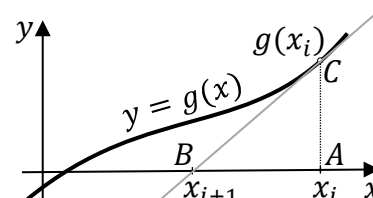


OPGAVE 1

(1a)

Newton-Raphsons metode er baseret på, at der på funktionens graf indlægges en tangentlinje i punktet $(x, y) = (x_i, g(x_i))$ som illustreret på den redigerede version af figur 3. Rødestimatet x_{i+1} bestemmes som den x -værdi, hvor tangentlinjen skærer x -aksen. Til bestemmelse af x_{i+1} benyttes, at tangentlinjens hældningskoefficient er lig med $g'(x_i)$, men samtidig også lig med $\frac{|AC|}{|AB|} = \frac{g(x_i)}{x_i - x_{i+1}}$ (jfr. trekant ABC på figuren). Det kan altså konkluderes, at $g'(x_i) = \frac{g(x_i)}{x_i - x_{i+1}}$, eller idet x_{i+1} isoleres: $x_{i+1} = x_i - \frac{g(x_i)}{g'(x_i)}$ Q.E.D.

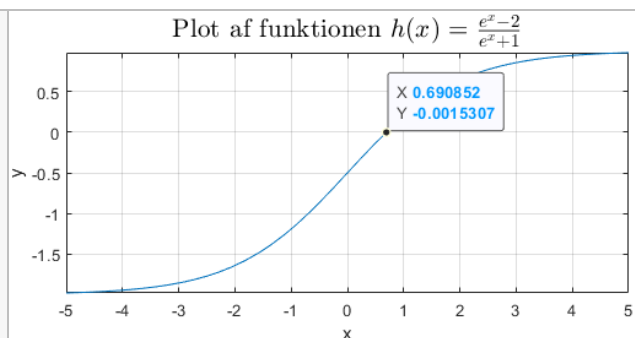


Figur 3: Newton-Raphson-iteration.

(1b)

Her følger kode anvendt til frembringelse af plottet samt det opnåede resultat:

```
h = @(x) (exp(x)-2)./(exp(x)+1);
fplot(h,[-5 5])
title(['Plot af funktionen ' ...
      '$h(x) = \frac{e^x-2}{e^x+1}$'], ...
      'Interpreter','Latex','FontSize',16)
xlabel('x'), ylabel('y'), grid
```



Det ses, at funktionens rod er ca. $\underline{x = 0,7}$.

(1c)

Til gennemførelse af Newton-Raphsons metode skal vi bruge den afledte af $h(x)$: $h'(x) = \frac{3e^x}{(e^x+1)^2}$.

Her følger beregningsresultater opnået i forbindelse med rodfinding vha. Newton-Raphsons metode:

	A	B	C	D	E
1	i	x_i	$h(x_i)$	$h'(x_i)$	$ \varepsilon_a , \%$
2	0	2,5	0,77243	0,21031	
3	1	-1,1728	-1,2909	0,54147	313,17
4	2	1,21135	0,31161	0,53043	196,815
5	3	0,62387	-0,0467	0,68151	94,1663
6	4	0,6924	-0,0005	0,66683	9,89719
7	5	0,69315	-6E-08	0,66667	0,10742

