Obliczenia naukowe

Lista 3

Jakub Brodziński
229781

1 Zadanie 1

1.1 Opis problemu

Celen zadania było napisanie funkcji rozwiącującej równanie $f(x) = 0$ metodą bisekcji. Sygnatura funkcji powinna być postaci:

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

gdzie $f(x)$ to funkcja zadan ajako anonimowa funkcja, a oraz b to przedziały początkowe, a δ i ϵ to dokładność obliczeń. Jako wynik funkcja powinna zwracać uporządkowną czwórkę (r, v, it, err), gdzie r to przybliżenie pierwiastka funkcji f, v to wartość funkcji $f(r)$, it to liczba wykonanych iteracj, a err to sygnalizacja błędu. err ma wartość 0, jeżeli podczas wykonywania algorytmu nie wystąpił żadnej bład, natomiast 1, gdy funkcja nie zmienia znaku w podanym przeż użytkownika przedziale.

1.2 Opis rozwiązania

Metoda bisekcji, zwana również metodą połowienia przedziału za zadanie ma znalezenie miejsca zerowego funkcji w zadanym przez użytkownika przedziale. Warunkiem koniecznym do prawidłowego działania funkcji jest warunek początkowy metody bisekcji, tj. $f(l)f(r) < 0$, co zostało zapisane w kodzie jako $sgn(f(l)) = sgn(f(r))$. Na mocy twierdzenia Darboux wiemy, że jeżeli zachodzi warunek początkowy, to na przedziale $[l, r]$ istnieje miejsce zerowe funkcji $f(x)$. Jeżeli funkcja na krańcach przedziału podanego przez użytkownika ma rózne znaki to zostaje wykonana *while* algorytmu, gdzie na podstawie $sgn(f(m))$, gdzie m to środek przedziału $[l, r]$ jest podejmowana decyzja czy algorytm w poszukiwaniu miejsca zerowego powininen przeszukiwać lewą część przedziału czy też prawą.
W celu zwiększenia czytelności kodu w przypadku wystąpienia błędu funkcja zwraca *error* zamiast $(0, 0, 0, error_code)$ oraz w przypadku znalezienia miejsca zerowego z odpowiednią dokładnością kod błędu (który w takim przypadku wynosi 0) został pominięty. Dodane przeze mnie sprawdzenie czy δ lub ϵ są mniejsze od zera (w takim przypadku kod błędu wynosi 2) rownież nie zostały uwzględnione w pseudokodzie z uwagi na jakiego czytelność.
Poniżej został przedstawiony pseudokod implementacji metody bisekcji.

```
function mbisekcji(f,a,b,δ,ε)
    l ← a, r ← b
    if sgn(f(l)) = sgn(f(r)) then
        return error
    end if
    dif ← (r − l)/2
    m ← l + dif
    it ← 1
    while |m| > δ and |f(m)| > ε do
        if sgn(f(l)) ≠ sgn(f(m)) then
            r ← m
        else
            l ← m
        end if
        dif ← dif/2
        m ← l + dif
        it ← it + 1
    end while
    return m, f(m), it
end function
```

1.3 Wyniki

W tabeli poniżej zostały przedstawione przykładowe wywołania funkcji jak rownież otrzymane wyniki, dla $f(x) = x^3 - 20$ oraz $g(x) = x^2 - 16$.

nr	$Wywolanie$	r	v	it	err
1	$mbisekcji(f,\ 2.0,\ 3.0,\ -2.0e-5,\ 0.5e-5)$	0	0	0	2
2	$mbisekcji(f,\ 0.0,\ 1.0,\ 0.5e-5,\ 0.5e-5)$	0	0	0	1
3	$mbisekcji(f,\ 0.0,\ 1.0,\ 0.5e-5,\ 0.5e-5)$	2.7144174575805664	$-3.5148828203546145e-6$	20	0
4	$mbisekcji(g,\ 3.5,\ 4.5,\ 0.5e-5,\ 0.5e-5)$	4.0	0.0	1	0

Dwa pierwsze wywołania to sprawdzenie czy implementowany algorytm zwraca odpowiednie kody błędu. W pierwszym przypadku jedna ze stałych wyrażających dokładność obliczeń była mniejsza niż 0, natomiast w drugim przypadku $f(0)f(1) > 0$, dwie ostatnie iteracje zakończyły się sukcesem.

