Eliza Zawisza, 244967

SPRAWOZDANIE 3

ZADANIE 1

Opis problemu:

W zadaniu należało zaimplementować funkcję rozwiązującą równanie $f(x) = 0$, metodą bisekcji, czyli połowienia przedziałów.

Rozwiązanie:

Dane wejściowe:

- f – funkcja $f(x)$zadana jako anonimowa funkcja,
- a, b – końce przedziału początkowego
- delta, epsilon – dokładności obliczeń

Dane wyjściowe: (r, v, it, err):

- 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 na przedziale [a,b]

Poniżej kod algorytmu, który można znaleźć w pliku *zadanie1.jl*

```
function mbisekcji(f, a::Float64, b::Float64, delta::Float64, epsilon::Float64)
  r = Float64(0.0)
  v = Float64(0.0)
  it = 0
  err = 0

  f_a = f(a)
  f_b = f(b)

  if signbit(f_a) == signbit(f_b)
    err = 1
    return (r, v, it, err)
  end

  w = b - a

  while w > epsilon
    it += 1

    w /= 2.0
    r = a + w
    v = f(r)
```

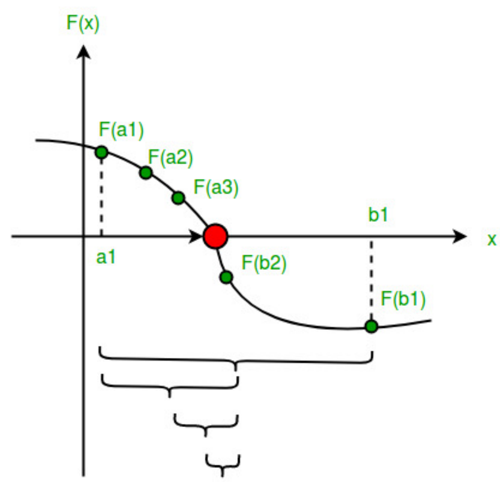
```
    if abs(w) < delta || abs(v) < epsilon
      return (r, v, it, err)
    end

    if signbit(v) != signbit(f_a)
      b = r
      f_b = v
    else
      a = r
      f_a = v
    end
  end
end
```

Podsumowanie algorytmu:

Aby móc zastosować powyższy algorytm, muszą zostać spełnione poniższe założenia:

1. Funkcja $f(x)$ musi być ciągła na przedziale domkniętym [a, b].
2. Funkcja musi przyjmować różne znaki na krańcach przedziału: $f(a)f(b) < 0$[1]

Metoda bisekcji polega na odpowiednim połowieniu przedziału początkowego i wyborze przedziału, w którym znajduje się miejsce zerowe. Na mocy własności Darboux, jeżeli funkcja jest ciągła na przedziale [a,b] i wartości funkcji na końcach przedziałów są różnych znaków, to funkcja ta posiada miejsce zerowe, które znajduje się w przedziale (a, b).



Źródło: https://www.geeksforgeeks.org

Powyższy algorytm wylicza nam miejsce zerowe funkcji metodą bisekcji z dokładnością do zadanego epsilona i delty. Podczas analizy algorytmu warto zwrócić uwagę na kilka rzeczy:

1. W celu obliczenia środka przedziału zastosowany został wzór: r=a + (b-a)/2 zamiast r=(a+b)/2. Z numerycznego punktu widzenia wzór, który został wykorzystany jest lepszy, gdyż podczas dodawania do jednej wartości połowy różnicy dwóch wartości poprawki nie jesteśmy w stanie wyjść poza zakres przedziału, jak może się stać, gdybyśmy wykorzystali wzór drugi.
2. W celu sprawdzenia czy zmienia się znak funkcji na krańcach przedziału również została zastosowana funkcja *signbit, która* sprawdza znaki wartości końców przedziałów, dzięki czemu nie popełniamy ewentualnych błędów podczas wymnażania dwóch liczb.

