Aufgabe 25.5

Lösen Sie die folgenden Aufgaben mit `MATLAB`. Protokollieren Sie Ihr Vorgehen in einer `diary`-Datei und speichern Sie erstellte Plots ab.

1. Es sei $A = \begin{pmatrix} 1 & 3 & 2 \\ 1 & -1 & 2 \\ 1 & 2 & -1 \end{pmatrix}$ (vgl. Aufgabe 6.173). Bestimmen Sie die inverse Matrix von A.
2. Bestimmen Sie die Determinante der Matrix aus Aufgabe Aufgabe 6.185 für
   a) $a = 1, b = 1, c = 1, d = 1$
   b) $a = 5, b = 4, c = 3, d = 2$
   c) $a = -2, b = 0, c = 2, d = 4$
   d) $a = 6, b = 4, c = 4, d = 6.$
3. Zeichnen Sie die Kanten des von den drei Vektoren $\vec{a} = \begin{pmatrix} 1 \\ 0 \\ -3 \end{pmatrix}, \vec{b} = \begin{pmatrix} 2 \\ 1 \\ -1 \end{pmatrix}$ und $\vec{a} \times \vec{b}$ aufgespannten Parallelepipedes. Versuchen Sie, die Ansicht so zu wählen, dass die Orthogonalität von $\vec{a}$ zu $\vec{a} \times \vec{b}$ und von $\vec{b}$ zu $\vec{a} \times \vec{b}$ zu erkennen ist.
4. a) Stellen Sie die Ebene aus Aufgabe 7.105, die drei gegebenen Punkte A, B, C sowie das von diesen erzeugte Dreieck in einem Plot geeignet dar.
   b) Projizieren Sie die Vektoren $\overrightarrow{AC}$ und $\overrightarrow{BC}$ in die Ebene (d.h., berechenen Sie die zum Normalenvektor senkrechte Komponente). Zeichnen Sie die projizierten Vektoren in die Ebene ein (angesetzt am Punkt A bzw. B).
   c) Berechnen Sie den Lotfußpunkt von C bzgl. der gegebenen Ebene. Zeichnen Sie den Lotfußpunkt und die Strecke vom Punkt zum Lotfußpunkt. Berechnen Sie auch den Abstand des Punktes zur Ebene.
   d) Bestimmen Sie den in Aufgabe 7.105 gefragten Flächeninhalt.

Öffnen Sie die erstellte `diary`-Datei (vorher mit $\gg$ `diary off` die Protokollierung abschließen) und entfernen Sie ggf. überflüssige Zeilen (z.B. Fehleingaben). Drucken Sie anschließend die bearbeitete `diary`-Datei und eventuell angefertigte Plots möglichst sparsam (d.h. nach Möglichkeit duplex, mehrere Seiten pro Blatt, kleine Schriftgröße) aus.

Hinweise zur MATLABaufgabe

Inverse Matrix, Determinante und Kreuzprodukt

In MATLAB kann die inverse Matrix mit dem Befehl `inv` und die Determinante mit dem Befehl `det` berechnet werden. Beispiel:

```
≫ A=[1, -1, 2; -4, 2, 0; 1, 0, 3]
≫ inv(A)
≫ det(A)
```

Für die Überprüfung von händisch ausgerechneten inversen Matrizen mit MATLAB ist es oft günstig, zusätzlich

```
≫ det(A)*inv(A)
```

zu berechnen.

Das Kreuzprodukt zweier Vektoren lässt sich mit dem Befehl `cross` und das Skalarprodukt mit dem Befehl `dot` bestimmen. Beispiel:

```
≫ cross([1; -1; 2], [-4; 2; 0])
≫ dot([1; -1; 2], [-4; 2; 0])
```

Darstellen von Punkten

Ein einzelner Punkt im Raum kann durch einen einfachen `plot3`-Befehl dargestellt werden. Beispielsweise wird durch

```
≫ plot3(1,1,1)
```

der Punkt mit den Koordinaten $\begin{pmatrix}1\\1\\1\end{pmatrix}$ gezeichnet. Die Darstellung lässt sich durch

```
≫ plot3(1,1,1,'Marker','.','MarkerSize',20,'Color','red')
≫ grid on
```

verdeutlichen.

Lösung:
nachbereitete `diary`-Datei (Kommentare durch % gekennzeichnet) und Plots auf dieser und den nächsten Seiten