2 Zadanie 2

2.1 Opis problemu

Celen zadania było napisanie funkcji rozwiącującej równanie $f(x) = 0$ metodą Newtona, zwaną metodą stycznych. Sygnatura funkcji powinna być postaci:

$$function\ mstycznych(f, pf, x_0 :: Float64, delta :: Float64, epsilon :: Float64, maxit :: Int)$$

gdzie f to funkcja zadan jako anonimowa funkcja, natomiast pf to jej pochodna, x_0 to przybliżenie początkowe, δ i ϵ to dokładność obliczeń,a $maxit$ to maksymalna liczba iteracji, których algorytm ma wykonać. Jako wynik funkcja powinna zwracać uporządkowną czwórkę (r, v, it, err), gdzie r to przybliżenie pierwiastka funkcji f, v to wartość funkcji $f(r)$, it to liczba wykonanych iteracj, a err to sygnalizacja błędu. err ma wartość 0, jeżeli podczas wykonywania algorytmu nie wystąpił żadnej bład, gdy w $maxit$ iteracji algorytm nie uzyskał wymaganej dokładności zwracany kod błedu to 1, natomiast wartośc 2 jest zarezerwowana dla przypadku, gdy pochodna jest bliska zeru.

2.2 Opis rozwiązania

Metoda Newtona przy każdej iteracji na podstawie x_n liczy kolejne przybliżenie x_n ze wzoru rekurencyjnego $x_n = x_p - f(x_p)/pf(x_p)$, gdzie $x_p = x_{n-1}$. Wartość pochodnej w punkcie x_n jest porównwana do *machepsa* czyli najmniejszą liczbę spełniająca równanie $fl(1.0 + macheps) > 1.0$, w przypadku, gdy $|pf(x_n)| < macheps$, pochodną uznajemy za bliską zeru i zwracamy odpowiedni kod błędu.
W celu zwiększenia czytelności kodu w przypadku wystąpienia błędu funkcja zwraca *error* zamiast $(0, 0, 0, error_code)$ jak rownież w przypadku znalezienia miejsca zerowego z odpowiednią dokładnością kod błędu (który w takim przypadku wynosi 0) został pominięty. Dodane przeze mnie sprawdzenie czy δ lub ϵ są mniejsze od zera (w takim przypadku kod błędu wynosi 3) rownież nie zostały uwzględnione w pseudokodzie z uwagi na jakiego czytelność.
Poniżej został przedstawiony pseudokod implementacji metody Newtona.

function $mstycznych(f, pf, x_0, \delta, \epsilon, maxit)$
    $x_n \leftarrow x_0$
    for $it = 1$ **to** $maxit$ **do**
        if $|pf(x_n)| < macheps$ **then**
            return $error$
        end if
        $x_p \leftarrow x_n$
        $x_n \leftarrow x_n - f(x_n)/pf(x_n)$
        if $|x_n - x_p| < \delta$ or $|f(x_n)| < \epsilon$ **then**
            return $x_n, f(x_n), it$
        end if
    end for
    return (x_n, x_n_value, it)
end function

2.3 Wyniki

W tabeli poniżej zostały przedstawione przykładowe wywołania funkcji jak rownież otrzymane wyniki, dla $f(x) = x^3 - 20$ oraz $g(x) = x^2 - 16$, gdzie $pf(x) = 3x^2$ oraz $pg(x) = 2x$.

