LambertSkriptitLIVE2.mlx

Heikki Apiola 15.11.2021

Kirjoituksen liite Solmun numerossa 2/2021 julkaistuun Lambert-juttuun. Liite on **parannettu versio** siitä, joka oli paikalla ennen tätä päivää.

Tässä ovat täydellisinä kaikki kirjoituksen koodit perusteellisine selityksineen.

Kyseessä on Matlab:n "Live editorilla" tehty dokumentti. Kaikki koodit voidaan ajaa Octavella, mutta siinä ei ole käytettävissä näin kehittynyttä editoria. Itse asiassa myös Matlab:ssa on suositeltavaa tehdä varsinainen ohjelmankehitys vaatimattomammalla, "Octave-tyylisellä" editorilla, mutta kokoaminen dokumentiksi on varsin palkitseva harjoitus, jos Matlab-ohjelma on käytettävissä.

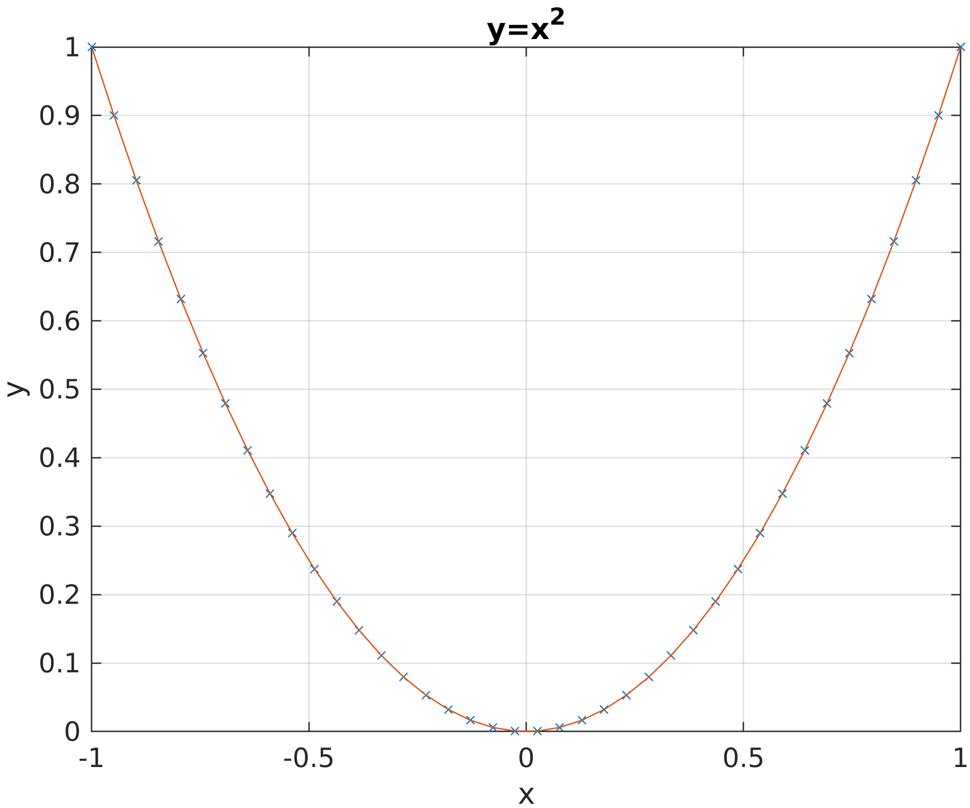
Tämä tiedosto on .mlx-tyyppisen Matlab-dokumentin pdf-tuloste. Voit kopioida koodeja Octave-editoriin ja tehdä siellä omia muunnelmiasi.

Käänteisfunktiot ja niiden haarat

Aloitetaan tuttuakin tutummasta:

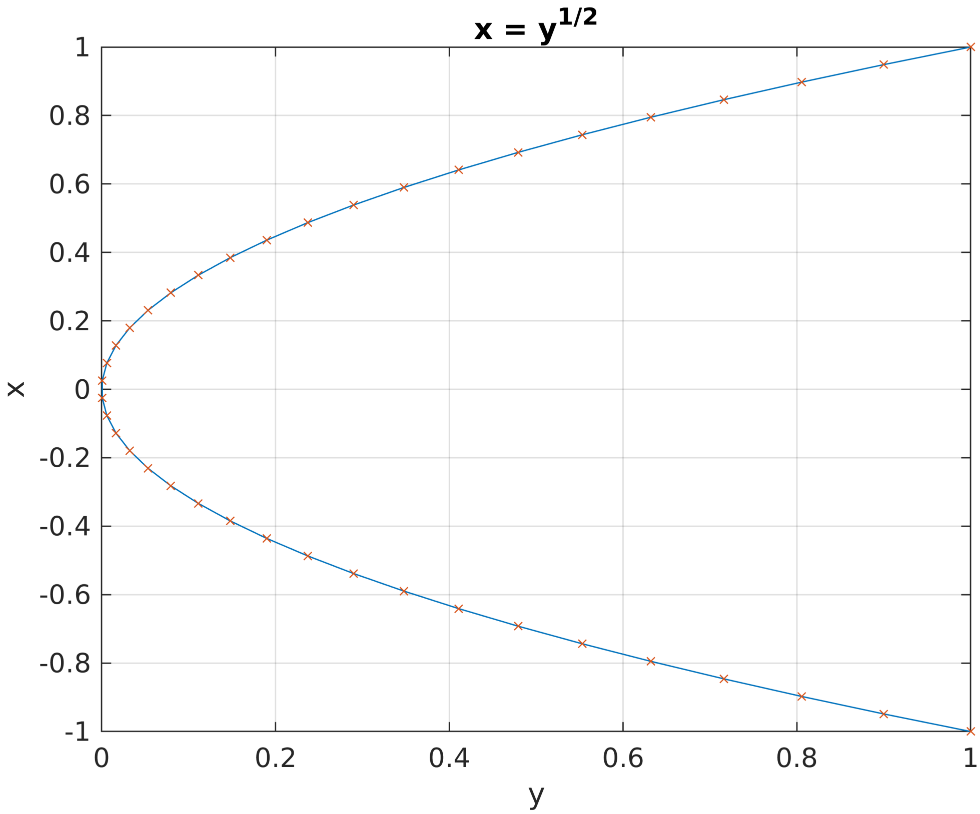
$f(x) = x^2$

```
close all             % Suljetaan grafiikkaikkunat.
x=linspace(-1,1,40); % Välin [-1,1] jako 40:een osaan.
y=x.^2;               % Huomaa "pisteittäinen" potenssi
plot(x,y,'x','MarkerSize',4); % Piirretään pisteet (x(k),y(k)) ilman yhdistysviivoja.
hold on
plot(x,y)                     % Piirretään myös yhdistysviivat.
grid on
title('y=x^2')
xlabel('x');ylabel('y')
```