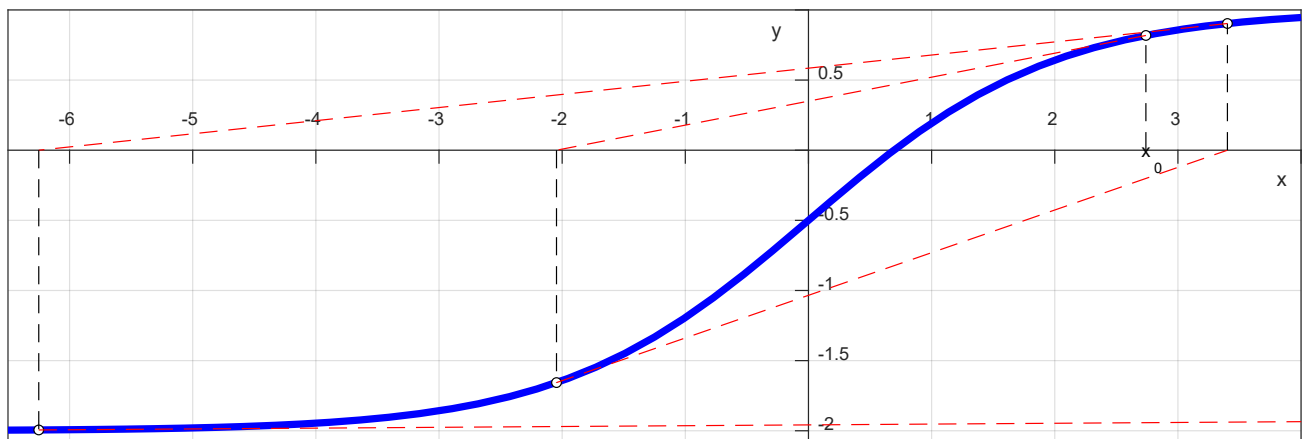
Formlerne bag tallene er som følger:

	A	B	C	D	E
1	i	x_i	$h(x_i)$	$h'(x_i)$	$ \varepsilon_a , \%$
2	0	2,5	$=(\text{EKSP}(B2)-2)/(\text{EKSP}(B2)+1)$	$=3*\text{EKSP}(B2)/(\text{EKSP}(B2)+1)^2$	
3	1	$=B2-C2/D2$	$=(\text{EKSP}(B3)-2)/(\text{EKSP}(B3)+1)$	$=3*\text{EKSP}(B3)/(\text{EKSP}(B3)+1)^2$	$=\text{ABS}((B3-B2)/B3)*100$
4	2	$=B3-C3/D3$	$=(\text{EKSP}(B4)-2)/(\text{EKSP}(B4)+1)$	$=3*\text{EKSP}(B4)/(\text{EKSP}(B4)+1)^2$	$=\text{ABS}((B4-B3)/B4)*100$
5	3	$=B4-C4/D4$	$=(\text{EKSP}(B5)-2)/(\text{EKSP}(B5)+1)$	$=3*\text{EKSP}(B5)/(\text{EKSP}(B5)+1)^2$	$=\text{ABS}((B5-B4)/B5)*100$
6	4	$=B5-C5/D5$	$=(\text{EKSP}(B6)-2)/(\text{EKSP}(B6)+1)$	$=3*\text{EKSP}(B6)/(\text{EKSP}(B6)+1)^2$	$=\text{ABS}((B6-B5)/B6)*100$
7	5	$=B6-C6/D6$	$=(\text{EKSP}(B7)-2)/(\text{EKSP}(B7)+1)$	$=3*\text{EKSP}(B7)/(\text{EKSP}(B7)+1)^2$	$=\text{ABS}((B7-B6)/B7)*100$

Tallet i celle B7, dvs. 0,69315, udgør estimatet af roden efter 5 iterationer. Estimatet er lig med estimatet efter 4 iterationer med op til to decimaler afrundet. Roden med sikre afrundede decimaler er derfor 0,69.

(1d)

For funktionen $h(x)$ vil Newton-Raphsons metode divergere, hvis startgættet, x_0 , ligger tilpas langt fra den faktiske rod. I så fald vil funktionens differentialkvotient nemlig være så tilpas lille, at Newton-Raphsons metode via tangentretningen vil finde et næste rodestimat, x_1 , der ligger længere væk fra den rigtige rod end startgættet (se følgende figur). Her vil funktionens differentialkvotient også være lille, så det følgende rodestimat, x_2 , fjerner sig endnu mere fra roden. Da differentialkvotienten nærmer sig nul for x gående mod såvel plus som minus uendelig, får man på den måde en sekvens af rodestimater, der efterhånden fjerner sig mere og mere fra roden. Altså, Newton-Raphsons metode divergerer.