nr	*Wywolanie*	*r*	*v*	*it*	*err*
1	$mstycznych(f,\ df,\ 0.0,\ -2.0e-5,\ 0.5e-5,\ 10)$	0	0	0	3
2	$mstycznych(f,\ df,\ 0.0,\ 0.5e-5,\ 0.5e-5,\ 10)$	(0	0	1	2
3	$mstycznych(f,\ df,\ 2.0,\ 0.5e-5,\ 0.5e-5,\ 2)$	2.740740740740741	0.587512066250067	2	1
4	$mstycznych(f,\ df,\ 2.0,\ 0.5e-5,\ 0.5e-5,\ 40)$	2.714417639988567	$5.170978987223407e-7$	4	0
5	$mstycznych(g,\ dg,\ 3.5,\ 0.5e-5,\ 0.5e-5,\ 40)$	4.0000000031214755	$2.4971804180040635e-8$	3	0

Trzy pierwsze wywołania to sprawdzenie czy implementowany algorytm zwraca odpowiednie kody błędu. W pierwszym przypadku jedna ze stałych wyrażających dokładność obliczeń była mniejsza niż 0, w drugim przypadku $df(x) \approx 0$, natomiast w trzecim przypadku bo zadanej liczbie iteracji algorytm nie był w stanie osiagnąć wymaganej dokładności, dwie ostatnie iteracje zakończyły się sukcesem.

3 Zadanie 3

3.1 Opis problemu

Celen zadania było napisanie funkcji rozwiącującej równanie $f(x) = 0$ metodą siecznych. Sygnatura funkcji powinna być postaci:

$$function\ mstycznych(f, x_0 :: Float64, x_1 :: Float64, delta :: Float64, epsilon :: Float64)$$

gdzie $f(x)$ to funkcja zadana jako anonimowa funkcja, x_0 oraz x_1 to przedziały początkowe, δ i ϵ to dokładność obliczeń, a *maxit* to maksymalna liczba iteracji, których algorytm ma wykonać. Jako wynik funkcja powinna zwracać uporządkowną czwórkę (r, v, it, err), gdzie r to przybliżenie pierwiastka funkcji

f, v to wartość funkcji $f(r)$, it to liczba wykonanych iteracj, a err to sygnalizacja błędu. err ma wartość 0, jeżeli podczas wykonywania algorytmu nie wystąpił żadnej bład, natomiast 1, gdy w zadanej liczbie iteracji algorytm nie osiągnął wymaganej dokładności.

3.2 Opis rozwiązania

Metoda siecznych podobnie jak metoda Newtona do obliczenia kolejnego przybliżenie wykorzystuje wzór rekurencyjny. Dużą zaletą tej metody jest fakt, że w odróżnieniu do metody Newtona nie jest wymagana znajomość pochodnej danej funkji $f(x)$. Wzór który jest wykorzystuwany do policzenia kolejnych przybliżeń to: $x_{n+1} = x_n - f(x_n) * \frac{(x_n - x_{n-1})}{(f(x_n) - f(x_{n-1}))}$.
W każdej iteracji pętli for sprawdzamy czy jest zachowana nierówność $|f(x_n)| < |f(x_{n-1})|$. Jeżeli nie, to zamieniamy wartościami x_n z x_{n-1}, aby zachować poprawność formy równania rekurencyjnego. Prócz liczenie kolejnych przybliżeń x_n sprwadzana jest rownież dokładność, którą w danej iteracji mamy i czy jest ona wystarczająca zadowlająca, aby została zwrocona użytkownikowi.
W celu zwiększenia czytelności kodu w przypadku wystąpienia błędu funkcja zwraca $error$ zamiast $(0, 0, 0, error_code)$, jak rownież w przypadku znalezienia miejsca zerowego z odpowiednią dokładnością kod błędu (który w takim przypadku wynosi 0) został pominięty. Dodane przeze mnie sprawdzenie czy δ lub ϵ są mniejsze od zera (w takim przypadku kod błędu wynosi 2) rownież nie zostały uwzględnione w pseudokodzie z uwagi na jakiego czytelność.
Poniżej został przedstawiony pseudokod implementacji metody Newtona.

