykracza poza zakres (liczba iteracji determinowana jest przez parametr *maxit*). Aby powyższy algorytm zadziałał muszą zostać spełnione następujące własności:

- Wyszukiwany pierwiastek jest pierwiastkiem jednokrotnym (bez odpowiednich modyfikacji przy rozwiązywaniu równań zawierających pierwiastki wielokrotne metoda Newtona może być dużo wolniejsza niż inne metody rozwiązywania równań)
- Pierwsza i druga pochodna muszą mieć stały znak w przedziale
- Funkcja musi mieć różne znaki na końcach rozpatrywanego przedziału

Ze względu na wykorzystanie pojęcia stycznej, które jest związane z pojęciem pochodnej (styczną możemy traktować jako geometryczną interpretację pochodnej funkcji), szybko zauważamy, iż wadą ów metody jest potrzeba obliczania pochodnej funkcji.

Przykładowy wykres ilustrujący ideę algorytmu:

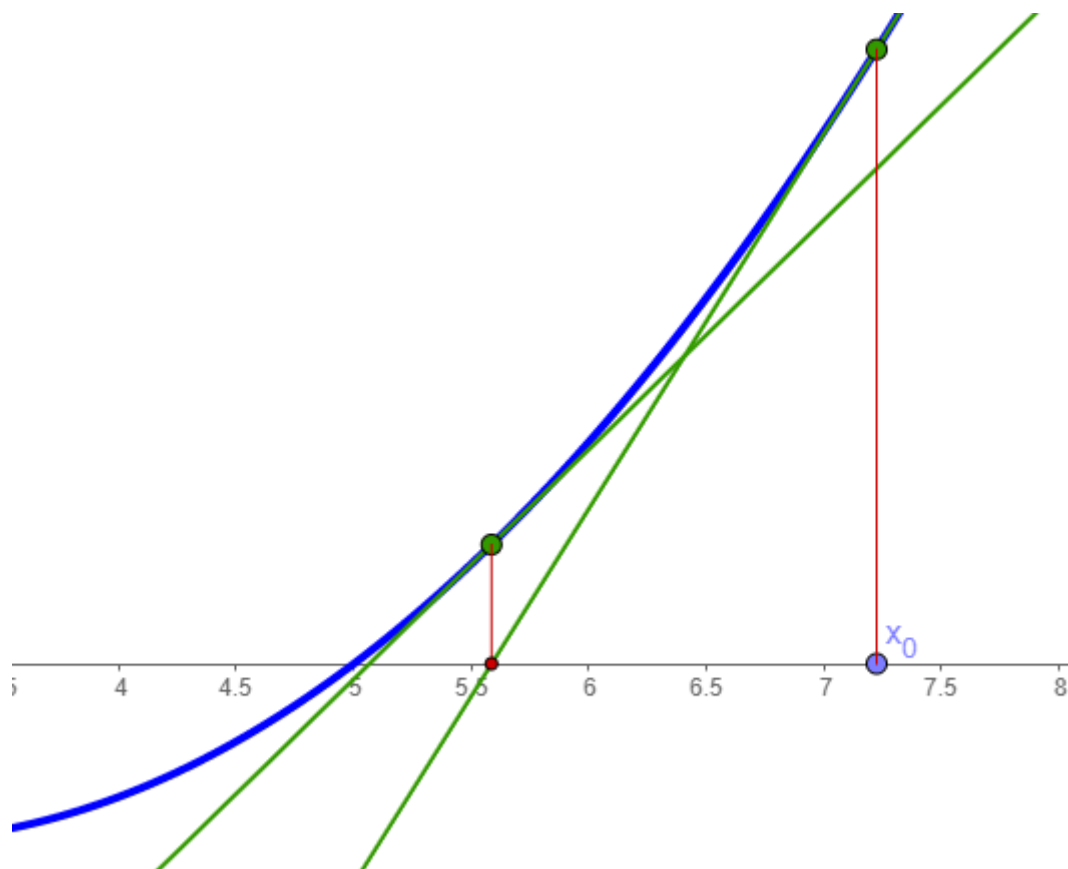


Figure 2.1: Pierwsza dwie iteracje metody Newtona dla $f(x) = (x-3)^2 - 4$ dla $x_0 = 7.25$

Zadanie 3.