OPGAVE 2

(2a)

Koden kan færdiggøres på forskellige måder. Her er to muligheder:

```
function d = massemidtdist(m,r)
n = length(m);
M = sum(m);
delta = 360/n;
sum_mx = 0;
sum_my = 0;
for i = 1:n
    theta = (i-1)*delta;
    x = r*cosd(theta);
```

```
function d = massemidtdist(m,r)
n = length(m);
M = sum(m);
delta = 360/n;
sum_mx = sum(m.*r.*cosd((0:n-1)*delta));
sum_my = sum(m.*r.*sind((0:n-1)*delta));
for i = 1:n
    % Gør ingenting
end
```

<pre> y = r*sind(theta); sum_mx = sum_mx + m(i)*x; sum_my = sum_my + m(i)*y; end xc = sum_mx/M; yc = sum_my/M; d = sqrt(xc^2 + yc^2); </pre>	<pre> xc = sum_mx/M; yc = sum_my/M; d = sqrt(xc^2 + yc^2); </pre>
--	---

Afprøvning af funktionen:

```

m = [81 62 65 76 90 102 86 77 107 85 80 77 67 44 60];
r = 3.8;
d = massemidtdist(m,r)

```

d = 0.4195

Massemidtpunktet ligger altså i afstanden 420 mm fra karrusellens centrum.

(2b)

Det vises ikke, hvordan funktionen suboptmindist kan rettes, da det vil afsløre for meget om, hvordan gruppeafleveringen kan løses.

```

function [m,d] = suboptmindist(m0,r)
% Kode ikke vist.

```

Afprøvning af funktionen:

```

m0 = [81 62 65 76 90 102 86 77 107 85 80 77 67 44 60];
r = 3.8;
[m,d] = suboptmindist(m0,r)

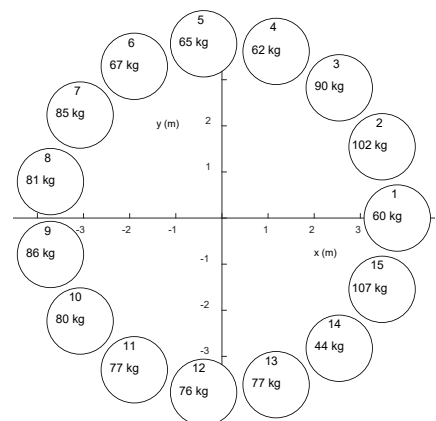
```

m = 1×15

60 102 90 62 65 67 85 81 86 80 77 76 77 44 107

d = 6.1241e-04

Den fundne suboptimale placering af masserne på karrusellens pladser er illustreret til højre. Med denne løsning ligger massemidtpunktet altså blot i afstanden 0.6 mm fra karrusellens centrum.



OPGAVE 3

(3a)

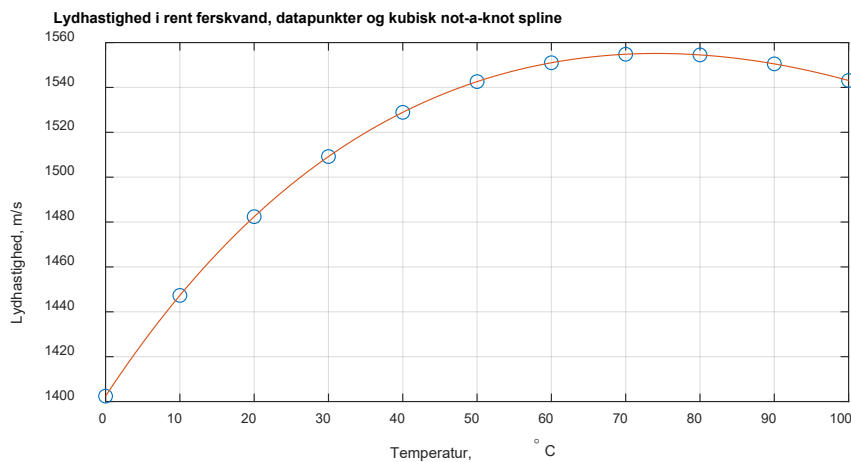
Her følger det efterspurgte plot og estimeret lydhastighed ved 62,3°C inkl. bagvedliggende MATLAB-kode:

```

T = 0:10:100; % grC
c = [1402.4 1447.3 1482.4 1509.2 1528.9 ...
     1542.6 1551.0 1554.8 1554.5 1550.5 1543.1]; % m/s
TT = linspace(0,100);