function $msiecznych$(f,x_0,x_1,δ,ϵ,$maxit$)
  $x_n \leftarrow x_1$
  $x_{n-1} \leftarrow x_0$
  for $it = 1$ **to** $maxit$ **do**
    if $|f(x_n| < |f(x_{n-1})|$ **then**
      $swap(x_n, x_{n-1})$
    end if
    $s \leftarrow (x_n - x_{n-1})/(f(x_n) - f(x_{n-1}))$
    $x_{n-1} \leftarrow x_n$
    $x_n \leftarrow x_n - f(x_n) * s$
    if $|x_n - x_{n-1}| < \delta$ or $|f(x_n)| < \epsilon$ **then**
      return $x_n, f(x_n), it$
    end if
  end for
  return $error$
end function

3.3 Wyniki

W tabeli poniżej zostały przedstawione przykładowe wywołania funkcji jak rownież otrzymane wyniki, dla $f(x) = x^3 - 20$ oraz $g(x) = x^2 - 16$.

nr	$Wywolanie$	r	v	it	err
1	$msiecznych(f,\ 2.0,\ 3.0,\ -2.0e-5,\ 0.5e-5,\ 10)$	0	0	0	2
2	$msiecznych(f,\ 0.0,\ 1.0,\ 0.5e-5,\ 0.5e-5,\ 10)$	1.0451306413301662	-18.85840583069651	2	1
3	$msiecznych(f,\ 2.0,\ 3.0,\ 0.5e-5,\ 0.5e-5,\ 2)$	2.7144176165194542	$-1.6678143310855376e-9$	5	0
4	$msiecznych(g,\ 3.5,\ 4.5,\ 0.5e-5,\ 0.5e-5,\ 10)$	3.999999997855693	$-1.715445563377216e-8$	4	0

Dwa pierwsze wywołania to sprawdzenie czy implementowany algorytm zwraca odpowiednie kody błędu. W pierwszym przypadku jedna ze stałych wyrażających dokładność obliczeń była mniejsza niż 0, natomiast w trzecim przypadku bo zadanej liczbie iteracji algorytm nie był w stanie osiagnąć wymaganej dokładności, dwie ostatnie iteracji zakończyły się sukcesem.

4 Zadanie 4

4.1 Opis problemu

Celem zadania było wyznaczenie pierwiastka równania $f(x) = sin(x) - (\frac{1}{2}x)^2$ przy użyciu metod zaimplementowanych w trzech wcześniejszych zadaniach.

4.2 Opis rozwiązania

Funkcja $f(x)$, przedziały, przybliżenia początkowe wraz z odpowiadającymi jej precyzjami zostały dodane do kodu programu, a następnie wykorzystane jako argumenty do wywołania odpowiednich funkcji. Za pochodną funkcji f przyjąłem funkcje określoną wzorem $f'(x) = cos(x) - \frac{x}{2}$.

4.3 Wyniki

Wyniki dla poszczegółncyh algorytmów zostały przedstawione w tabeli poniżej. Zgodnie z notacją r oznacza przyblizenie miejsca zerowego, v to przybliżenie funkcji w punkcie r, it to liczba wykonanych iteracji, a err odpowiadaj numerowi błędu. Obliczenia zostały przeprowadzone dla $\delta = 0.5e - 5$ oraz $\epsilon = 0.5e - 5$.