[1] https://en.wikipedia.org/wiki/Bisection_method

Algorytm sprawdza, za pomocą funkcji opisanej w powyżej w punkcie 2, czy wartości zadanej funkcji na krańcach przedziałów są różnych znaków. Jeżeli są wykonuje się dalszy etap algorytmu, w przypadku, kiedy mamy do czynienia z tymi samymi znakami zwracane są użytkownikowi odpowiednie dane wyjściowe z kodem błędu 1. W momencie, kiedy na krańcach przedziału mamy wartości różnych znaków wykonujemy szereg obliczeń, mający na celu z zadaną dokładnością obliczyć nam miejsce zerowe funkcji. Obliczamy środek przedziału, następnie sprawdzamy, czy funkcja zmienia znak na odpowiednich(wyliczonych jako środek poprzedniego przedziału) krańcach przedziału (a, środek przedziału) lub (środek przedziału, b) i prawidłowy przedział, w którym znajduje się miejsce zerowe funkcji bierzemy jako wejściowy do kolejnej operacji połowienia. Kroki te wykonywane są do momentu, kiedy błąd przybliżenia jest wystarczająco mały lub wartość funkcji f w punkcie środkowym jest wystarczająco bliska 0.

ZADANIE 2

Opis problemu:

Celem zadania było zaimplementować funkcję rozwiązującą równanie $f(x) = 0$ metodą Newtona, czyli z wykorzystaniem stycznych.

Rozwiązanie:

Dane wejściowe:

f , pf – funkcja $f(x)$zadana jako anonimowa funkcja oraz pochodna $f'(x)$,
x_0 - przybliżenie początkowe
delta, epsilon – dokładności obliczeń
maxit – maksymalna dopuszczalna liczba iteracji

Dane wyjściowe: (r, v, it, err):

r – przybliżenie pierwiastka równania $f(x) = 0$
v – wartość $f(r)$
it – liczba wykonanych iteracji
err – sygnalizacja błędu:

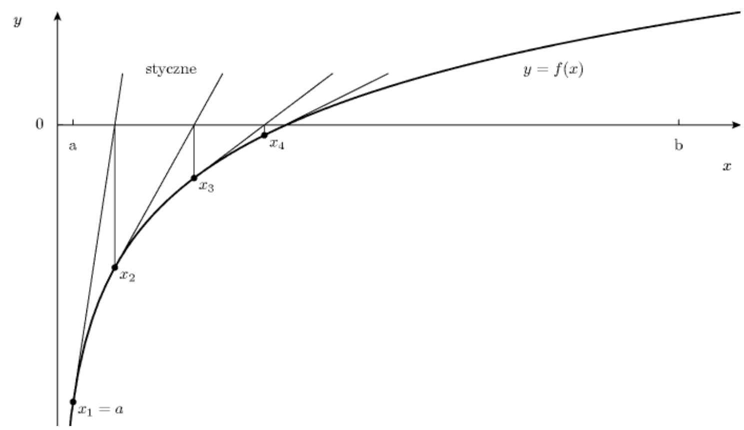
0 – metod zbieżna,
1 – nie osiągnięto wymaganej dokładności w maxit iteracji
2 – pochodna bliska 0

Poniżej kod algorytmu, który można znaleźć w pliku *zadanie2.jl*

```
function mstycznych(f,pf,x0::Float64, delta::Float64,
epsilon::Float64, maxit::Int)
  r = Float64(0.0)
  v = Float64(0.0)
  it = 0
  err = 0
  r = x0
  v = f(r)

  if abs(pf(r)) < epsilon
    err = 2
    return (r, v, it, err)
```

```
    end

    if abs(v) < epsilon
        err = 0
        return (r, v, it, err)
    end

    for it in 1:maxit
        x1 = r - v / pf(r)
        v = f(x1)
        if abs(x1 - r) < delta || abs(v) < epsilon
            r = x1
            return (r, v, it, err)
        end
        r = x1
    end

    err=1
    return (r, v, it, err)
end
```

