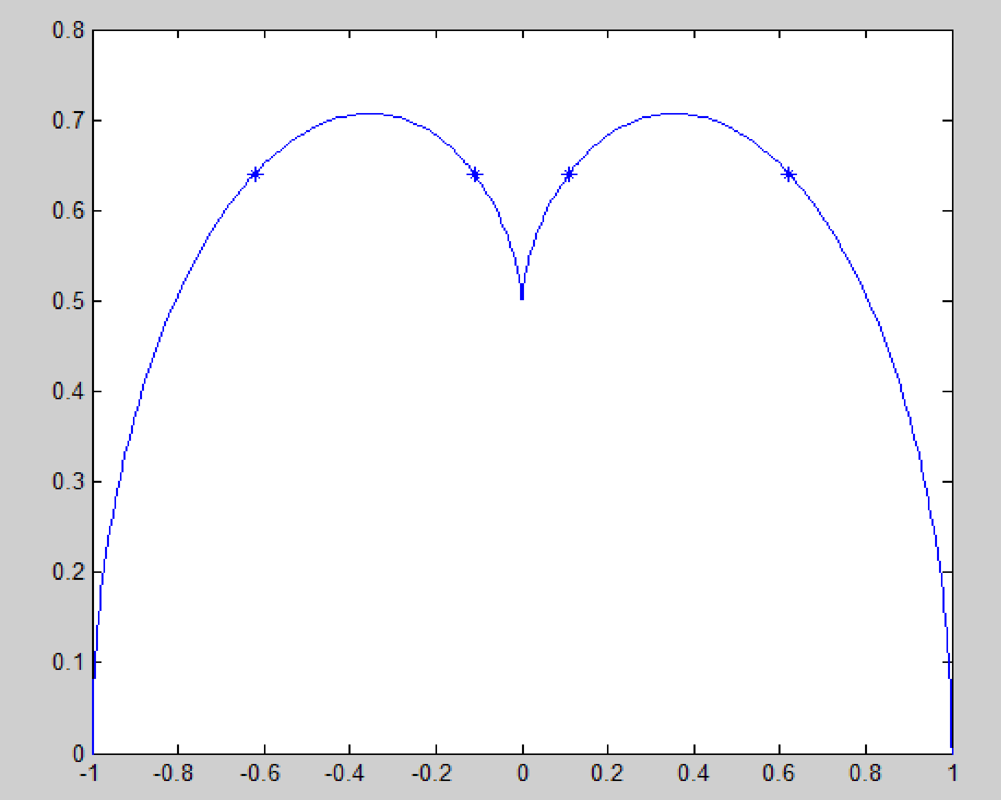
Uppgift 1.

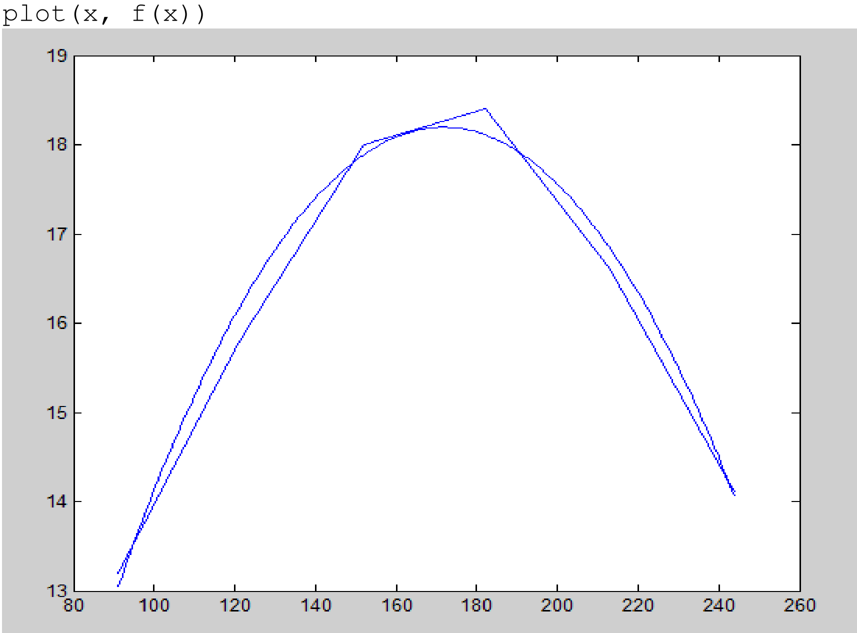
```
phi = [0:0.01:pi/2];
x = cos(phi).^3; % x är beroende av phi
y = 1.5*sin(phi)-sin(phi).^3;

plot(x, y) % Rita halva kaustikan
hold on;
plot(-x, y) % Spegla halvan i y-axeln
hold on;

h = 10;
format compact
phi = acos(0.6^(1/3)); % Phi when x=0.6
while abs(h) > 1.0e-10*abs(x),
    f = 1.5*sin(phi)-sin(phi)^3 -0.64;
    fp = -3*cos(phi)*(-0.5+sin(phi)^2);
    h = f/fp;
    disp([phi, f, fp h]);
    phi = phi-h;
end
% konvertera från phi till x
x = cos(phi)^3;
% rita stjärnor där y = ~0.64
plot(x, 0.64, '*')
plot(-x, 0.64, '*')

h = 10;
phi = acos(0.1^(1/3)); % Phi when x=0.1
while abs(h) > 1.0e-10*abs(phi),
    f = 1.5*sin(phi)-sin(phi)^3 -0.64;
    fp = -3*cos(phi)*(-0.5+sin(phi)^2);
    h = f/fp;
    phi = phi-h;
end
x = cos(phi)^3;
plot(x, 0.64, '*')
plot(-x, 0.64, '*')
hold off
```