Käänteisfunktion kuvaamiseksi vaihdetaan x:n ja y:n roolit:

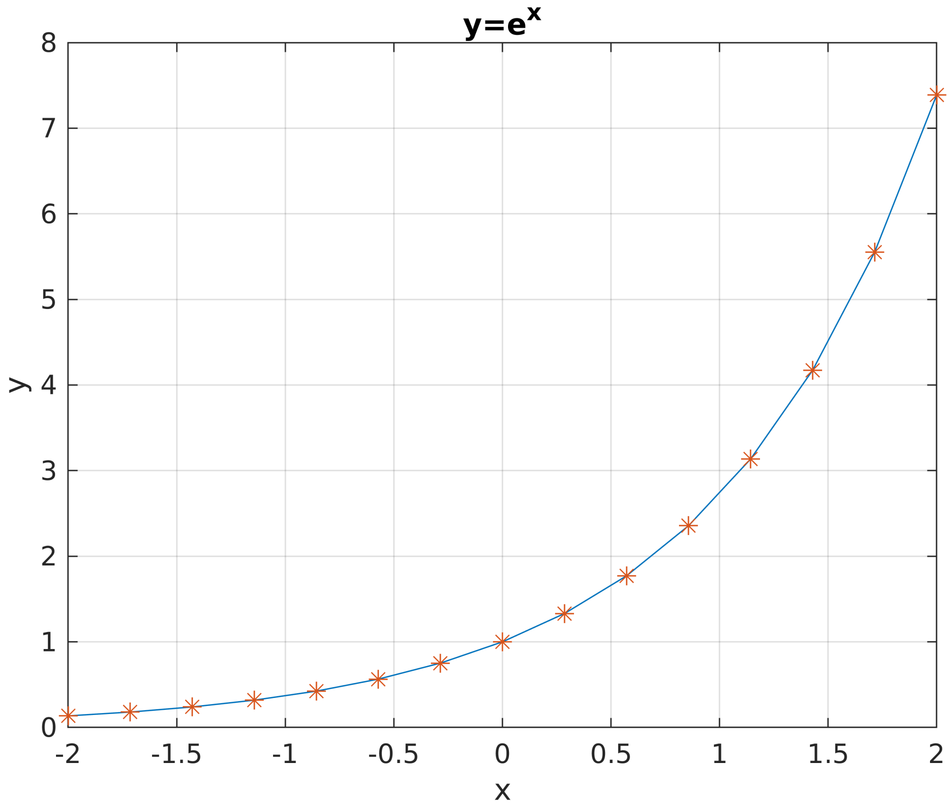
```
figure       % Uusi grafiikkaikkuna
plot(y,x);   % Pisteet (y(k),x(k)) yhdistysjanoineen
hold on
plot(y,x,'x','MarkerSize',4); % Pelkät pisteet 'x':llä
grid on
title('x = y^{1/2}')
xlabel('y');ylabel('x')
```



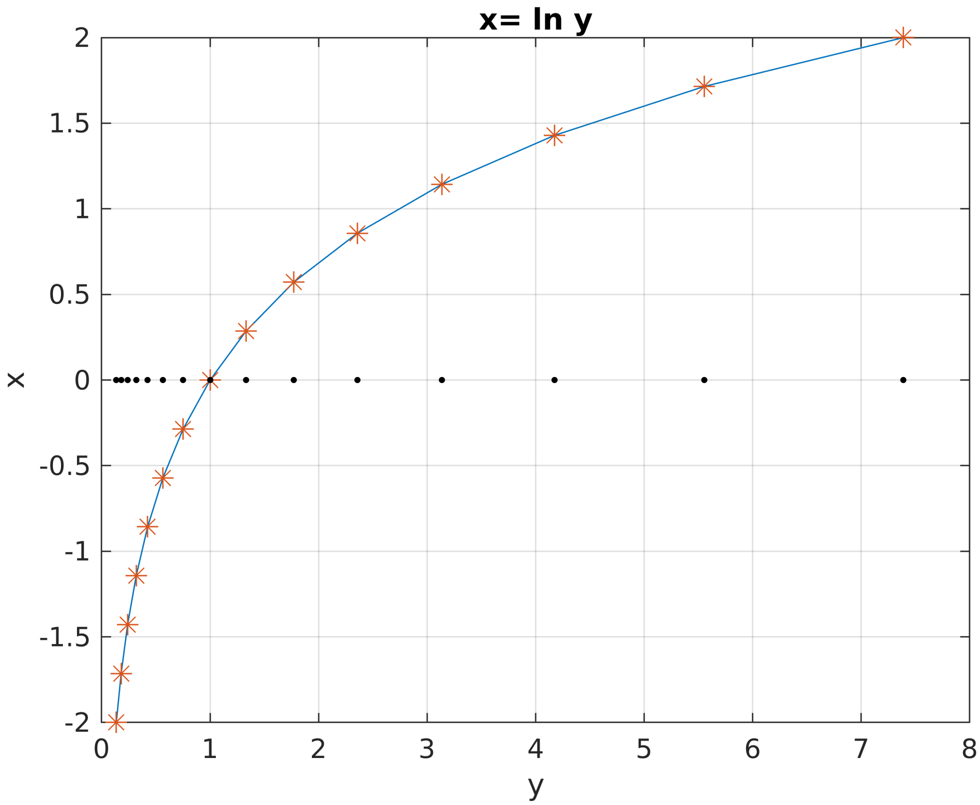
Pisteet (y_k, x_k) muodostavat f:n käänteisfunktion 2 haaraa, neliöjuurenpositiivisen ja negatiivisen haaran: $+\sqrt{y}$ ja $-\sqrt{y}$.