Podsumowanie algorytmu:



Źródło: https://pl.wikipedia.org/

Aby używać metody Newtona do wyznaczanie miejsc zerowych muszą być spełnione następujące warunki:

1. W przedziale [a, b] musi się znajdować dokładnie 1 miejsce zerowe
2. Funkcja ma różne znaki na krańcach przedziału
3. W przedziale [a,b] pierwsza pochodna $f'(x)$ jest różna od 0 [2]

Algorytm Newtona polega na wyznaczaniu kolejnych aproksymacji miejsca zerowego funkcji jako miejsc zerowych stycznych do wykresu funkcji w poprzednim punkcie. Analizując powyższy algorytm można zauważyć, że na początku zostaje sprawdzony warunek kończące działanie algorytmu, gdy wartość funkcji w zadanym x0 jest dostatecznie bliska 0 zwracany jest kod błędu 2 i program kończy swoje działanie

W celu obliczania kolejnych przybliżeń używa się wzoru rekurencyjnego $x_{k+1} = x_k - \frac{f(x)}{f'(x)}$:. Początkowo wybierany jest punkt startowy, dla którego prowadzona jest styczna do wykresu funkcji. Następnie obliczana jest wartość funkcji, jeżeli nie jest on wystarczająco blisko 0 to rekurencyjnie wg wcześniej podanego wzoru wyznaczamy kolejny punkt i obliczamy wartość funkcji w nowo wyznaczonym punkcie, następnie prowadzimy styczną do wykresu funkcji w tym punkcie, a punkt przecięcia z osią OX jest

[2] https://eduinf.waw.pl/inf/alg/005_root/m0013.php

kolejnym przybliżeniem x_{k+1}. Czynność wyznaczania miejsca zerowego z wykorzystaniem stycznej powtarzamy do momentu, kiedy odległość pomiędzy $x_{k+1}, a\ x_k$ jest mniejsza niż zadana dokładność obliczeń lub przybliżenie miejsca zerowego jest wystarczająco bliskie 0.

ZADANIE 3

Opis problemu:

Celem zadania było zaimplementować funkcję rozwiązującą równanie $f(x) = 0$ metodą siecznych.

Rozwiązanie:

Dane wejściowe:

f – funkcja $f(x)$zadana jako anonimowa funkcja ,

x_0, x_1 - przybliżenia początkowe

delta, epsilon – dokładności obliczeń

maxit – maksymalna dopuszczalna liczba iteracji

Dane wyjściowe: (r, v, it, err):

r – przybliżenie pierwiastka równania $f(x) = 0$

v – wartość $f(r)$

it – liczba wykonanych iteracji

err – sygnalizacja błędu:

0 – metod zbieżna,

1 – nie osiągnięto wymaganej dokładności w maxit iteracji

Poniżej kod algorytmu, który można znaleźć w pliku *zadanie3.jl*

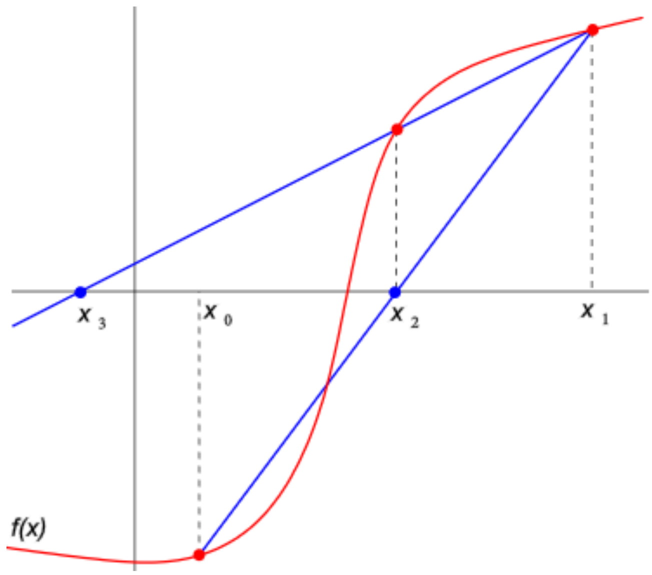
```
function msiecznych(f, x0::Float64, x1::Float64, delta::Float64, epsilon::Float64, maxit::Int)
  f_x0 = f(x0)
  f_x1 = f(x1)

  for it in 1:maxit
    if abs(f_x0) > abs(f_x1)
      x0, x1 = x1, x0
      f_x0, fx1 = f_x1, fx0
    end

    s = (x1 - x0) / (f_x1 - f_x0)
    x1 = x0
    f_x1 = f_x0
    x0 -= f_x0 * s
    f_x0 = f(x0)

    if abs(x1 - x0) < delta || abs(f_x0) < epsilon
      r = x0
      v = f_x0
      err = 0
      return (r, v, it, err)
    end
  end
  r = x0
```

```
    v = f_x0
    err = 1
    return (r, v, it, err)
end
```