Uppgift 2.

a)

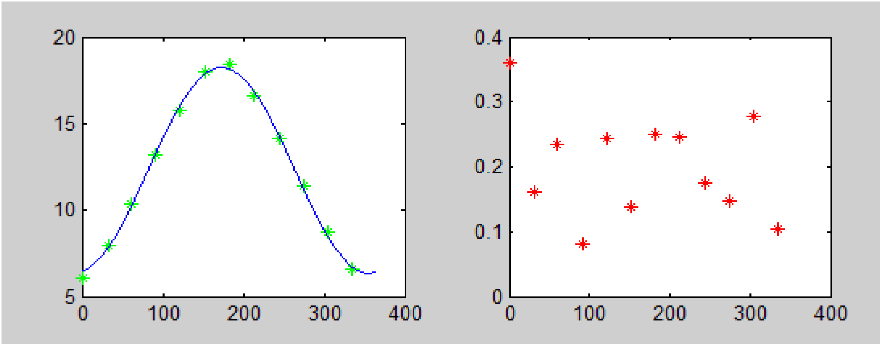
```
x = [91 121 152 182 213 244];
y = [13.2 15.8 18.0 18.4 16.6 14.1];
A = [x'.^2 x' ones(size(x'))];
coeffs = A\y';
f = @(x) coeffs(1)*x.^2 + coeffs(2)*x + coeffs(3);

plot(x, y);
hold on;
x = [91:244];
plot(x, f(x))
```



b)

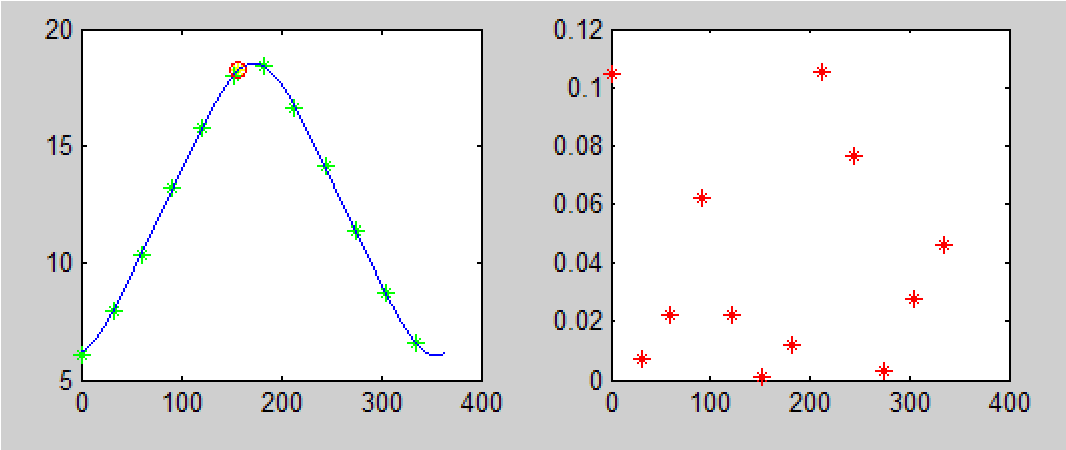
```
hold off;
```

```
t = [1 32 60 274 305 335 91 121 152 182 213 244];
y = [6.1 8.0 10.4 11.4 8.7 6.6 13.2 15.8 18.0 18.4 16.6 14.1];
T = 365;
w = 2*pi/T;
A = [ones(size(t')) cos(w*t') sin(w*t')];
c = A\y';
subplot(2, 2, 1)
plot(t, y, 'g*')
hold on
f = @(t) c(1) + c(2)*cos(w*t) + c(3)*sin(w*t);
subplot(2, 2, 2)
plot(t, abs(f(t)-y), 'r*')
subplot(2, 2, 1)
t2 = 1:365;
plot(t2, f(t2))
```