```

```
cc = spline(T,c,TT);
plot(T,c,'o',TT,cc)
xlabel('Temperatur, \circ C')
ylabel('Lydhastighed, m/s')
title('Lydhastighed i rent ferskvand, datapunkter og kubisk not-a-knot spline')
grid
```



```
c_62_3 = spline(T,c,62.3)
```

```
c_62_3 = 1.5523e+03
```

Lydhastigheden ved 62,3°C beregnes således til 1552,3 m/s.

(3b)

Det undersøges, hvor godt henholdsvis et første-, et tredje- og et femtegradspolynomium estimerer lydhastigheden ved 60°C ud fra de nærmeste nabopunkter i datasættet:

```
% Førstegradspolynomium
Tp = T([6 8]); % Datapunkt 6 og 8 er de to nærmeste "naboer" til 60 grC
cp = c([6 8]);
p = polyfit(Tp,cp,1); % Estimér koeff. i førstegradspolynomium
c60 = polyval(p,60)
```

```
c60 = 1.5487e+03
```

```
% Tredjegradspolynomium
Tp = T([5 6 8 9]); % Fire nærmeste "naboer" til 60 grC
cp = c([5 6 8 9]);
p = polyfit(Tp,cp,3); % Estimér koeff. i tredjegradspolynomium
c60 = polyval(p,60)
```

```
c60 = 1.5510e+03
```

```
% Femtegradspolynomium
Tp = T([4 5 6 8 9 10]); % Seks nærmeste "naboer" til 60 grC
cp = c([4 5 6 8 9 10]);
p = polyfit(Tp,cp,5); % Estimér koeff. i femtegradspolynomium
c60 = polyval(p,60)
```

```
c60 = 1.5510e+03
```

Et førstegradspolynomium (lineær interpolation) giver ikke det korrekte resultat på 1551,0 m/s. Det gør derimod såvel et tredje- som et femtegradspolynomium, og begge disse polynomiumsgrader er

således egnede. Da man bør vælge lavest mulig polynomiumsgrad, er et tredjegradspolynomium at foretrække.

(3c)

Lydhastigheden ved 62,3°C estimeres ved at interpolere med henholdsvis et første-, et tredje- og et femtegradspolynomium:

```
% Førstegradspolynomium
Tp = T(7:8); % Datapunkt 7 og 8 er de to nærmeste "naboer" til 62,3 grC
cp = c(7:8);
p = polyfit(Tp,cp,1); % Estimér koeff. i førstegradspolynomium
c_62_3 = polyval(p,62.3)
```

c_62_3 = 1.5519e+03

```
% Tredjegradspolynomium
Tp = T(6:9); % Fire nærmeste "naboer" til 60 grC
cp = c(6:9);
p = polyfit(Tp,cp,3); % Estimér koeff. i tredjegradspolynomium
c_62_3 = polyval(p,62.3)
```

c_62_3 = 1.5523e+03

```
% Femtegradspolynomium
Tp = T(5:10); % Seks nærmeste "naboer" til 60 grC
cp = c(5:10);
p = polyfit(Tp,cp,5); % Estimér koeff. i femtegradspolynomium
c_62_3 = polyval(p,62.3)
```

c_62_3 = 1.5523e+03

Både et tredje- og et femtegradspolynomium, som i delopgave (3b) begge bedømtes egnede til interpolation i omegnen af 60°C, estimerer lydhastigheden til at være 1552,3 m/s ved 62,3°C. Estimatet stemmer i øvrigt overens med *spline*-resultatet i delopgave (3a). Konklusionen er derfor, at et godt estimat af lydhastigheden ved 62,3°C er 1552,3 m/s.