Podsumowanie algorytmu:

Aby używać metody siecznych do wyznaczanie miejsc zerowych muszą być spełnione następujące warunki:

1. Funkcja musi być ciągła na przedziale [a,b].
2. Funkcja ma różne znaki na krańcach przedziału
3. W przedziale [a,b] pierwsza pochodna $f'(x)$ jest różna od 0 [3]



Źródło: https://www.bragitoff.com

Metoda siecznych zwana również metodą interpolacji liniowej lub metodą Eulera służy do wyliczania miejsc zerowych funkcji, opiera się na założeniu, że na odpowiednio niewielkim przedziale [a,b] funkcja może zostać zastąpiona sieczną przechodzącą przez dwa punkty A(a, f(a)), B(b, f(b)).

Metoda siecznych wywodzi się bezpośrednio z metody Newtona. Obliczanie wartości pochodnej obecnej w metodzie Newtona zastąpiono ilorazem różnicowym. W celu obliczenia przybliżenia miejsca zerowego funkcji wykorzystujemy zmodyfikowany wzór rekurencyjny z poprzedniego zadania. $x_{k+1} = x_k - f(x_k)\frac{x_k - x_{k-1}}{f(x_k) - f(x_{k-1})}$. W przeciwieństwie do metody Newtona w tym przypadku potrzebujemy dwóch argumentów początkowych, natomiast w każdej kolejnej iteracji obliczamy już tylko jedną wartość. W powyższym algorytmie w celu utrzymania nierówności $|f(a)| \leq |f(b)|$ następuje zamiana końców przedziałów, dzięki tej zależności mamy zapewnienie, że moduły funkcji f, w kolejno wyliczanych przez nas punktach nie rosną. Algorytm kończy swoje działanie, jeżeli przybliżenie pierwiastka jest mniejsze od zadanej dokładności, bądź odległość kolejnych przybliżeń jest dostatecznie mała.

ZADANIE 4

Opis problemu:

Celem zadania było zastosowanie wcześniej zaimplementowanych metod w celu obliczenia pierwiastka równania $\sin x - \left(\frac{1}{2}x\right)^2 = 0$.

1. Bisekcji z przedziałem początkowym [1.5,2], $\delta = \frac{1}{2}10^{-5}, \epsilon = \frac{1}{2}10^{-5}$,
2. Newtona z przybliżeniem początkowym $x_0 = 1.5, \delta = \frac{1}{2}10^{-5}, \epsilon = \frac{1}{2}10^{-5}$

---

[3] https://eduinf.waw.pl/inf/alg/005_root/m0012.php

3. Siecznych z przybliżeniami początkowymi $x_0 = 1,\ x_1 = 2, \delta = \frac{1}{2}10^{-5}, \epsilon = \frac{1}{2}10^{-5}$

Rozwiązanie:

W celu rozwiązania zadania użyłam tak jak zostało opisane algorytmów z zadań 1-3, które znajdują się w module MiejscaZerowe w pliku *MiejscaZerowe.jl.* Maskymalna ilośc iteracji w tym przypadku została ustawiona na 20. Kompletny kod programu znajduje się w pliku zadanie4.jl.

```
println("Bisection method:")
a = 1.5
b = 2.0
(r,v,it,err) = MiejscaZerowe.mbisekcji(f, a, b, delta, epsilon)
println("r: $(r), v: $(v), it: $(it), err: $(err)")

println("Newton's method:")
x0 = 1.5
(r,v,it,err) = MiejscaZerowe.mstycznych(f, pf, x0, delta, epsilon, maxit)
println("r: $(r), v: $(v), it: $(it), err: $(err)")

println("Secant method:")
x0 = 1.0
x1 = 2.0
(r,v,it,err) = MiejscaZerowe.msiecznych(f, x0, x1, delta, epsilon, maxit)
println("r: $(r), v: $(v), it: $(it), err: $(err)")
```

Wyniki oraz ich interpretacja:

Metoda	x_0	$f(x_0)$	Ilość iteracji	Kod błędu
Bisekcji	1.9337539672851562	-2.7027680138402843e-7	16	0
Stycznych/Newtona	1.933753779789742	-2.2423316314856834e-8	4	0
Siecznych	1.933753644474301	1.564525129449379e-7	4	0

Na podstawie analizy powyższej tabeli łatwo można zauważyć, że najwięcej iteracji w celu obliczenia wyniku potrzebowała metoda bisekcji (16), podczas gdy metoda siecznych i stycznych(Newtona) potrzebowała zaledwie 4 iteracji. Najbardziej dokładny wynik ($x \approx 1.93375376282702125334$[4]) podała metoda Newtona, natomiast najmniej dokładny uzyskujemy wykorzystując metodę bisekcji.

Wnioski:

Powyższe wyniki dokładnie ukazują różnicę w działaniu poszczególnych metod obliczania miejsca zerowego funkcji. Ilość iteracji przeprowadzonych w celu uzyskania wyniku obrazuje teoretyczną zbieżność poszczególnych metod. Metoda bisekcji posiada zbieżność liniową, metoda stycznych(Newtona) ma kwadratowy współczynnik zbieżności, natomiast metoda siecznych nadliniowo. Metoda bisekcji pomimo najwolniejszego współczynnika zbieżności, jest metodą najstabilniejszą, w powyższym przykładzie wyliczając miejsce zerowe tą metodą uzyskaliśmy wynik najbardziej zbliżony do zadanej na wejściu dokładności obliczeń.

[4] wynik obliczony za pomocą programu wolframAlpha

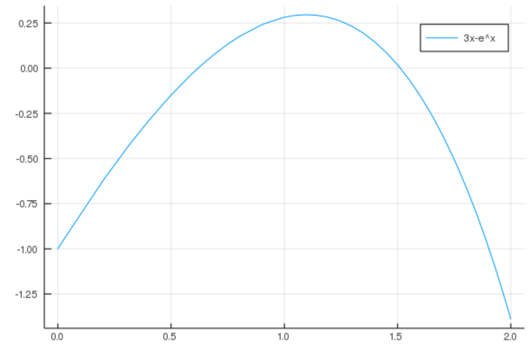
ZADANIE 5

Opis problemu:

Celem zadania było znalezienie za pomocą metody bisekcji zmiennej x, dla której przecinają się wykresy funkcji $y = 3x$ i $y = e^x$. Wymagana dokładność obliczeń: $\delta = 10^{-4}, \epsilon = 10^{-4}$.

Rozwiązanie:

W celu obliczenia miejsc przecięcia wykresów należy obliczyć równanie: $3x - e^x = 0$. Aby znaleźć odpowiednie przedziały do poszukiwania miejsc zerowych funkcji $y = 3x - e^x$ skorzystałam z wykresu narysowanego za pomocą pakietu Plots.jl. Analizując wykres obok łatwo można zauważyć, że ma on dwa miejsca zerowe, znajdujące się odpowiednio w przedziałach [0,1] i [1,2].