c)

```
t = [1 32 60 274 305 335 91 121 152 182 213 244];
t = t';
y = [6.1 8.0 10.4 11.4 8.7 6.6 13.2 15.8 18.0 18.4 16.6 14.1];

w = 2*pi/365;
k = 3; % 9 funkar ungefär lika bra
A = [ones(size(t)) cos(w*t) sin(w*t) cos(k*w*t) sin(k*w*t)];

c = A\y';

f = @(t) c(1) + c(2)*cos(w*t) + c(3)*sin(w*t) + c(4)*cos(k*w*t) +
c(5)*sin(k*w*t);
subplot(2, 2, 4)
plot(t, abs(f(t')-y), 'r*')

subplot(2, 2, 3)
plot(t, y, 'g*')
hold on
t2 = 1:365;
plot(t2, f(t2))

disp(sum((f(t')-y).^2)) % Felkvadratsumma
disp(f(157)) % Nationaldagen
plot(157, f(157), 'rO') % Utritad nationaldag
```



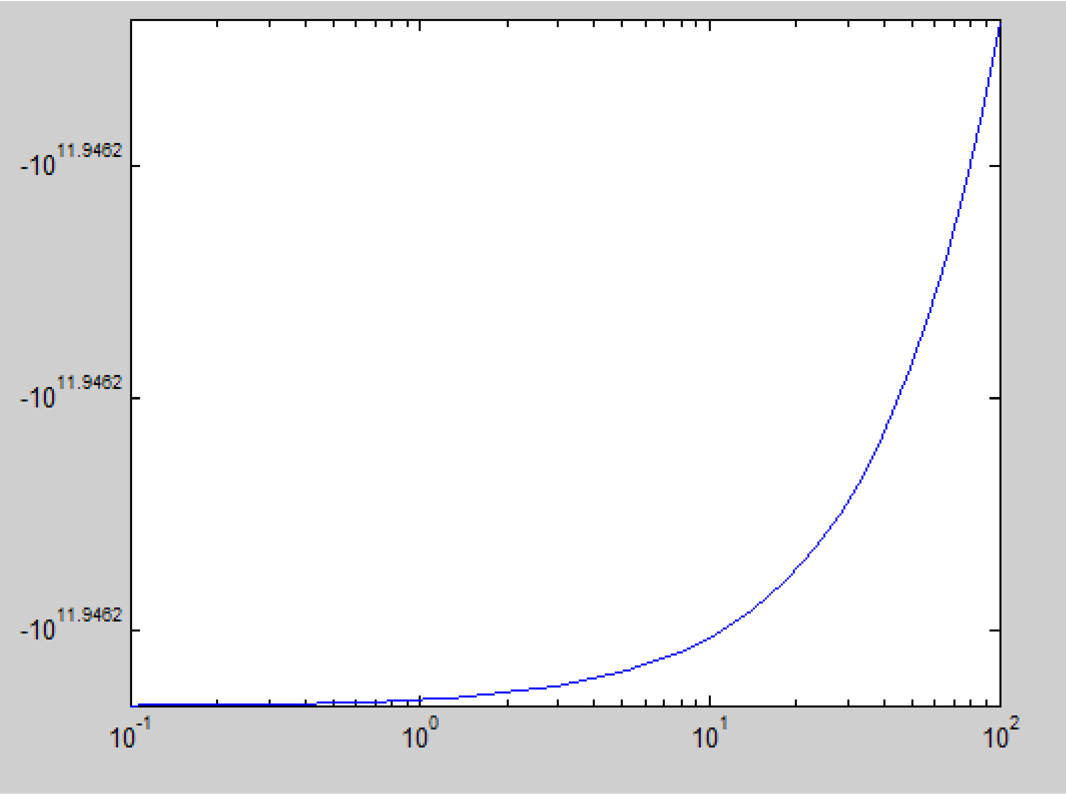
Felkvadratsumma 0.0358
Nationaldagslängd 18.2186