3.1 Wstęp

Ostatnią z funkcji jest funkcja, której działanie opiera się na metodzie siecznych (zwanej również metodą *Eulera*), której ideą jest wykorzystanie pojęcia siecznej (prostej przecinającej daną krzywą[tu:wykres funkcji] w co najmniej dwóch punktach) w celu znalezienia odpowiedniego przybliżenia pierwiastka równania.

3.2 Opis algorytmu

Ta metoda jest drobną modyfikacją metody Newtona- zamiast wykorzystywać pochodną, dokonujemy jej aproksymacji przy pomocy ilorazu różnicowego (oczywiście ów ułatwienie spowoduje, iż ta metoda nie zawsze będzie zbieżna). Wybieramy dwa punkty startowe- x_0 oraz x_1 . Konstruujemy sieczną, której przecięcie z osią OX jest kolejnym przybliżeniem miejsca zerowego (dla przykładu niech ów miejsce będzie równe x_2). Następnie powtarzamy procedurę: dla x_1 i x_2 , x_2 i x_3 i tak dalej, aż do momentu, gdy otrzymany wynik jest dostatecznie bliski zera (warunkowane parametrem *epsilon*), odległość między przybliżeniami jest zbyt bliska (warunkowane parametrem *delta*) lub liczba iteracji wykracza poza zakres (liczba iteracji determinowana jest przez parametr *maxit*). Aby powyższy algorytm zadziałał muszą zostać spełnione następujące własności:

- Funkcja musi być określona w przedziale $[a, b]$
- Funkcja musi być ciągła w przedziale $[a, b]$
- Na krańcach przedziału $[a, b]$ funkcja ma różne znaki
- W przedziale $[a, b]$ pierwsza pochodna funkcji jest różna od zera