$Wywolanie$	r	v	it	err
$mbisekcji(f,\ 1.5,\ 2.0,\ 0.5e-5,\ 0.5e-5)$	1.9337539672851562	$-2.7027680138402843e-7$	16	0
$mstycznych(f,\ 0.0,\ 1.0,\ 0.5e-5,\ 0.5e-5,\ 25)$	1.933753779789742	$-2.2423316314856834e-8$	4	0
$msiecznych(f,\ 1.0,\ 2.0,\ 0.5e-5,\ 0.5e-5,\ 25)$	1.933753644474301	$1.564525129449379e-7$	4	0

4.4 Wnioski

Najmniej dokładny wynik został otrzymany przy użyciu metody bisekcji, został on również uzyskany przy największej liczbie iteracji, może to wynikać m.in. z liniowej zbieżności funkcji bisekcji. Najbardziej dokładną metodą w tym przypadku była metoda Netwona, która była dokładniejsza od metody siecznych o $1.3402919663008106e - 7$.

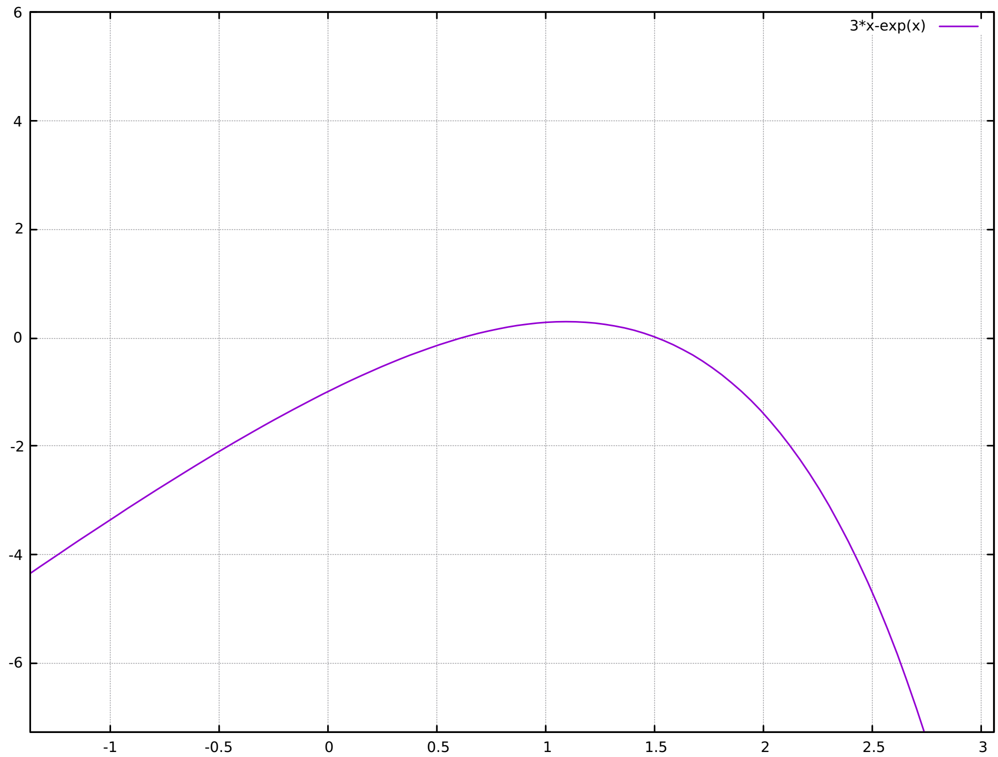
5 Zadanie 5

5.1 Opis problemu

Celem zadania było znalezenie wartości x dla których funkcje $y_1 = 3x$ oraz $y_2 = e^x$ uzyskują tą samą wartość. Dokładność która jest wymagana do tych obliczeń to $\sigma = 10^{-4}$ oraz $\epsilon = 10^{-4}$.

5.2 Opis rozwiązania

Do rozwiązania zadania została użyta implementacja metody bisekcji z zadania 1. Funkcja miała za zadanie znaleźć miejsce zerowe nowej funkcji $y = 3x - e^x$, której to miejsca zerowe są punktami przecięcia funkcji y_1 oraz y_2 zadanych w poleceniu. W oszacowaniu przedziału $[a, b]$, gdzie $f(a)f(b) < 0$ pomógł wykres wygenerowany w *GNUPlot* przedstwiony poniżej.