Eksponentti- ja logaritmifunktiot

```
close all
x=linspace(-2,2,15);
y=exp(x);
plot(x,y);grid on; hold on
plot(x,y,'*','MarkerSize',7);
title('y=e^x')
xlabel('x');ylabel('y')
```




```
% Vaihdetaan taas x ja y
figure             % Uusi grafiikkaruutu
plot(y,x);grid on;hold on
plot(y,x,'*','MarkerSize',8);
title('x= ln y')
xlabel('y');ylabel('x')
hold on
plot(y,zeros(size(y)),'.k','MarkerSize',8)
```



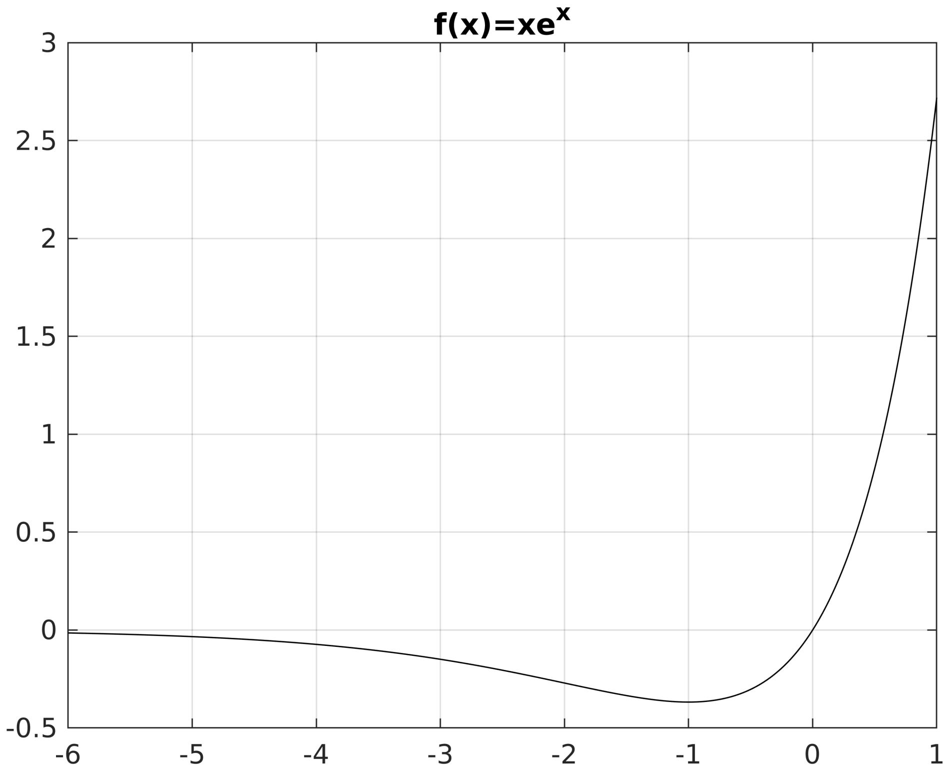
Tässä tapauksessa käänteisfunktiota sanotaan luonnolliseksi logaritmiksi.

Viimeisellä koodirivillä piirrettiin myös vaaka-akselin pisteet, joissa käänteisfunktion arvot on laskettu. ('.k' , jossa 'k' viittaa väriin 'blac**k**')

Lambertin funktio

Lähdetään nyt e^x:n sijasta liikkeelle funktiosta $f(x) = x e^x$.

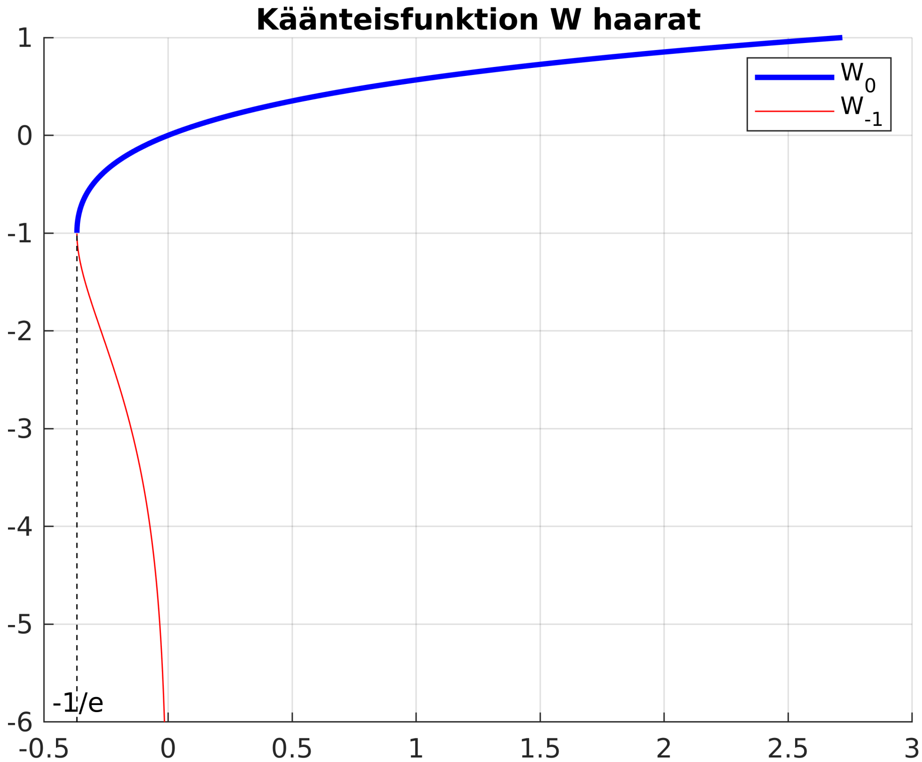
```
close all
f = @(x) x.*exp(x);      % Funktiomääritys ("at x" f saa arvon x.*exp(x))
x=linspace(-6,1,1000);  % 1000:n pituinen x-vektori poistaa mahd. "kulmikkuuden".
y=f(x);
plot(x,y,'k');grid on  % Piirretään värillä 'k' eli 'black'
title('f(x)=xe^x')
```