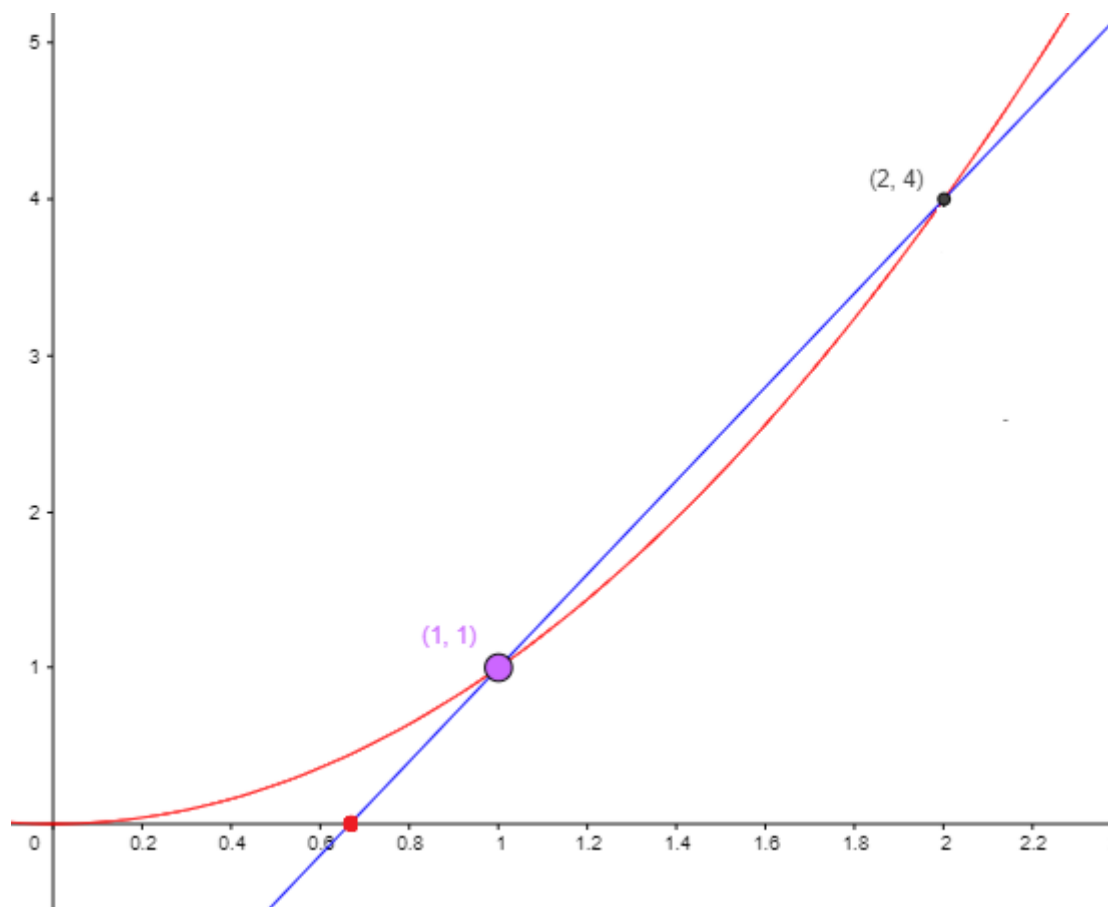


Figure 3.1: Pierwsza iteracja metody Eulera dla $f(x) = x^2$ dla $x_0 = 2$, $x_1 = 1$

Zadanie 4.

4.1 Krótki opis problemu

Dla zadanej funkcji

$$f(x) = \sin(x) - \left(\frac{1}{2} * x\right)^2 \quad (4.1)$$

należało znaleźć miejsca zerowe funkcji przy pomocy wcześniej zaimplementowanych metod.

4.2 Rozwiązanie

Rozwiązanie znajdujące się w pliku **zad4.jl** polega na dosłownej implementacji funkcji podanej w treści zadania oraz sprawdzeniu dla niej przybliżonych miejsc zerowych z podanymi parametrami.

Korzystając z narzędzi matematycznych dostępnych do publicznego użytku spodziewamy się, iż ów miejscem zerowym w wyznaczonym przedziale będzie $x \approx 1.93375$.

4.3 Wyniki oraz ich interpretacje

Poniższa tabela przedstawia wyniki otrzymane w ramach uruchomienia pliku **zad4.jl**

Algorytm	r	$f(r)$	Liczba iteracji
Metoda bisekcji	1.9337539672851562	-2.7027680138402843e-7	16
Metoda stycznych	1.933753779789742	-2.2423316314856834e-8	4
Metoda siecznych	1.933753644474301	1.564525129449379e-7	4

Table 4.1: Otrzymane przybliżenia miejsc zerowych

Interpretacja: Zauważamy, iż dla żadnej z ów metod wartość $f(r)$ nie jest dokładnie równa 0 (czego z perspektywy teorii matematycznej powinniśmy się spodziewać), jednakże wynika to z faktu użycia odpowiednich parametrów określających poprawność rozwiązania. Na tej podstawie wnioskujemy, iż otrzymany wynik jest jedynie przybliżeniem miejsca zerowego rozpatrywanej funkcji, jednak każda z powyższych metod była w stanie zwrócić satysfakcjonujące przybliżenie. Dodatkowo zauważamy, iż metoda bisekcji potrzebowała 2 razy więcej

wywołań niż metoda stycznych czy siecznych, aby rozwiązać otrzymany problem.