Rysunek 1: $f(x) = 3x - e^x$ przy użyciu $GNUPlot'a$

5.3 Wyniki

Zostały wykonane dwa wywołania *mbisekcji* dla różnych przedziałów początkowych. W pirwszym przypadku był to $[-2.0, 1.0]$, natomaist w drugim $[1.0, 5.0]$. Wyniki zostały przedstawione w tabeli poniżej. Zgodnie z notacją r oznacza przyblizenie miejsca zerowego, v to przybliżenie funkcji w punkcie r, it to liczba wykonanych iteracji, a err odpowiadaj numerowi błędu. Obliczenia zostały przeprowadzone dla $\delta = 10^{-4}$ oraz $\epsilon = 10^{-4}$.

$Wywolanie$	r	v	it	err
$mbisekcji(f,\ -2.0,\ 1.0,\ 10^{-4},\ 10^{-4})$	0.619140625	$9.066320343276146e-5$	9	0
$mbisekcji(f,\ 1.0,\ 5.0,\ 10^{-4},\ 10^{-4})$	1.5120849609375	$7.618578602741621e-5$	15	0

5.4 Wnioski

Metoda bisekcji znalazła przybliżenia dwóch różnych pierwiastków. Na podstawie wyników można zaobserwować, że im lepiej sprecyzujemy przedział, tym liczba iteracji jest mniejsza. Zaletą metody bisekcji jest to, że nie musimy się martwić czy pochodna funkji w danym punkcie zbiega do zera czy też nie, jak w przypadku metody Newtona, a jedynie znakiem funkcji w danym punkcie, a w przypadku gdy podany przedział jest zbyt szeroki (przy założeniu że $f(a)f(b) < 0$) to nadal otrzymamy odpowiedni wynik, lecz tym razem przy większej liczbie iteracji.