Uppgift 3.

```
% Noggrannhetsordning = p dÃ¤r h^p * konstant = trunkeringsfelet (1 fÃ¶r
% front och 2 fÃ¶r central)

% funktioner (f, f-prim, f-prim med centraldifferens och f-prim med
% framÃ¥tdifferens [uppgift a) och uppgift b)]
f = @(x) 60*x - (((x.^2 + x + 0.1).^6)/((x+1).^6)) - 10 * x *exp(-x);
fp = @(x) 60 - 6 * (2*x + 1) * (x.^2 + x + 0.1).^5 / ((x+1).^6) + 6 * (x.^2 + x
+ 0.1).^6 / ((x+1).^7) + 10 * x * exp(-x) - 10 * exp(-x);
fpc = @(x, hh) (f(x+hh) - f(x-hh))/(2*hh);
fpf = @(x, hh) (f(x+hh) - f(x))/(hh);

% uppskattar derivatans vÃ¤rde numeriskt och analytiskt i x=0.2 och x=1 [uppgift
c)]
% steglÃ¤ngden vid approximationen
hh = 10^-6

% x = 0.2
f_prim_0_2 = fp(0.2)
f_prim_central_0_2 = fpc(0.2, hh)
f_prim_front_0_2 = fpf(0.2, hh)

% x = 1
f_prim_1 = fp(1)
f_prim_central_1 = fpc(1, hh)
f_prim_front_1 = fpf(1, hh)
```

```
F = [[hh hh]; [f_prim_0_2 f_prim_1]; [f_prim_central_0_2 f_prim_central_1];
[f_prim_front_0_2 f_prim_front_1]];

printmat(F', 'c)', 'x=0.2 x=1', 'h prim central front')


% [uppgift d)]
% jÃ¤mfÃ¶r ett diffarna mellan numeriskt och analytiskt fÃ¶r ett flertal
% h-vÃ¤rden (se nedan). Dessa presenteras som enskilda resultat samt med en
% diff mellan fpc och fp.

% en lista av alla approximationssteg h
h=[1.E-3 1.E-4 1.E-5 1.E-6 1.E-7 1.E-8 1.E-9 1.E-10 1.E-11 1.E-12 1.E-13];

% itererar Ã¶ver h fÃ¶r att generera all data
for i = 1:11
    diff_fpc(i) = fpc(1, h(i));
    diff_fp(i) = fp(1);
    diff_fpc_fp(i) = diff_fpc(i) - diff_fp(i);
end

% skapar matrisen med all data
M = [h; diff_fpc_fp; diff_fpc; diff_fp];

% skriver ut matrisen
printmat(M', 'Test', 'Row1 Row2 Row3 Row4 Row5 Row6 Row7 Row8 Row9 Row10 Row11',
'h (fpc-fp) fpc fp')

% sÃ¤tter upp vÃ¤rden/variabler fÃ¶r att plotta funktionen i loglog-diagram
xx = logspace(-1,2);
yy = 60 .* xx - (((xx.^2 + xx + 0.1).^6)/((xx+1).^6)) - 10 .* xx .* exp(-xx);

% plottar funktionen
loglog(xx, yy)
```

```
Test =

                    h      (fpc-fp)          fpc           fp
     Row1       0.00100 -2.30819e-05     52.53373     52.53375
     Row2       0.00010 -2.30805e-07     52.53375     52.53375
     Row3   1.00000e-05 -1.83298e-09     52.53375     52.53375
     Row4   1.00000e-06 -2.18824e-09     52.53375     52.53375
     Row5   1.00000e-07  2.62335e-08     52.53375     52.53375
     Row6   1.00000e-08 -5.77728e-07     52.53375     52.53375
     Row7   1.00000e-09  4.88086e-07     52.53375     52.53375
     Row8   1.00000e-10  1.11462e-05     52.53376     52.53375
     Row9   1.00000e-11     -0.00049     52.53327     52.53375
    Row10   1.00000e-12      0.00022     52.53398     52.53375
    Row11   1.00000e-13      0.01088     52.54464     52.53375
```



Uppgift 4.