Funktion $f(x) = xe^x$ käänteisfunktio

Edetään logaritmimallin mukaisesti.

```
figure
clf
% Vaihdetaan x ja y, eli
% piirretään pisteet (y(k),x(k)):
hold on
x1=x(x>=-1);y1=f(x1);  %1. haara -> W_0
x2=x(x<=-1);y2=f(x2);  %2. haara -> W_{-1}
p1=plot(y1,x1,'b','LineWidth',2); % 'blue'
p2=plot(y2,x2,'r');               % 'red'
title('Käänteisfunktion W haarat')
legend([p1,p2],{'W_0','W_{-1}'})
e=exp(1); % e ei ole varattu symboli.
plot([-1/e -1/e],[-6,-1],'--k')
text(-1/e-0.1,-5.8,'-1/e')
grid on
legend([p1,p2],{'W_0','W_{-1}'})
```



Käytetään valmista lambertw-funktiota

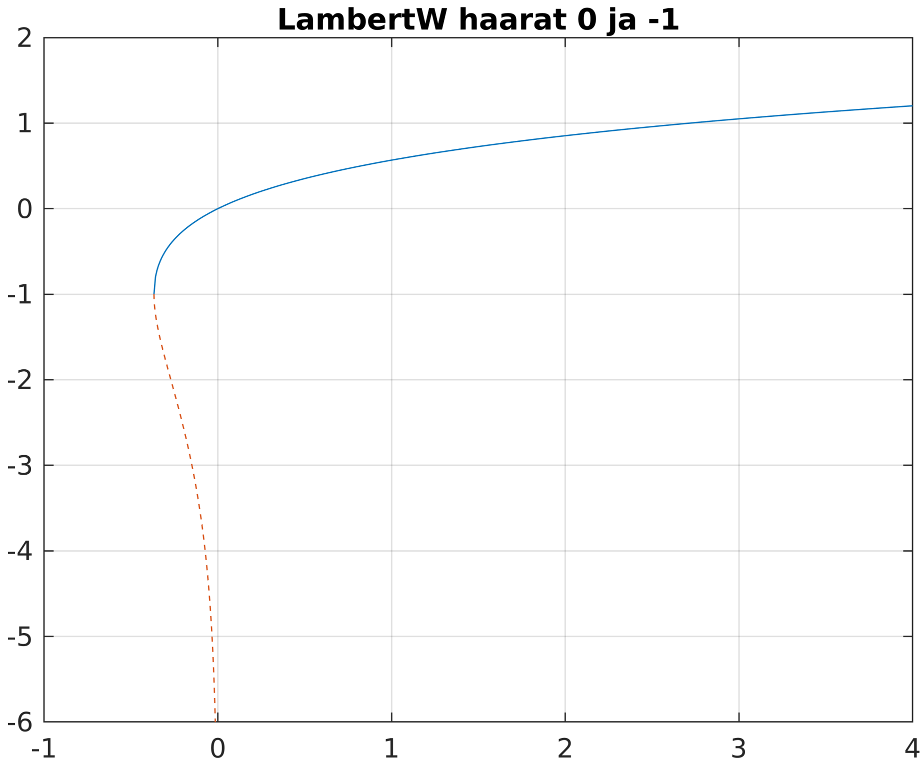
```
close all
e=exp(1);
x0=linspace(-1/e,4,500);
x1=linspace(-1/e,-0.01,500);
y0=lambertw(x0);   % sama kuin lambertw(0,x)
y1=lambertw(-1,x1); % Laskeva haara
max(abs(imag(y1)))
```

```
ans = 5.7397e-42
```

```
% ans = 5.7397e-42
% plot varoittaa pienen pienistäkin imag. komponenteista
plot(x0,y0,x1,y1,'--')
```

```
Warning: Imaginary parts of complex X and/or Y arguments ignored.
```

```
ylim([-6,2]);
grid on
title('LambertW haarat 0 ja -1')
```