PS! Hvis man ikke har løst delopgave (3b) og evt. heller ikke (3a), kan man hæfte sig ved, at et plot af dataværdierne afslører, at sammenhængen mellem temperatur og lydhastighed tydeligvis ikke er lineær, hvorfor lineær interpolation (et førstegradspolynomium) næppe giver det mest troværdige resultat. Derfor bør man stole mest på det samstemmende resultat, som opnås med både et tredje- og et femtegradspolynomium.

(3d)

Ligningerne kan udledes som følger:

Polynomium går igennem de tre datapunkter:

$$\text{Punkt 1: } c(0) = 1402,4 \Rightarrow p_1 \times 0^3 + p_2 \times 0^2 + p_3 \times 0 + p_4 = 1402,4 \Rightarrow \underline{p_4 = 1402,4}$$

$$\begin{aligned} \text{Punkt 2: } c(74,17) = 1555,1 &\Rightarrow p_1 \times 74,17^3 + p_2 \times 74,17^2 + p_3 \times 74,17 + p_4 = 1555,1 \\ &\Rightarrow \underline{408023,180713p_1 + 5501,1889p_2 + 74,17p_3 + p_4 = 1555,1} \end{aligned}$$

$$\begin{aligned} \text{Punkt 3: } c(100) = 1543,1 &\Rightarrow p_1 \times 100^3 + p_2 \times 100^2 + p_3 \times 100 + p_4 = 1543,1 \\ &\Rightarrow \underline{1000000p_1 + 10000p_2 + 100p_3 + p_4 = 1543,1} \end{aligned}$$

Polynomiet har lokalt maksimum, når $T = 74,17^\circ\text{C}$, dvs. polynomiets afledte, $c'(T) = 3p_1T^2 + 2p_2T + p_3$, er lig med nul, når $T = 74,17^\circ\text{C}$:

$$c'(74,17) = 0 \Rightarrow 3p_1 \times 74,17^2 + 2p_2 \times 74,17 + p_3 = 0$$

$$\Rightarrow \underline{16503,5667p_1 + 148,34p_2 + p_3 = 0}$$

Q.E.D

(3e)

Det givne datasæt defineres som vektorer T og c :

```
T = [    0  74.17   100]; % grC
c = [1402.4 1555.1 1543.1]; % m/s
```

Ligningssystemet kan opstilles på matrixform, $\mathbf{A}\mathbf{p} = \mathbf{B}$, hvor p er en søjlevektor indeholdende de ubekendte koefficienter i polynomiet ($\mathbf{p} = [p_1 \ p_2 \ p_3 \ p_4]^T$), mens \mathbf{A} og \mathbf{B} er matricer indeholdende henholdsvis koefficienter og højresidekonstanter fra ligningerne defineret under delopgave (3d):

```
format longg
A = [T(1)^3    T(1)^2 T(1) 1
      T(2)^3    T(2)^2 T(2) 1
      T(3)^3    T(3)^2 T(3) 1
      3*T(2)^2 2*T(2) 1    0]
```

```
A = 4x4
      0      0      0      1
408023.180713 5501.1889 74.17 1
      1000000      10000      100 1
16503.5667      148.34      1    0
```

```
B = [c'; 0]
```

```
B = 4x1
      1402.4
      1555.1
      1543.1
      0
```

Koefficienterne bestemmes ved at løse ligningen ($\mathbf{p} = \mathbf{A}^{-1}\mathbf{B}$):