Wyniki oraz ich interpretacja:

Przedział	Miejsce zerowe	Wartość funkcji w miejscu zerowym	Ilość iteracji
[0,1]	0.619140625	9.066320343276146e-5	9
[1,2]	1.5120849609375	7.618578602741621e-5	13

Powyższa tabela przedstawia wyniki, jakie otrzymałam stosując metodę bisekcji w celu obliczenia miejsc zerowych funkcji. Porównując dane zawarte w tabeli wraz z wykresem funkcji powyżej można zaobserwować, że obliczone wyniki są poprawnymi, co do dokładności obliczeń.

Wnioski:

Stosując metodę bisekcji do obliczania miejsc zerowych funkcji należy bardzo uważać przy wybierze przedziału na jakim próbujemy szukać miejsc zerowych. Najlepsza metodą na wybranie przedziału jest ilustracja graficzna ukazująca przebieg zmienności funkcji, na jej podstawie możemy zobaczyć, jaki przedział wybrać, by wartości na jego krańcach były przeciwnych znaków i funkcja zawierała na nim tylko jedno miejsce zerowe. Niewątpliwie w przypadku, kiedy nie możemy skorzystać z ilustracji graficznej wykresu funkcji lepszymi metodami do znajdowania miejsc zerowych: metoda stycznych(newtona) i metoda siecznych.

ZADANIE 6

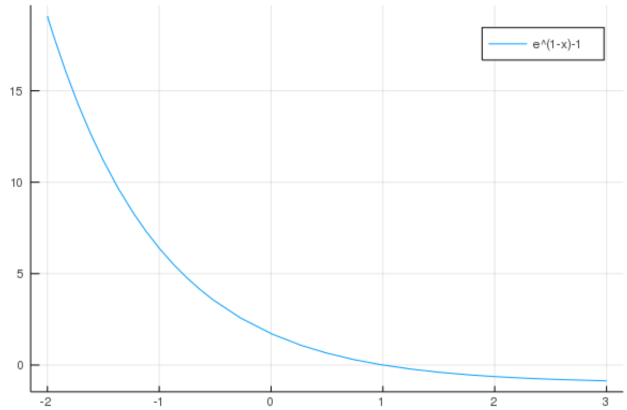
Opis problemu:

Celem zadania było znalezienie miejsc zerowych funkcji $f_1 = e^{1-x} - 1$ oraz $f_2 = xe^{-x}$ za pomocą metod bisekcji, Newtona i siecznych. Wymagana dokładność obliczeń: $\delta = 10^{-5}, \epsilon = 10^{-5}$. Ponadto należało sprawdzić co się stanie, gdy w metodzie Newtona dla $f_1$wybierzemy $x_0 \in (1, \infty]$, a dla $f_2$wybierzemy $x_0 > 1$, czy można wybrać $x_0 = 1$?

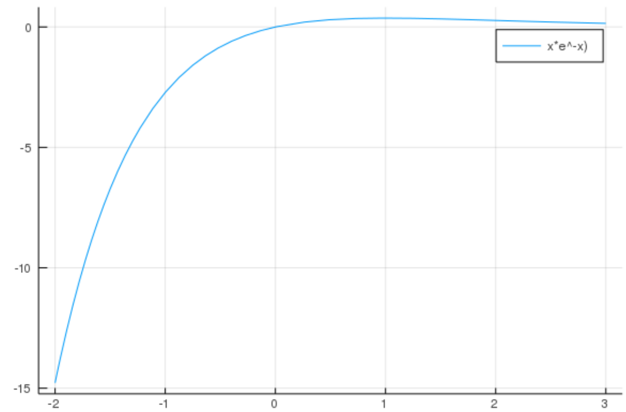
Rozwiązanie:

Aby rozwiązać zadanie należy dobrać odpowiedni przedział dla metody bisekcji, w którym znajduje się miejsce zerowe, natomiast dla metod Newtona i siecznych należy dobrać odpowiednie przybliżenia początkowe. Aby dobrać odpowiednie dane wygenerowałam wykresy funkcji f_1 i f_2 za pomocą pakietu Plots.jl

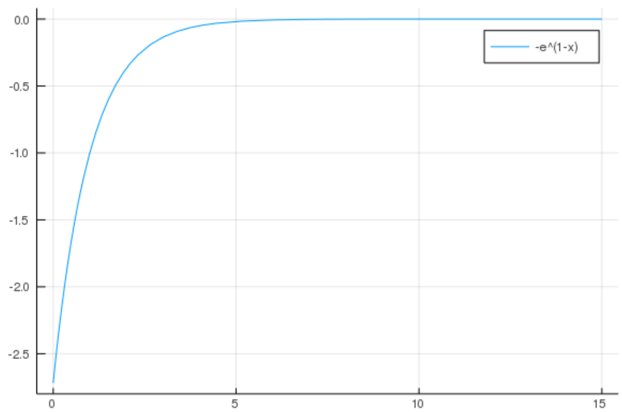
Wykres $f_1 = e^{1-x} - 1$



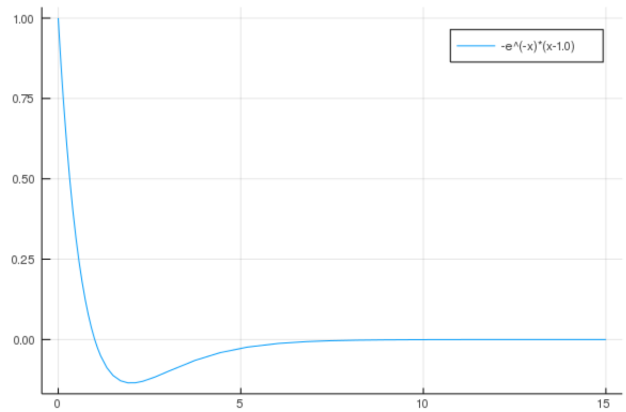
Wykres $f_2 = xe^{-x}$



Wykres pochodnej $f_1{}' = -e^{1-x}$



Wykres pochodnej $f_2' = -e^{-x} * (x - 1)$



Wyniki oraz ich interpretacja:

1. Wyniki $f_1 = e^{1-x} - 1$

- **Metoda bisekcji**

Przedział	r	$f_1(r)$	Ilość iteracji
[0,2]	1.0	0.0	1
[0.5, 2.5]	1.0	0.0	2
[-1.4, 3.2]	1.0000038146972656	-3.814689989667386e-6,	18
[0.8, 5.2]	1.0000061035156251,	-6.103496998699498e-6,	16
[0.99, 1.02]	0.999990234375,	9.765672683847981e-6,	10

- **Metoda Newtona (maxit=100)**

x0	r	$f_1(r)$	Ilość iteracji
-2.5	0.9999999026648796	9.733512507104081e-8,	7
0.0	0.9999984358892101,	1.5641120130194253e-6,	4
0.97	0.9999999007646667,	9.923533816902363e-8,	2

1.9	0.999999999431732,	5.682680992435962e-10,	5
4.3	0.9999969433248612,	3.0566798103759396e-6,	27

- **Metoda siecznych (maxit=100)**

x0	x1	r	$f_1(r)$	Ilość iteracji
-2.0	3.0	0.9999993443793663,	6.556208484997939e-7,	15
0.5	2.5	0.9999999931413072,	6.8586929469205415e-9,	6
0.7	1.5	1.000002366860475,	-2.366857673963274e-6,	4
0.91	1.07	1.0000001744987157,	-1.744987004892451e-7,	3