4.4 Wnioski

Przy odpowiednich danych początkowych każdy z algorytmów produkuje poprawne rozwiązania z dokładnością do ustalonych parametrów, jednakże ów rozwiązania różnią się liczbą iteracji potrzebną do uzyskania oczekiwanego wyniku, co wynika z różnic w zbieżności między wykorzystanymi metodami.

Zadanie 5.

5.1 Krótki opis problemu

Naszym zadaniem było znalezienie punktów przecięcia funkcji $y = 3x$ z funkcją $y = e^x$. Następnie przy pomocy metody bisekcji mieliśmy znaleźć odpowiednie przybliżenie miejsca zerowego uzyskanej funkcji.

5.2 Rozwiązanie

W celu wyznaczenia odpowiedniego równania przeprowadzamy następujący szereg przekształceń $y = y \longrightarrow e^x = 3 * x \longrightarrow e^x - 3 * x = 0$.

Następnie przy pomocy dostępnych narzędzi matematycznych obliczamy miejsca zerowe otrzymanej funkcji: są to $x_1 \approx 0.619061$ i $x_2 \approx 1.51213$. Dzięki tej wiedzy jesteśmy w stanie odpowiednio spreparować rozpatrywane przedziały.

W pliku **zad5.jl** znajduje się implementacja powyższego rozumowania.

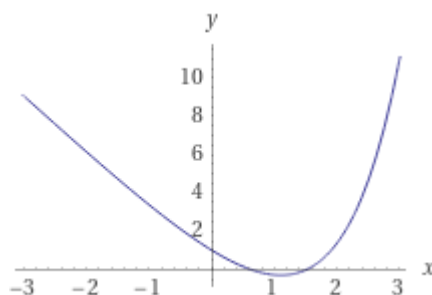


Figure 5.1: Wykres funkcji $e^x - 3 * x$

5.3 Wyniki oraz ich interpretacje

Poniższa tabela przedstawia wyniki otrzymane w ramach uruchomienia pliku **zad5.jl**

Przedział	r	$f(r)$	Liczba iteracji
[0, 1]	0.619140625	-9.066320343276146e-5	9
[1, 2]	1.5120849609375	-7.618578602741621e-5	13

Table 5.1: Otrzymane przybliżenia miejsc zerowych

Interpretacja: W wyniku uruchomienia programu otrzymaliśmy odpowiednie przybliżenia miejsc zerowych, jednakże do tego potrzebowaliśmy wiedzy na temat tego w jakich przedziałach funkcja przyjmuje dodatnie jak i ujemne wartości. Bez tej wiedzy, próbując dowiedzieć się jaki przedział będzie odpowiedni, musielibyśmy w pewien sposób "odgadywać" (np. poprzez iterowanie po przedziałach) przedziały, co w znacznym stopniu spowodowałoby przedłużenie pracy ów metody.

5.4 Wnioski

Metoda bisekcji produkuje poprawne rozwiązania dla dobrze dobranych parametrów początkowych- bez żadnej wiedzy na temat kształtu funkcji istnieje problem z odpowiednim dobraniem parametrów początkowych, gdyż nie potrafimy podać dwóch takich argumentów, dla których funkcja przyjmuje wartości o przeciwnych znakach.

Zadanie 6.

6.1 Krótki opis problemu

W tym zadaniu należało znaleźć przybliżenia miejsc zerowych dwóch następujących funkcji:

$$f_1(x) = e^{1-x} - 1 \quad (6.1)$$

$$f_2(x) = x * e^{-x} \quad (6.2)$$

przy pomocy wcześniej zaimplementowanych metod przy zadanych dokładnościach. Dodatkowym zadaniem było przetestowanie metody stycznych dla konkretnych parametrów początkowych.