a) x^6 + 2 * x^4 - 2 * x^2 = 10^6
x = 10 => 1019800 = ~10^6

b) x - 10^-3 + x^4 = 0
x = 10^-3 => ~0

c) x + x^4 = 100
x = 4 => 260
x = 3 => 84
x = 3.1 => ~92.16
x = 3.2 => ~103

d) x^-4 + e^(x-100) = 10^4
x = 4*ln(10) + 100 = ~109 => ~10^4

Uppgift 5.

```
% FrÃ¥gor:
% * Uppvisar iterationerna kvadratisk konvergens?
% Svar: Ja, som man kan se i utskriften disp(h)
%
% * Kan du uppskatta den asymptotiska felkonstanten fÃ¶r de tvÃ¥ rÃ¶tterna. Hur
skattas felet?
% Svar: f'(x)/2f''(x) fÃ¶r de x-vÃ¤rden som finns fÃ¶r rÃ¶tterna
%
% * Ange en grÃ¤ns fÃ¶r felet?
% Svar: Felet avgrÃ¤nsas till 10^-7. DÃ¥ felet kommer vara <= abs(f / fp * x)
sÃ¥ kommer
% det i den hÃ¤r uppgiften att vara 10^-6. Men dÃ¥ talet kan vara lika med 10^-6
sÃ¥ vÃ¤ljer
% vi istÃ¤llet 10^-7 som felgrÃ¤ns.
```

```
% funktionen
f = @(x) 60*x - (((x.^2 + x + 0.1).^6)/((x+1).^6)) - 10*x.*exp(-x);
fp = @(x) 60 - 6 * (2*x + 1) * (x.^2 + x + 0.1).^5 / ((x+1).^6) + 6 * (x.^2 + x
+ 0.1).^6 / ((x+1).^7) + 10 * x * exp(-x) - 10 * exp(-x);
fb = @(x) 6 * (x^2 + x + 0.1)^4 / (x + 1)^6 * (-7 * (x^2 + x + 0.1)^2 / (x +
1)^2 + 12 * (2*x + 1) * (x^2 + x + 0.1) / (x + 1) - 2 * (x^2 + x + 0.1) - 5 *
(2*x + 1)^2) + 10 * exp(-x) * (2 - x)

% funktionen deriveras numeriskt med central differens
% approxiamationssteget Ã¤r 10^-3
hh = 1E-3;
fp = @(x) (f(x+hh) - f(x-hh))/(2*hh);

% itererar med newton-raphson far att hitta nollstÃ¤lle fÃ¶r funktionen
format compact
for x = [0.1, 2]
    h = 10;

    disp(['    h' '         x' '         f(x)'])
    % iterar tills forandringsvÃ¤rdet (h) ar mindre an en miljontedel
    % relativt x
    while abs(h/x) > 1E-7,
        fval = f(x);
        fpval = fp(x);
        h = (fval/fpval);
        disp([h x fval])
        x = x-h;
    end
    x
end

felkonstant_2_e_8 = fp(2.0*1E-8)/(2 * fb(2.0*1E-8))
felkonstant_2_2224 = fp(2.2224)/(2 * fb(2.2224))
```

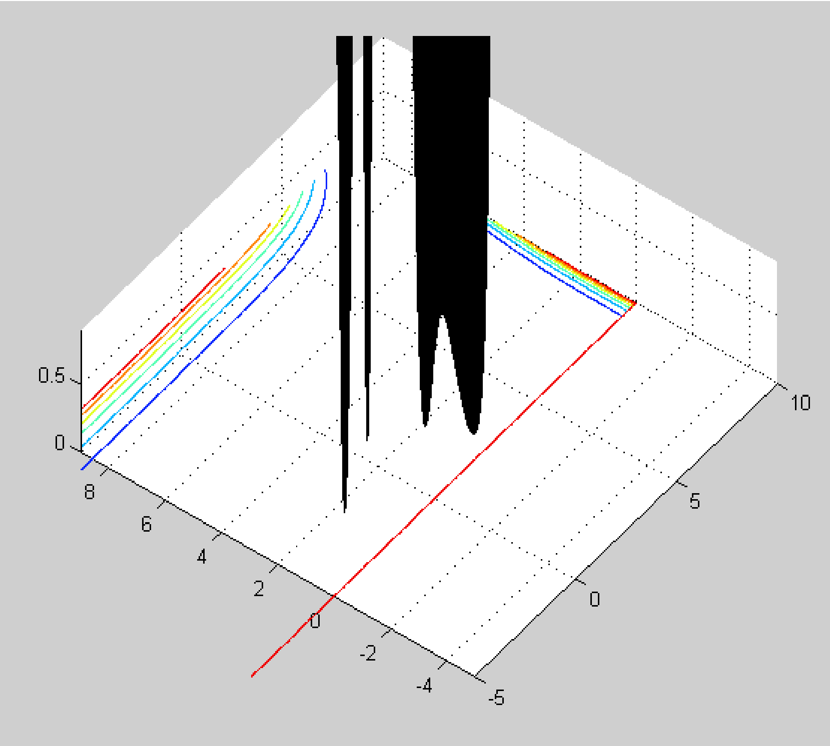
```
      h         x         f(x)
     0.0983    0.1000    5.0951
     0.0017    0.0017    0.0872
   1.0e-04 *
     0.0061    0.0063    0.3030
   1.0e-07 *
     0.0000    0.2000    0.0000
   1.0e-07 *
    -0.0000    0.2000   -0.0000
x =
   2.0000e-08
      h         x         f(x)
    -0.3218    2.0000   46.6206
     0.0879    2.3218  -32.2174
     0.0113    2.2339   -3.3231
     0.0002    2.2226   -0.0497
     0.0000    2.2224   -0.0000
x =
     2.2224
felkonstant_2_e_8 =
     1.2502
felkonstant_2_2224 =
     0.1866
```