Esimerkkejä, tehtäviä

Solmu s. 8

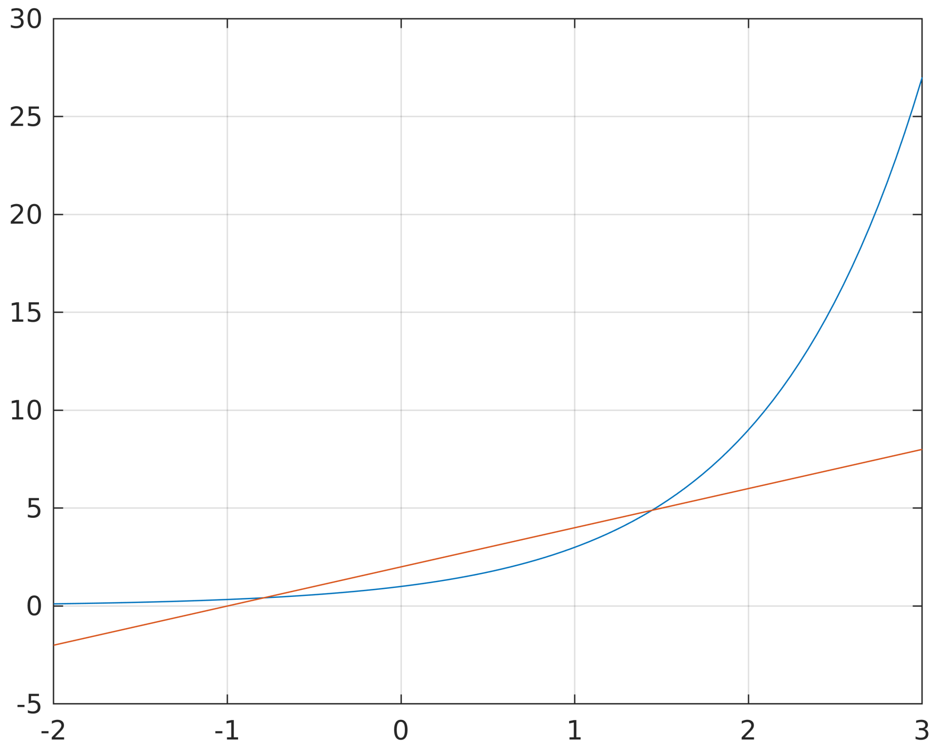
$$3^x = 2x + 2$$

Ratkaisut:

$$x_k = -1 - \frac{1}{\ln 3} W_{-k}(\frac{-\ln 3}{6}), k = 0, 1$$

Piirretään ensin yhtälön vasen ja oikea puoli:

```
clear;close all;format compact
x=linspace(-2,3);
u=3.^x;v=2*x +2;
plot(x,u,x,v)
grid on
```




```
% Katsotaan, mille välille W:n argumentti sijoittuu.
-1/exp(1)    % -0.3679
```

```
ans = -0.3679
```

```
-log(3)/6    % -0.1831  Siis välille (-1/e,0), joten molemmat haarat määriteltyjä.
```

```
ans = -0.1831
```

$-\frac{1}{e} < -\frac{\ln 3}{6} < 0$, joten $-\frac{\ln 3}{6}$ on kummankin W-haaran määrittelyjoukossa,

ja voidaan siis laskea:

```
x0=-1-lambertw(0,-log(3)/6)/log(3)  % -0.7901
```

```
x0 = -0.7901
```

```
y0=2*x0+2
```

```
y0 = 0.4198
```

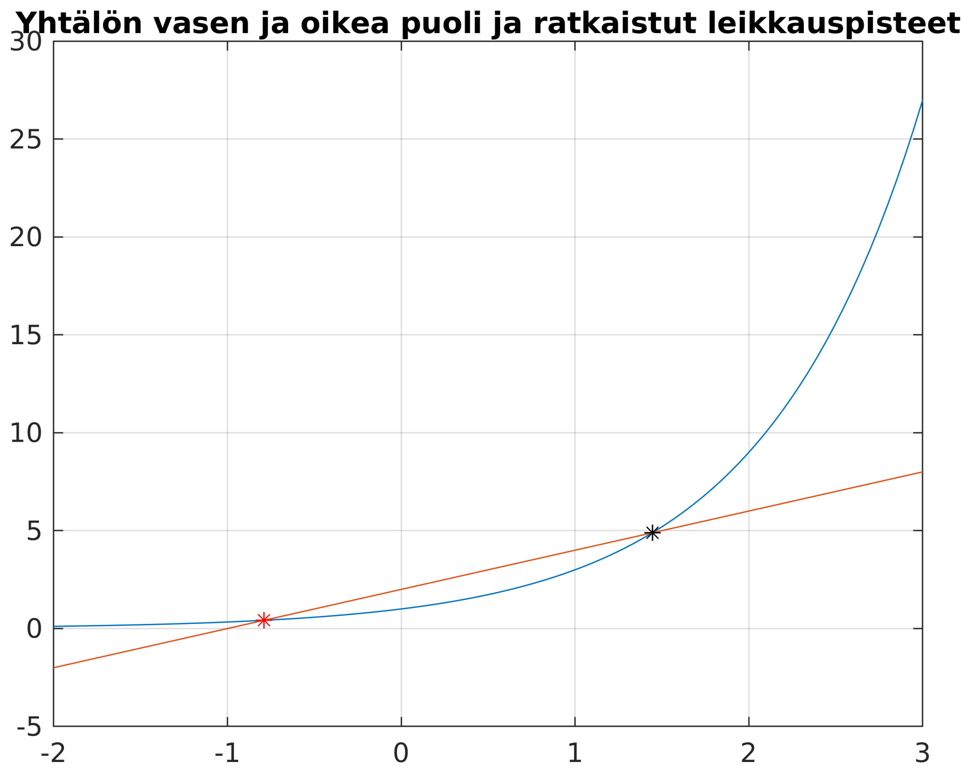
```
x1=-1-lambertw(-1,-log(3)/6)/log(3) % 1.4446
```

```
x1 = 1.4446
```

```
y1=2*x1+2
```

```
y1 = 4.8891
```

```
hold on
plot(x0,y0,'*r')
plot(x1,y1,'*k');shg
grid on
title('Yhtälön vasen ja oikea puoli ja ratkaistut leikkauspisteet')
```



Voit kokeilla myös ratkaisemista Matlabin (Octaven) valmiilla ratkaisijalla **fsolve.** Alkuarvot vaikka kuvasta katsomalla (karkeasti) -1 ja 1. Laitan komennot kommenttien taakse säästääkseni lukijaa, koska tulostuksessa on runsaasti selostusta ratkaisun tarkkuuden arvioinnista, joka ensialkuun vaikuttaa virheilmoittelulta.

```
% F=@(x) 3.^x - (2*x+2) % Määritellään funktio F. Yhtälö on siis F(x) = 0.
% xa=fsolve(F,-1)   %  1.4446    (Lähtöarvo -1)
% xb=fsolve(F,1)    % -0.7901    (Lähtöarvo 1)
% Näyttötarkkuudella samat kuin edellä.
```