2. Wyniki $f_2 = xe^{-x}$
3. **Metoda bisekcji**

Przedział	r	$f_2(r)$	Ilość iteracji
[-1.0, 2.0]	7.62939453125e-6,	7.62933632381113e-6,	17
[-0.75, 0.5]	7.62939453125e-6,	7.62933632381113e-6,	15
[-1.4, 0.2]	-2.7755575615628914e-17,	-2.7755575615628914e-17,	3
[-0.5, 0.5]	0.0	0.0	1
[-0.01, 0.02]	-9.765625000000578e-6,	-9.765720367897881e-6,	10

4. **Metoda Newtona (maxit=100)**

x0	r	$f_2(r)$	Ilość iteracji
-2.5	-3.3084197593330218e-6,	-3.3084307049924325e-6,	7
0.01	-1.0202010000017587e-8,	-1.0202010104098596e-8,	2
-0.97	-1.7434891787411477e-7,	-1.743489482716626e-7,	5
1.2	14.974320149741843,	4.699833827208103e-6,	8
-4.3	-9.674600373818791e-10,	-9.674600383178582e-10,	10

5. **Metoda siecznych (maxit=100)**

x0	x1	r	$f_2(r)$	Ilość iteracji
-0.5	0.1	7.89618162073052e-7,	7.896175385764562e-7,	4
0.0	0.5	0.0	0.0	1
-0.7	1.5	14.850014182115977,	5.277735885143915e-6,	14
-0.91	1.07	0.7934483098783266,	0.3588627796511534,	3

Metoda Newtona f_1 $x_0 \in (1, \infty]$:

x0	r	$f_2(r)$	Ilość iteracji	Kod błędu
2.5	0.9999934982589662,	6.501762170207925e-6,	6	0
10.0	NaN	NaN	0	1
100.0	100.0,	-1.0,	0	2
1000.0	1000.0,	-1.0,	0	2

Jak widać analizując powyższą tabelę dla niewielkiej wartości x0 większej od 1 możemy obliczyć jeszcze miejsce zerowe funkcji, niestety wraz z oddalaniem się od 1 pochodna funkcji zaczyna dążyć do 0, przez co wyniki, które otrzymujemy nie są wiarygodne. W przypadku x0 = 10.0 nie udało się wyliczyć miejsca zerowego w maksymalnej ilości iteracji. (w moim przypadku było ich 100).

Metoda Newtona $f_2\ x_0 > 1$:

x0	r	$f_2(r)$	Ilość iteracji	Kod błędu
1.0	1.0,	0.36787944117144233,	0	2
10.0	14.380524159896261,	8.173205649825554e-6,	4	0
100.0	100.0,	3.7200759760208363e-42,	0	2
1000.0	1000.0,	0.0,	0	2

Jak widać analizując powyższą tabelę dla x0 = 1 pochodna funkcji jest bliska 0(przyglądając się wykresowi pochodnej można zauważyć, ze pochodna w punkcie x0=1 ma wartość 0) i nie można otrzymać wiarygodnego wyniku, podobne błędy otrzymuję w momencie, gdy x0 staje się coraz większe.

Wnioski:

Niewątpliwie pierwszy wniosek jaki nasuwa się po przeanalizowaniu powyższych wyników jest taki, że należy zwracać szczególną uwagę, przy doborze odpowiednich przedziałów do szukania miejsc zerowych funkcji. Jak widać na przykładzie wszystkich metod im wybierzemy lepsze przedziały dla metody bisekcji lub miejsca początkowe dla metod stycznych i siecznych otrzymujemy dokładniejsze wyniki w mniejszych ilościach iteracji (1, 2, 3) w metodzie bisekcji, (2, 4, 5) w metodzie Newtona i (1 , 3, 4) w metodzie siecznych. W celu jak najlepszego dobrania miejsc początkowych należy znać dokładnie przebieg zmienności funkcji np. poprzez zilustrowanie graficzne. W metodzie bisekcji podczas wyboru przedziału najlepiej jak najbardziej zwęzić przedział, na którym chcemy operować, natomiast podczas wyboru miejsc początkowych do metody Newtona i metody siecznych należy dobrać odpowiednio mało oddalone punkty od miejsca zerowego.