Uppgift 6.

Hittar fyra par av y och z som startvärden för newton-raphson

```
f1 = @(x, y, z) sin(x) + y.^2 + log(z) - 3;
f2 = @(x, y, z) 3*x +    2.^y - z.^3;
f3 = @(x, y, z) x.^2 +    y.^2 + z.^3 - 6;

elx = @(y, z) (z.^3 - 2.^y)/3; % value of x from f2
% Derivative of elx by y and z
delxy = @(y, z) (z.^3 - log(2)*2.^y)/3;
delxz = @(y, z) (3*z.^2 - 2.^y)/3;

%substitute x for elx in f1 and f3
f1n = @(y, z) f1(elx(y, z), y, z);
f3n = @(y, z) f3(elx(y, z), y, z);

% Derivative of ¨f1n and f3n by y and z
df1ny = @(y, z) delxy(y, z)*cos(elx(y,z)) + 2*y;
df1nz = @(y, z) delxz(y, z)*cos(elx(y,z)) + 1/z;
df3ny = @(y, z) delxy(y, z)*2*elx(y, z) + 2*y;
df3nz = @(y, z) delxz(y, z)*2*elx(y, z) + 3*z.^2;

% Phi as described by assignment
phi = @(y, z) (f1n(y, z)-3).^2 + (f3n(y, z)-6).^2;

[X, Y] = meshgrid([-10:0.02:10], [0:0.01:10]);
t = [1:100];
surfc(X, Y ,phi(X, Y))
% Shows the interesting intervall
```

```
axis([-5 10 -5 9 0 0.9])
```



Newton-raphson:

```
% Definieras i första delen av 6
f1
f2
f3

elx
delxy
delxz
f1n
f3n

% Derivator för f1 och f3 med avseende på y eller z
df1ny = @(y, z) delxy(y, z)*cos(elx(y,z)) + 2*y;
df1nz = @(y, z) delxz(y, z)*cos(elx(y,z)) + 1/z;
df3ny = @(y, z) delxy(y, z)*2*elx(y, z) + 2*y;
df3nz = @(y, z) delxz(y, z)*2*elx(y, z) + 3*z.^2;

i = 0;
dtnorm = 10;
ys = [2.5 -2 2.2 -3];
zs = [1.8 1.9 0.3 0];
result = [];
disp(['     h'])
for n = 1:4
    t = [ys(n) zs(n)];
     % Dålig konvergens kräver högt tak för i
    while dtnorm > 1E-6 & i < 20,
        y = t(1);
```

```
        z = t(2);
        f = [f1n(y,z);
             f3n(y,z)];
        J = [df1ny(y,z) df1nz(y,z);
             df3ny(y,z) df3nz(y,z)];
        dt = -J\f;
        t = t+dt';
        dtnorm = norm(dt, Inf); % Felvektorns längd
        i = i+1;
        disp([dtnorm])
    end
    result = [result; t];
end
result
i
```

```
    h
    0.5844
    0.2107
    0.0820
    0.0483
    0.0252
    0.0121
    0.0056
    0.0025
    0.0011
   4.9930e-04
   2.2116e-04
   9.7863e-05
   4.3287e-05
   1.9143e-05
   8.4649e-06
   3.7430e-06
   1.6550e-06
   7.3181e-07
result =
    1.5272    1.5339
   -2.0000    1.9000
    2.2000    0.3000
   -3.0000         0
i =
    18
```

Result består av [y0, z0
y1, z1
y2, z2
y3, z3]