Esim 2 s. 8 $x = ae^x$

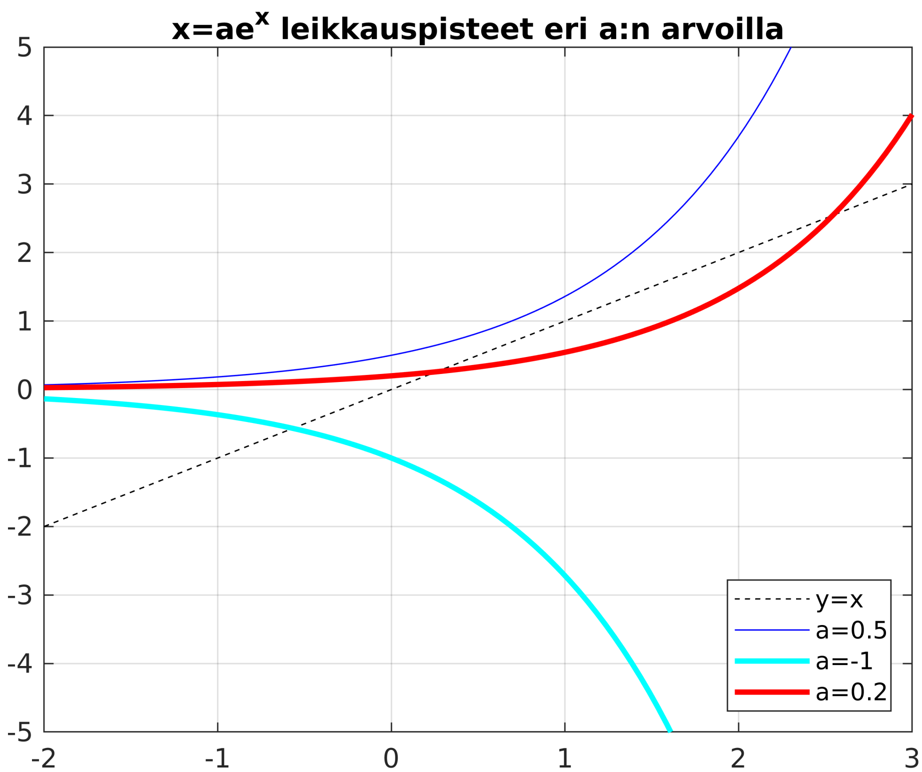
Millä a:n arvoilla on 2,1,0 ratkaisua?

Ratkaisu:

$x = -W(-a)$

```
% 1) a> 1/e = 0.3679 => ei ratk. Esim: a=0.5
% 2) a < 0           => 1 ratk.   Esim. a=-1
% 3) 0 < a < 1/e     => 2 ratk.   Esim. a=0.2
clf
x=linspace(-2,3);
plot(x,x,'--k')   % Suora y=x
hold on

a=0.5; plot(x,a*exp(x),'b') % Ei ratk.
a=-1; plot(x,a*exp(x),'c','LineWidth',2)  %  1 ratk.
a=0.2; plot(x,a*exp(x),'r','LineWidth',2) %  2 ratk.
grid on; shg
ylim([-5 5])
hold off
legend('y=x','a=0.5','a=-1','a=0.2','Location','SouthEast')
title('x=ae^x leikkauspisteet eri a:n arvoilla')
```



Lasketaan ratkaisut:

1)

```
a=0.5
```

```
a = 0.5000
```

```
-lambertw(0,-a)    % 0.7940 - 0.7701i
```

```
ans = 0.7940 - 0.7701i
```

```
-lambertw(-1,-a)   % 0.7940 - 0.7701i
```

```
ans = 0.7940 + 0.7701i
```

```
% Ei reaalisia ratk.
% 2)
a=-1
```

```
a = -1
```

```
x0=-lambertw(0,-a)  % -0.5671
```

```
x0 = -0.5671
```

```
-lambertw(-1,-a)    % 1.5339 + 4.3752i
```

```
ans = 1.5339 + 4.3752i
```

```
% 1 reaalinen ratkaisu
% 3)
a=0.2
```

```
a = 0.2000
```

```
x0=-lambertw(0,-a)  % 0.2592
```

```
x0 = 0.2592
```

```
x1=-lambertw(-1,-a) % 2.5426
```

```
x1 = 2.5426
```

```
% 2 reaalista ratk.
% Kuva sopusoinnussa, zoomaamalla saadaan tarkkuutta.
% Kokeillaan vielä fsolvella, mutta jätetään tässä kommenttien taakse.
```

```
% f=@(a,x) x-a*exp(x)
% a=0.2;
% fsolve(@(x)f(a,x),0.2)  % 0.2592    samat saadaan
% fsolve(@(x)f(a,x),2)    % 2.5426    samat sanat
% a=-1
% fsolve(@(x)f(a,x),-0.5) % -0.5671   ja taas!!
% a=0.5
% fsolve(@(x)f(a,x),0) %" No sol found"
% fsolve(@(x)f(a,x),1+i)%  0.7940 + 0.7701i
```

Esim 3 s. 9

Yhtälö $x^y = y^x$

Ratkaisu: $y = -\frac{x}{\ln x} W(-\frac{\ln x}{x})$

Symmetrian takia y ja x voidaan vaihtaa.

```
x=(1:10)'; % Sarakevektori (heittomerkki transponoi.)
y=-x./log(x).*lambertw(-log(x)./x);
A=[x y x.^y y.^x ] % 4 sarakevektoria vierekkäin
```

```
A = 10×4
    1.0000       NaN       NaN       NaN
    2.0000    2.0000    4.0000    4.0000
    3.0000    2.4781   15.2171   15.2171
    4.0000    2.0000   16.0000   16.0000
    5.0000    1.7649   17.1249   17.1249
    6.0000    1.6242   18.3615   18.3615
    7.0000    1.5301   19.6390   19.6390
    8.0000    1.4625   20.9301   20.9301
    9.0000    1.4114   22.2229   22.2229
   10.0000    1.3713   23.5119   23.5119
```

```
T=array2table(A,'VariableNames',{'x','y','x^y','y^x'}) % Taulukko otsikoineen
```