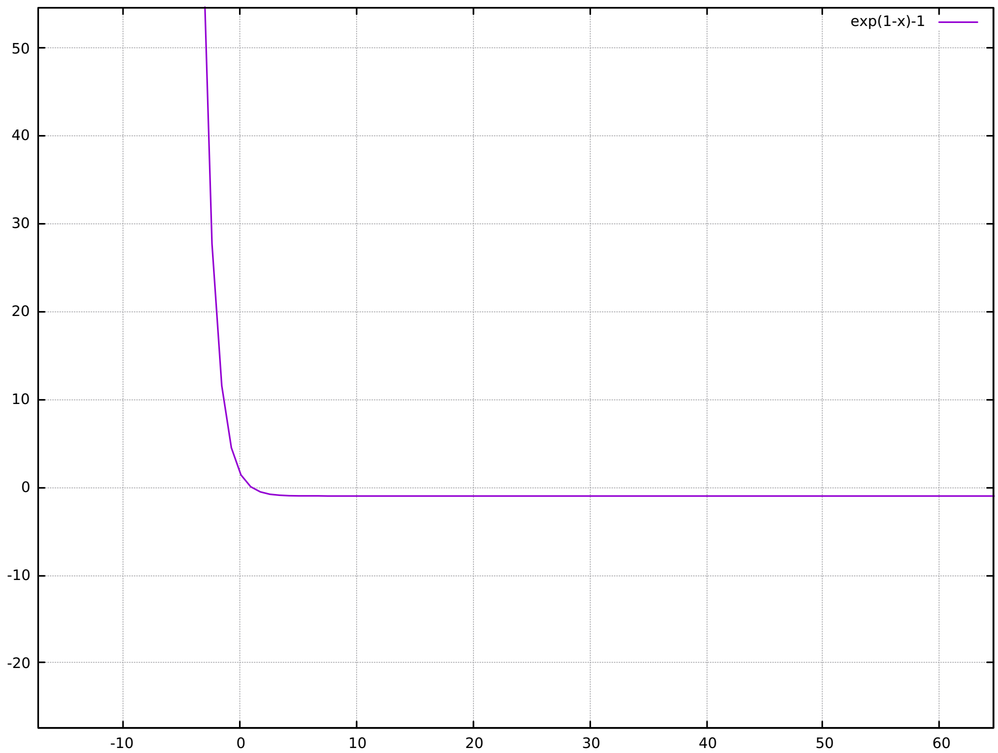
6 Zadanie 6

6.1 Opis problemu

Celem zadania było znalezienie miejsca zerowego funkcji $f_1(x) = e^{1-x} - 1$ oraz $f_2(x) = xe^{-x}$ za pomoca metod bisekcji, Newtona oraz siecznych. Wymagana dokładność to $\delta = 10^{-5}$ oraz $\epsilon = 10^{-5}$.