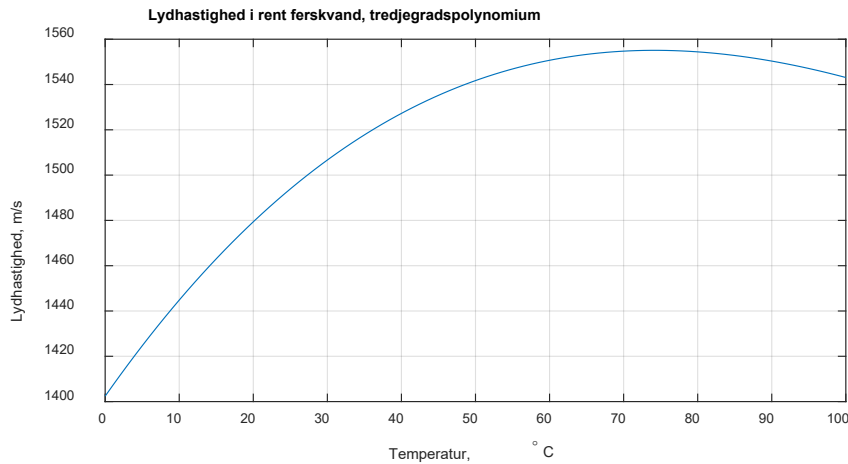
```
p = A\B
```

```
p = 4x1
      9.77172543524025e-05
      -0.0422530136826999
      4.65512882474596
      1402.4
```

Med `linspace` og `polyval` oprettes vektorer, TT og cc , så polynomiet kan plottes:

```
format short
TT = linspace(0,100);
cc = polyval(p,TT);
plot(TT,cc)
xlabel('Temperatur, \circC')
```

```
ylabel('Lydhastighed, m/s')
title('Lydhastighed i rent ferskvand, tredjegradspolynomium')
grid
```

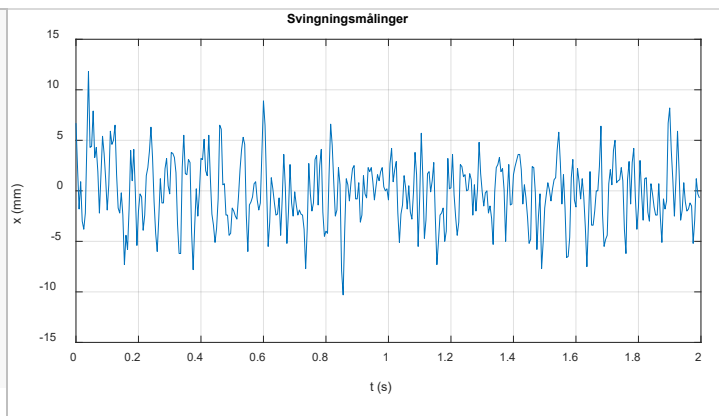


OPGAVE 4

(4a)

Her følger plottet og tilhørende MATLAB-kode:

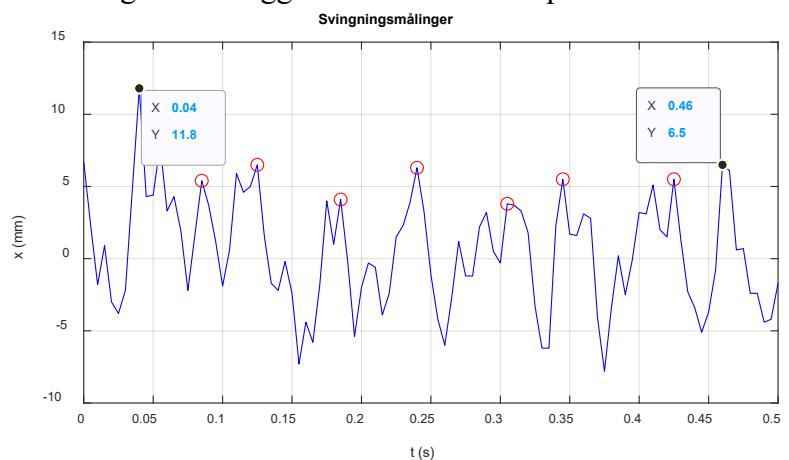
```
svingningsdata % Indlæs vektor x, mm
n = length(x);
dt = 0.005; % s
t = (0:n-1)*dt; % s
plot(t,x,'b')
title('Svingningsmålinger')
xlabel('t (s)'), ylabel('x (mm)')
grid
```