T = 10×4 table

	x	y	x^y	y^x
1	1	NaN	NaN	NaN
2	2	2	4	4
3	3	2.4781	15.2171	15.2171
4	4	2	16	16
5	5	1.7649	17.1249	17.1249
6	6	1.6242	18.3615	18.3615
7	7	1.5301	19.6390	19.6390
8	8	1.4625	20.9301	20.9301
9	9	1.4114	22.2229	22.2229
10	10	1.3713	23.5119	23.5119

```
% Tämäkin hieno funktio on uusissa versioissa, kuten Octave-online:ssa.
```

Edellisen datan suhteen ratkaisu ainakin toimii. Katsotaan vielä kuvia.

Implisiittimuotoinen piirros 0-korkeuskäyränä, "raakaa voimaa" käyttäen:

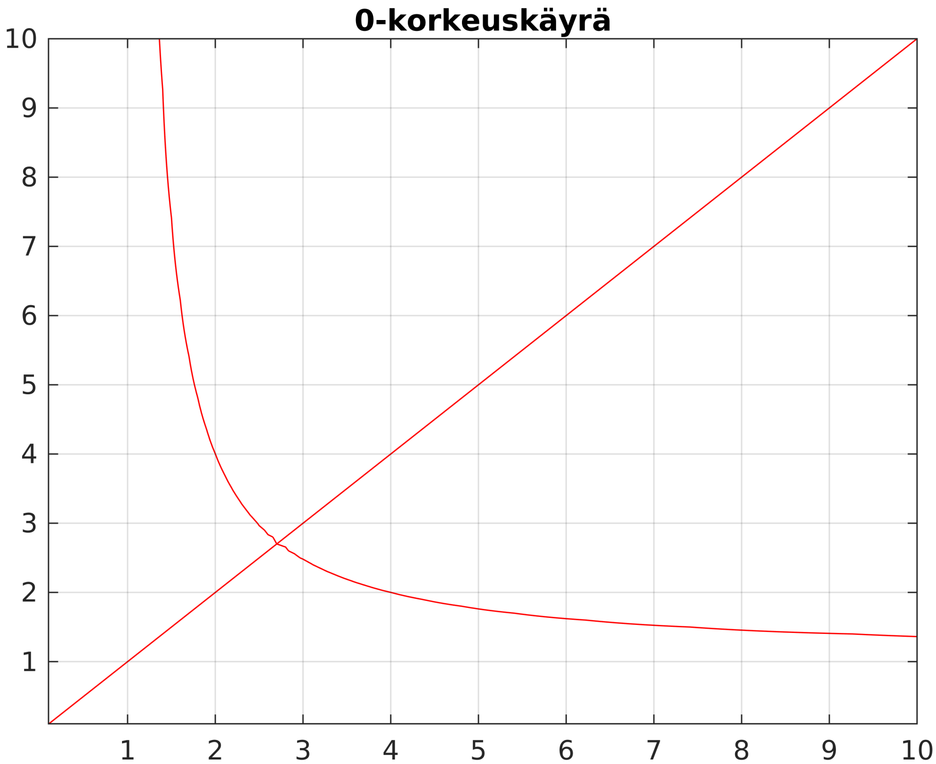
```
close all
F=@(x,y) x.^y - y.^x
```

```
F = function_handle with value:

    @(x,y)x.^y-y.^x
```

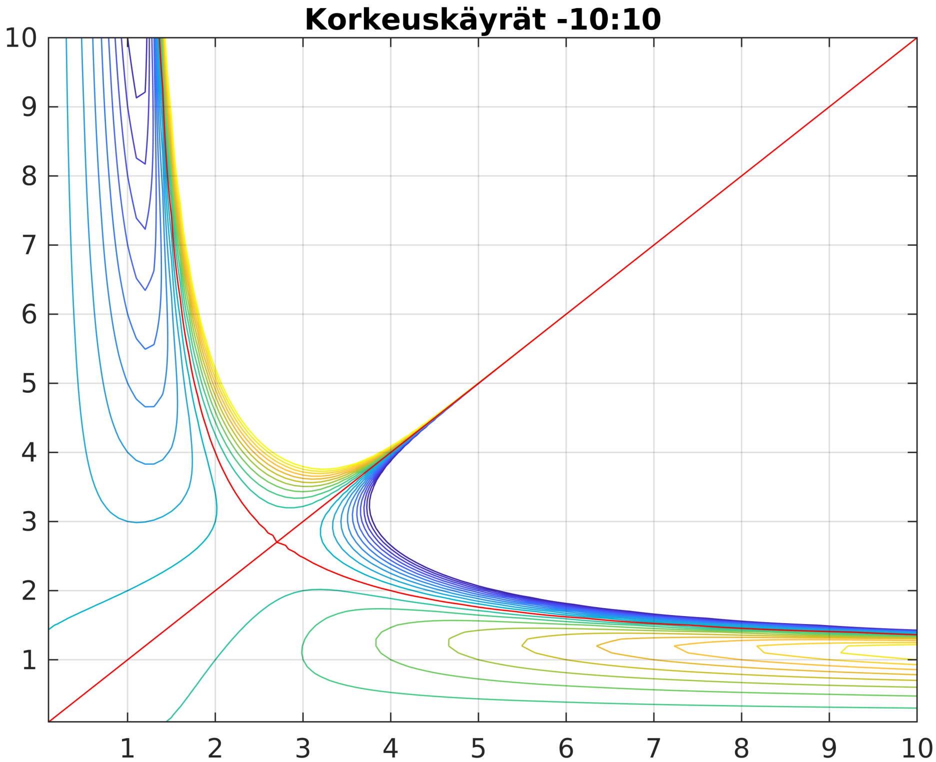
```
x=linspace(.1,10); y=x;
[X,Y]=meshgrid(x,y);          % Tason hilapisteistö
contour(x,y,F(X,Y),[0 0],'r') % 0-korkeuskäyrä.
```

```
title('0-korkeuskäyrä')
grid on
```