6.2 Rozwiązanie

Miejszem zerowym funkcji f_1 jest $x = 1$, natomiast miejscem zerowym funkcji f_2 jest $x = 0$. Odpowiednie parametry początkowe zostały wybrane na podstawie analizy wykresów wyżej wspomnianych funkcji. Implementacja funkcji oraz ich pochodnych, oraz wszystkie poniżej wykonane obliczenia zawarte są w pliku **zad6.jl**

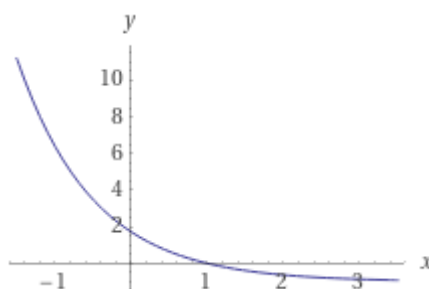


Figure 6.1: Wykres funkcji $f_1(x) = e^{1-x} - 1$

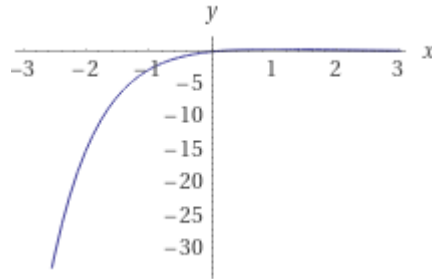


Figure 6.2: Wykres funkcji $f_2(x) = x * e^{-x}$

6.3 Wyniki oraz ich interpretacje

Poniższe tabele przedstawiają wyniki otrzymywane w wyniku uruchomienia programu **zad6.jl**

Metoda;Przedział	r	$f(r)$	Liczba iteracji
Bisekcji;[-1.0, 2.0]	0.9999923706054688	7.629423635080457e-6	17
Bisekcji:[0.0, 2.0]	1.0	0.0	1
Stycznych;0.5	0.999999998878352	1.1216494399945987e-10	4
Stycznych;1.0	1.0	0.0	1
Siecznych:[0.0, 2.0]	1.0000017597132702	-1.7597117218937086e-6	6
Siecznych:[1.0, 2.0]	1.0	0.0	1
Siecznych;[-20.0, 50.0]	49.999999946922074	-1.0	1

Table 6.1: Otrzymane przybliżenia miejsc zerowych dla funkcji $f_1(x) = e^{1-x} - 1$

Metoda;Przedział	r	$f(r)$	Liczba iteracji
Bisekcji;[-1.0, 1.0]	0.0	0.0	1
Bisekcji:[0.0, 2.0]	7.62939453125e-6	7.62933632381113e-6	18
Stycznych;-0.5	-3.0642493416461764e-7	-3.0642502806087233e-7	4
Stycznych;0.0	0.0	0.0	1
Siecznych;[-1.0, 1.0]	1.744165849924562e-8	-1.7441658195034172e-8	18
Siecznych:[0.0, 1.0]	0.0	0.0	1
Siecznych;[-50.0, 100.0]	100.0	3.7200759760208363e-42	1

Table 6.2: Otrzymane przybliżenia miejsc zerowych dla funkcji $f_2(x) = x * e^{-x}$

x_0	r	$f(r)$	Liczba iteracji	Error
1.0	1.0	0.0	1	0
2.0	0.9999999810061002	1.8993900008368314e-8	5	0
3.0	0.9999999710783241	2.892167638712806e-8	9	0
4.0	0.999999995278234	4.721765201054495e-10	21	0
...
11.0	—	—	—	1
12.0	—	—	—	1
13.0	—	—	—	2

Table 6.3: Otrzymane przybliżenia miejsc zerowych dla funkcji 1 dla wariacji podanych w treści zadania z metodą Newtona

x_0	r	$f(r)$	Liczba iteracji	Error
2.0	14.398662765680003	8.036415344217211e-6	10	0
3.0	14.787436802837927	5.594878975694858e-6	10	0
20.0	20.0	4.122307244877116e-8	1	0
1.0	—	—	—	2

Table 6.4: Otrzymane przybliżenia miejsc zerowych dla funkcji 2 dla wariacji podanych w treści zadania z metodą Newtona