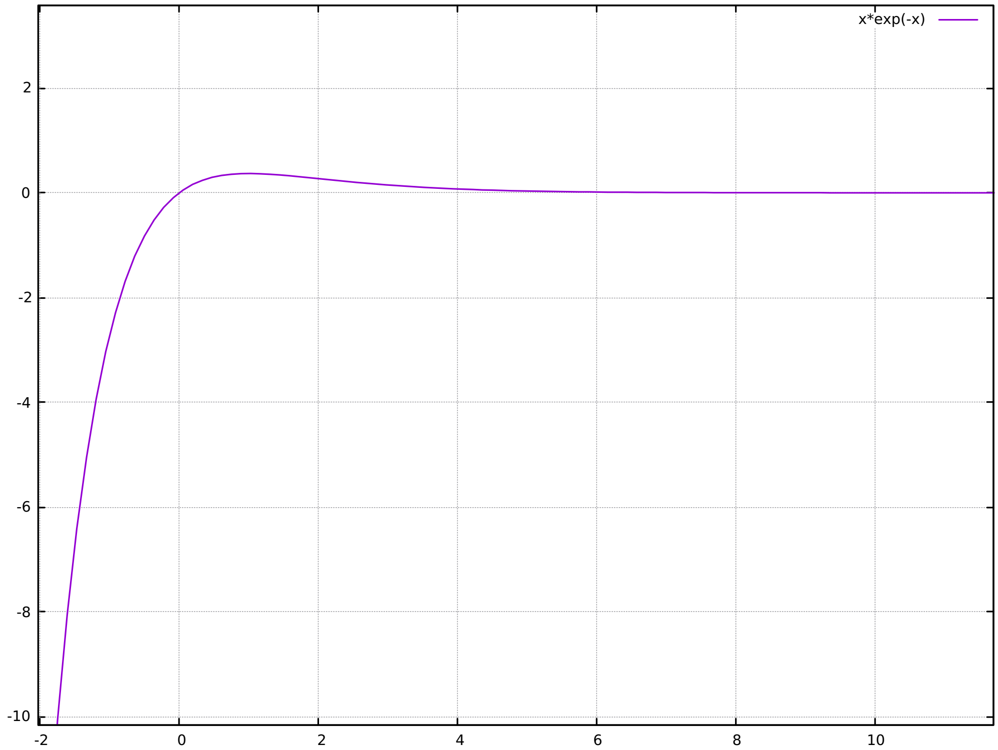
6.2 Opis rozwiązania

Główną trudnością zadania było dobranie odpowiednich parametrów tj. przedziału dla metody bisekcji oraz przybliżeń początkowych dla metody Newtona oraz siecznych. W przypadku funkcji f_1 z wykresu można wywnioskować, że miejsce zerowe jest zawiera się w przedziale $[0, 5]$, jako że $f_1(0)f_2(5) < 0$ owy przedział możemy wykorzystać do policzenia miejsc zerwoych metodą bisekcji. W przypadku metody Newtona musimy uważać na spłaszczenie funkcji w przedziale $(5, \inf)$, gdzie wartość pochodnej będzie bliska 0, co w przypadku wybrania liczby z tego zakresu jako przybliżenia początkowego poskutowałoby otrzymaniem błędu z funkcji *mstycznych*. Jeśli chodzi o metodę siecznych, ważne jest wybranie dwóch pierwszych przybliżeń w ten sposób, aby prosta poprowadzona przez te dwa punkty przecinała oś OX. Poniżej przedstawiony został wykres funkcji f_1 wygenerowany w *GNUPlot*.



Rysunek 2: $f_1(x) = e^{1-x} - 1$ przy użyciu $GNUPlot'a$

W funkcji f_2 zachowujemy się analogicznie jak w przypadku f_1. Należy uwazać na późniejsze spłaszczenie funkcji f_2, ponieważ dla punktów z tego przedziału metoda Newtona nie zadziała, podobnie jest w przypadku metody stycznych, sieczna poprowadzona przez punkty, które są pierwszym przybliżeniem musi przecinać oś OX, za przedział do metody bisekcji wybrałem przedział $[-1.0, 2.0]$, ponieważ z wykresu wynika, że na krańcach tego przedziału wartość funkcji ma różny znak.Poniżej przedstawiony został wykres funkcji f_2 wygenerowany w $GNUPlot$.



Rysunek 3: $f_2(x) = xe^{-x}$ przy użyciu $GNUPlot'a$

6.3 Wyniki

Wyniki zostały przedstawione w tabeli poniżej. Zgodnie z notacją r oznacza przyblizenie miejsca zerowego, v to przybliżenie funkcji w punkcie r, it to liczba wykonanych iteracji, a err odpowiadaj numerowi błędu. Obliczenia zostały przeprowadzone dla $\delta = 10^{-4}$ oraz $\epsilon = 10^{-4}$. Pierwsza tabela przedstawia wyniki dla f_1, natomiast druga dla f_2

$Wywolanie$	r	v	it	err
$mbisekcji(f_1,\ 0.0,\ 2.0,\ 10^{-5},\ 10^{-5})$	1.0000038146972656	$-3.814689989667386e-6$	18	0
$mstycznych(f_1,\ df_1,\ -1.0,\ 10^{-5},\ 10^{-5}, 50)$	0.9999999999700886	$2.991140668484604e-11$	6	0
$msiecznych(f_1,\ 0.0,\ -1.0,\ 10^{-5},\ 10^{-5}, 50)$	0.9999990043764041	$9.956240916153547e-7$	6	0

$Wywolanie$	r	v	it	err
$mbisekcji(f_2,\ 0.0,\ 2.0,\ 10^{-5},\ 10^{-5})$	$-3.814697265625e-6$	$-3.814711817567984e-6$	18	0
$mstycznych(f_2,\ df_2, -2.0,\ 10^{-5},\ 10^{-5}, 50)$	$-1.425500682806244e-9$	$-1.425500684838296e-9$	7	0
$msiecznych(f_2,\ -2.0,\ -1.0,\ 10^{-5},\ 10^{-5}, 50)$	$-4.522991265486159e-9$	$-4.52299128594361e-9$	8	0

Dla funkcji f_2 metoda Newtowna dla $x_0 = 1.0$ zwraca bład, sygnalizujący, że pochodna w tym punkcie jest bliska zeru.

6.4 Wnioski

We wszystkim metodach zaimplementowanych w zadaniach 1-3 ważny jest dobór pierwszy przybliżeń lub również przedziału w którym szukamy miejsca zerowego. W przypadku wyboru wybrania złego przedziału zaimplementowane metody nie zadziałają. Przed próbą znalezenia dokładnej wartości miejsca zerowego warto zapoznać się ze szkicem wykresu funkcji.