Piirretään vielä huvin vuoksi korkeuskäyräpiirros korkeuksille -10,-9,...,0,1,...,10

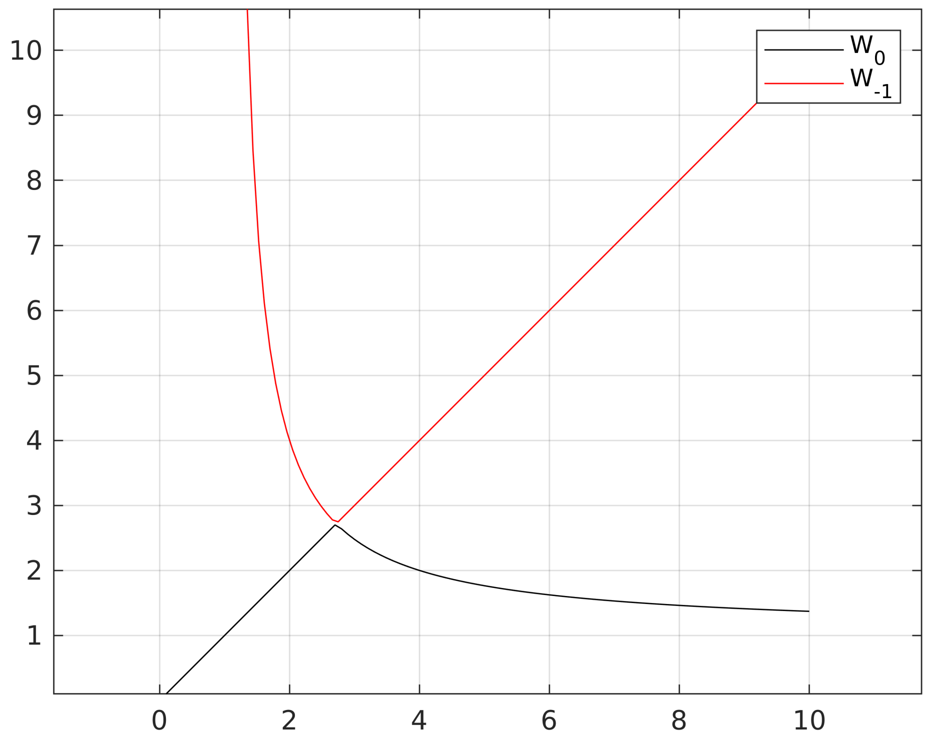
```
figure
contour(x,y,F(X,Y),-10:10)
hold on
contour(x,y,F(X,Y),[0 0],'r')
title('Korkeuskäyrät -10:10')
grid on
```




```
shg
```

Nyt se elegantti Lambert-ratkaisu

```
close all
x=linspace(.1,10);
y=-x./log(x).*lambertw(-log(x)./x); % W_0
plot(x,y,'k')
x=linspace(1.35,10);  % Huomaa haaran W_{-1} määrittelyjoukko
hold on
y=-x./log(x).*lambertw(-1,-log(x)./x); % W_{-1}
plot(x,y,'r')
axis equal
grid on
legend('W_0','W_{-1}')
```

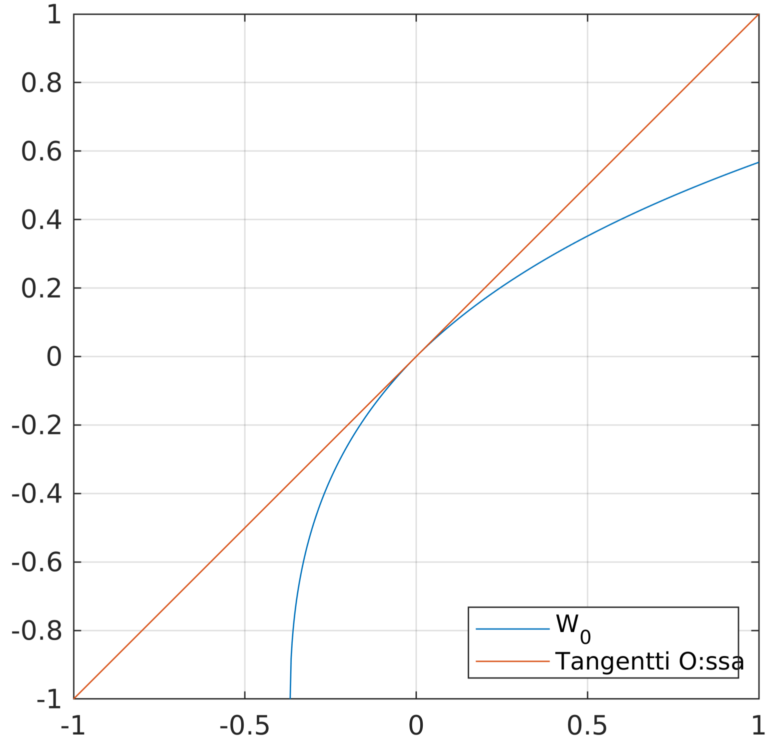


Käänteisfunktion derivoimiskaavan avulla W'(0):

$$W'(0) = \frac{1}{f'(W(0))} = \frac{1}{e^0} = 1$$

Tarkistetaan kuvasta:

```
clf
e=exp(1);
x=linspace(-1/e,1,500);
plot(x,lambertw(x))
hold on
plot([-1 1],[-1 1])
legend('W_0','Tangentti O:ssa','Location','SouthEast')
grid on
axis equal
axis square
```



Kaikki hyvin!