Interpretacja: Zauważamy, że znajomość własności funkcji zdecydowanie wpływa na liczbę iteracji wykonaną przez algorytmy ze względu na to, iż z ów wiedzą jesteśmy w stanie odpowiednio dobrać wartości początkowe. Metoda bisekcji mimo zdecydowanie większej liczby iteracji potrzebnych do znalezienia odpowiedzi, dla każdego przetestowanego przykładu zwraca dobry wynik, w odróżnieniu od pozostałych dwóch metod, które dla odpowiednio złych danych zwracają wręcz niewłaściwe wyniki. Zauważamy to między innymi w drugiej części zadania, kiedy sprawdzamy zachowanie metody Newtona dla odpowiednich wartości x_0 . Dla funkcji 1, wybranie odpowiednio oddalonego x_0 od faktycznego miejsca zerowego skutkuje wystąpieniem **ERROR 1**, czyli przekroczenia maksymalnej liczby iteracji, które od wartości $x_0 = 13.0$ przeradza się w **ERROR 2**, który informuje nas o pochodnej bliskiej zero. Również **ERROR 2** otrzymujemy w sytuacji, gdy w funkcji 2 dokonamy podstawienia $x_0 = 1.0$, gdyż faktycznie ów pochodna w tym miejscu jest równa wartości 0. Pozostałe badania dla funkcji 2 pokazują, jak zbieżność lokalna wpływa negatywnie na otrzymane wyniki w sytuacji złego doboru parametrów początkowych

6.4 Wnioski

Każda z powyższych metod ma swoje wady i zalety

- Metoda Bisekcji jest metodą potrzebującą zdecydowanie więcej iteracji niż pozostałe metody (zbieżność liniowa z ilorazem $\frac{1}{2}$). Dodatkowo odpowiedni dobór początkowego przedziału silnie wpływa na liczbę iteracji jaką funkcję potrzebuje aby znaleźć pierwiastek równania (dla powyższego przykładu widzimy, iż znając dokładnie miejsce zerowe możemy dobrać taki przedział,

że funkcja natychmiastowo znajdzie miejsce zerowe, wystarczy że przedział będzie postaci $[r - \frac{\epsilon}{2}, r + \frac{\epsilon}{2}]$, gdzie r oznacza znane nam miejsce zerowe). Zaletą bisekcji jest to, że jeśli tylko dysponujemy dwoma punktami a i b takimi, że f przyjmuje w nich wartości przeciwnych znaków, to ów metoda z pewnością znajdzie miejsce zerowe funkcji, choćby początkowa długość przedziału $|b-a|$ była bardzo duża: wynika to z globalnej zbieżności metody bisekcji.

- Metoda Stycznych(Newtona) mimo małej liczby iteracji (zbieżność z wykładnikiem **2**) jest bardzo podatna na niepoprawne warunki początkowe- jest to związane z lokalną zbieżnością ów metody. Dodatkową wadą metody Newtona jest fakt, iż korzysta ona z pojęcia pochodnej, której obliczenie dla niektórych funkcji może sprawiać problemy (badź nawet być niemożliwe).
- Metoda Siecznych ze względu na swoje podobieństwo do Metody Stycznych również zbiega szybciej niż liniowo (na wykładzie pokazywaliśmy, że wykładnik α dla metody siecznych wynosi tyle samo co tak zwana "złota liczba"- $\frac{1+\sqrt{5}}{2}$) oraz jej zbieżność jest tylko lokalna, co sprawia, że również jest mocno podatna na nieodpowiednie dane wejściowe. Mimo wolniejszej zbieżności metoda siecznych nie korzysta z pochodnej tak jak metoda Newtona. Ponadto, często zdarza się, że wyznaczenie wartości pochodnej, $f'(x_k)$, jest tak samo, albo i bardziej kosztowne od wyznaczenia wartości $f(x_k)$ - w takim wypadku okazuje się, że metoda siecznych jest bardziej efektywna od metody Newtona