

Aufgabe 3.1 Alte Rechenaufgabe

```
% Aufgabe 3.1
% Drei Reisende finden, waehrend sie zusammen so dahingehen, einen Ranzen
% mit 73 Gulden.

% Sie stellen fest, dass diese 73 Gulden zusammen mit der
% Barschaft der ersten beiden das Doppelte dessen sind, was
% der erste und dritte und der zweite und dritte gemeinsam bei sich haben.

% Diese 73 Gulden ergeben aber auch mit dem Besitz des ersten und dritten
% das Dreifache dessen, was der zweite und dritte und der erste und zweite
% mit sich fuehren.

% Und schliesslich ergeben diese 73 Gulden zusammen mit der Barschaft des
% zweiten und dritten das Vierfache dessen, was der erste und zweite
% und der erste und dritte zusammen bei sich tragen.

% Wie viele Gulden besitzt jeder?

% Setzen Sie dieses Skript mit Ihrer Lösung fort.
% Geben Sie die Ergebnisse in Sätzen aus (Strings benutzen).
% Testen Sie anschließend die Ergebnisse und geben Sie auch diese
% Antworten aus.

%*****
echo off
% Loesungsvorschlag:
%
% Umformung der Gleichungen:
%  $B_1 + B_2 + 4 \cdot B_3 = 73$ 
%  $2 \cdot B_1 + 6 \cdot B_2 + 2 \cdot B_3 = 73$ 
%  $8 \cdot B_1 + 3 \cdot B_2 + 3 \cdot B_3 = 73$ 
% In Matlab Vektor-Matrix-Schreibweise:  $A \cdot x = b$ 
A = [1 1 4 ; 2 6 2 ; 8 3 3];
b = 73*[1;1;1];
% Test: Berechnung der Determinante und Ausgabe
% da = det(A);
% disp(['Determinante von A ist ' num2str(da)]);
% Lösung des Gleichungssystems
x = A\b;
% Ausgabe der Ergebnisse für die Barschaften
for i=1:length(x)
disp(['Barschaft des ' num2str(i) '. Reisenden: ' num2str(x(i))]);
end
disp('-----');
% Test der Ergebnisse
disp('Test der Ergebnisse: Einsetzen in 1. Gleichung: ');
disp([num2str(x(1)), ' + ', num2str(x(2)), ' + 4*', num2str(x(3)), ...
' = ', num2str(x(1)+x(2)+4*x(3))]);
disp('-----');
disp('Test der Ergebnisse: Einsetzen in 2. Gleichung: ');
disp(['2*', num2str(x(1)), ' + 6*', num2str(x(2)), ' + 2*', ...
num2str(x(3)), ' = ', num2str(2*x(1)+6*x(2)+2*x(3))]);
disp('-----');
disp('Test der Ergebnisse: Einsetzen in 3. Gleichung: ');
disp(['8*', num2str(x(1)), ' + 3*', num2str(x(2)), ' + 3*', ...
num2str(x(3)), ' = ', num2str(8*x(1)+3*x(2)+3*x(3))]);
```

Aufgabe 3.2 Widerstandsnetzwerk

```
% Aufgabe 3.2
% Kirchhoffsche Regeln
%
% Sortieren der Gleichungen nach I1...I5, U,V,W auf rechte Seite bringen
%
% Gleichungssystem der Struktur A*x=b mit
% x = [I1 I2 I3 I4 I5]'

% Parameter, zum Spielen auch Veraendern!
R = 300;
U = 300;
V = 300;
W = 200;

% Matrix A
%      I1    I2    I3    I4    I5
A = [  1    -1     1     0    -1
      0     0     1     1    -1
      0     0     R     0     R
      3*R    R     0     0     0
      0    -R    -R    2*R     0];

b = [0 ; 0 ; W ; U+V ; -V-W];

% Lösung
x = A\b;
% Alternative:
% x = inv(A)*b;

% Ausgabe
for i=1:5
    disp(['Strom I'    num2str(i) ' = ' num2str(x(i))    ' A']);
end
```

zu Aufgabe 3.3 Exzentrischer Stoß

```
% Aufgabe 3.3
% Exzentrischer Stoss benoetigt Funktion fmat33
close all; clear all;

% Globale Parameter (auch in Funktion fmat33 verfügbar)
global M a b
% Modellparameter:
% [kg]; [m]; [m]; [kg]; [m/s]; []
M=1; a=0.4; s=0.3; m=2; v=10; e=0.1;

% Umsortieren der Gleichungen in A*x=b; Vektor der rechten Seite: b
b = [0 ; 0 ; m*v*(1+e) ; 0 ; 0];

% Matrix ist vom zu varierenden Parameter s abhängig. Deshalb Matrix
% und Lösung für s=0.3 aus Funktion fmat33
x=fmat33(s);

disp('Für s = 0.3 m lauten dei Unbekannten:')
% Ausgabe. Lösungsvektor x = [Om1 Om2 pa pb pc]
disp(['Omega1 = ' num2str(x(1))]);
disp(['Omega2 = ' num2str(x(2))]);
disp(['deltapa = ' num2str(x(3))]);
disp(['deltapb = ' num2str(x(4))]);
disp(['deltapc = ' num2str(x(5))]);
disp('Für 0 < s < a sind die unbekannten Größen im Graphen angegeben')
% Variation von s zwischen 0 und a

s = 0 : 0.01 : a ;
om1=zeros(0,length(s));
om2=om1; dpa=om1; dpb=om1; dpc=om1;
for i=1:length(s)
    x=fmat33(s(i));
    om1(i)=x(1);
    om2(i)=x(2);
    dpa(i)=x(3);
    dpb(i)=x(4);
    dpc(i)=x(5);
end

% plot der Größen in zwei Subplots - in zwei Zeilen, in eine Spalte
row = 2; col = 1;
subplot(row,col,1)
plot(s,om1,s,om2,'linewidth',1.5);
grid on
title('Winkelgeschwindigkeiten');
xlabel('');
ylabel('omega [rad/s]');
legend('om1','om2');

subplot(row,col,2)
plot(s,dpa,s,dpb,s,dpc,'linewidth',1.5);
grid on
title('Kraftstoesse');
xlabel('s [m]');
ylabel('dp [Ns]');
legend('dpa','dpb','dpc');
```

```

function x=fmat33(s)
% feste Parameter sind global definiert

global M a b

%      Om1      Om2      pa      pb      pc
A = [   M*a/2      0      1      -1      0
        M*a      M*a/2      0      1      -1
        M*a      M*s      0      0      1
        M*a*a/12      0     -a/2     -a/2      0
          0   M*a*a/12      0     -a/2   -(s-a/2)  ];
x=A\b;

% Gehört zur Aufgabe A_3_3_Kugel_Stoss.m
end
  
```