Ud fra plottet er det ikke umiddelbart muligt at identificere et svingningsmønster med fast periode, så det vil være rimeligt at konkludere, at det er umuligt at fastlægge frekvensen ud fra plottet. Der kan dog evt. alligevel gøres et forsøg som følger:

Til højre er vist et plot, hvor der er zoomet ind på det første halve sekund, hvor svingningerne bør være størst og dermed mest tydelige. Som skitseret er der tilsyneladende 8 dominerende svingninger i tidsrummet fra $t = 0,04$ til $0,46$ s svarende til en frekvens på

$$\frac{8}{0,46-0,04} = \underline{\underline{19,0 \text{ Hz}}}.$$



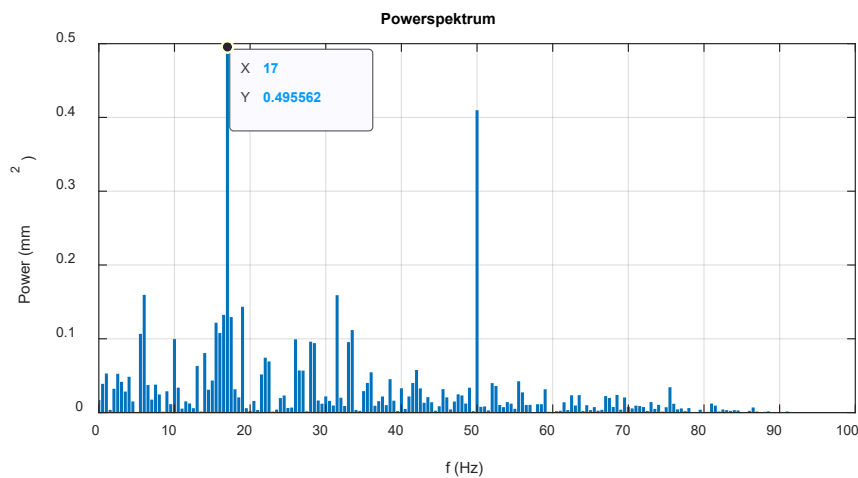
(4b)

Måleseriens powerspektrum beregnes og plottes med følgende MATLAB-kode:

```

svingningsdata % Indlæs vektor x (mm)
n = length(x); % Antal samplinger
dt = 0.005;    % Samplingsinterval (s)
fs = 1/dt;     % Samplingsfrekvens (Hz)
T = n*dt;     % Måleseriens længde (s)
X = fft(x)/n;  % Fast Fouriertransformation
fmax = fs/2;   % Maks. detekterbare frekvens (Hz)
df = 1/T;      % Frekvensstep (Hz)
P = real(X).^2 + imag(X).^2; % Power-vektor
f = (0:n-1)*df; % Frekvensvektor (Hz)
bar(f,P);      % Plot power som søjler
xlim([0 fmax]); % Afgræns frekvensaksen
xlabel('f (Hz)'); ylabel('Power (mm^2)');
title('Powerspektrum');
grid

```



Powerværdierne er markant højest ved henholdsvis 17 og 50 Hz, og disse er således de to dominerende frekvenser. Det kan altså konkluderes, at måleserien er under indflydelse af 50 Hz støj, og at masse-fjeder-dæmpersystems svingningsfrekvens kan identificeres til 17 Hz.