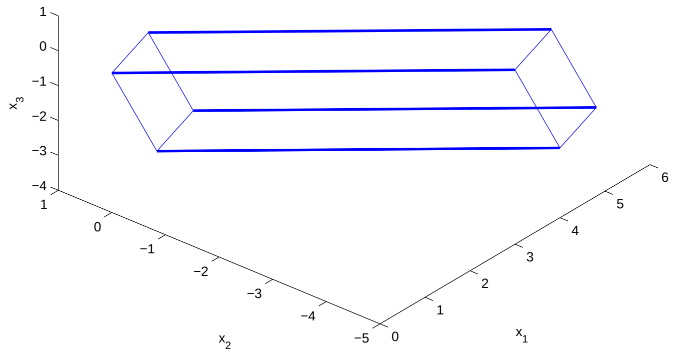
```
% ---------------------------------------------------------------------------
% Aufgabe 1
% ---------------------------------------------------------------------------

% Matrix aufstellen
A=[1 3 2; 1 -1 2; 1 2 -1]
A =
     1     3     2
     1    -1     2
     1     2    -1

% Matrix invertieren
inv(A)
ans =
   -0.2500    0.5833    0.6667
    0.2500   -0.2500         0
    0.2500    0.0833   -0.3333

% Vergleich mit dem von Hand errechneten Ergebnis
det(A)
ans =
    12

det(A)*inv(A)
ans =
    -3     7     8
     3    -3     0
     3     1    -4


% ---------------------------------------------------------------------------
% Aufgabe 2
% ---------------------------------------------------------------------------

a=[1, 5, -2, 6];
b=[1, 4, 0, 4];
c=[1, 3, 2, 4];
d=[1, 2, 4, 6];

for i=1:4
    % Matrix aufstellen
    B=[ 5   0   0    1    5    4;
        7   3  a(i) b(i) c(i) d(i);
        0   0   0    2    4    0;
        5   0  b(i)  1    2   c(i);
        0   0   0    3   d(i)  0;
       a(i) 0   0    1    5    4]
    % Determinante berechnen
    Determinante=det(B)
end;

B =
     5     0     0     1     5     4
     7     3     1     1     1     1
     0     0     0     2     4     0
     5     0     1     1     2     1
     0     0     0     3     1     0
     1     0     0     1     5     4
Determinante =
   480

B =
     5     0     0     1     5     4
     7     3     5     4     3     2
     0     0     0     2     4     0
     5     0     4     1     2     3
     0     0     0     3     2     0
     5     0     0     1     5     4
Determinante =
     0

B =
     5     0     0     1     5     4
     7     3    -2     0     2     4
     0     0     0     2     4     0
     5     0     0     1     2     2
     0     0     0     3     4     0
    -2     0     0     1     5     4
Determinante =
     0

B =
     5     0     0     1     5     4
     7     3     6     4     4     6
     0     0     0     2     4     0
     5     0     4     1     2     4
     0     0     0     3     6     0
     6     0     0     1     5     4
Determinante =
     0


% ---------------------------------------------------------------------------
% Aufgabe 3
% ---------------------------------------------------------------------------

% Vektoren aufstellen

a=[1; 0; -3]
a =
     1
     0
    -3
b=[2; 1; -1]

b =
     2
     1
    -1

% Kreuzprodukt berechnen
c=cross(a,b)
c =
     3
    -5
     1

% Vektoren zeichnen
figure;
hold on;
```

```
% Mit p=..; q=..; plot3( [p(1), q(1)] , [p(2), q(2)] , [p(3), q(3)] )
% wird die Verbindungsstrecke vom Punkt p zum Punkt q gezeichnet.
% Dies wird jetzt für alle 12 Kanten ausgeführt:

% "unteres" (i=0) und "oberes" (i=1) Parallelogramm
for i=0:1
    p=i*c; q=i*c+a; plot3( [p(1), q(1)] , [p(2), q(2)] , [p(3), q(3)] )
    p=i*c+a; q=i*c+a+b; plot3( [p(1), q(1)] , [p(2), q(2)] , [p(3), q(3)] )
    p=i*c+a+b; q=i*c+b; plot3( [p(1), q(1)] , [p(2), q(2)] , [p(3), q(3)] )
    p=i*c+b; q=i*c; plot3( [p(1), q(1)] , [p(2), q(2)] , [p(3), q(3)] )
end;

% senkrechte Verbindung zwischen "unterem" und "oberem" Parallelogramm
% (zur Verdeutlichung fett gezeichnet)
O=[0 0 0];
p=O; q=c; plot3( [p(1),q(1)],[p(2),q(2)],[p(3),q(3)], 'LineWidth',2 )
p=a; q=c+a; plot3( [p(1),q(1)],[p(2),q(2)],[p(3),q(3)], 'LineWidth',2 )
p=a+b; q=c+a+b; plot3( [p(1),q(1)],[p(2),q(2)],[p(3),q(3)], 'LineWidth',2 )
p=b; q=c+b; plot3( [p(1),q(1)],[p(2),q(2)],[p(3),q(3)], 'LineWidth',2 )

% Achsen beschriften
xlabel('x_1')
ylabel('x_2')
zlabel('x_3')

% ggf. von Hand in gewünschte Lage drehen,
% mit der folgenden Anweisung wird Betrachterstandpunkt vorgegeben:
view(-50,50)

% Titel setzen
viewparam=get(gca(),'view');
title(['Darstellung des von den Vektoren a, b und a x b aufgespannten
Paralellepipedes mit view(',num2str(viewparam(1)),',',num2str(viewparam(2)),
')']);

print -depsc ak5_zusatz_3.eps
```

```
% -------------------------------------------------------------------------
% Aufgabe 4
% -------------------------------------------------------------------------

% a) ----------------------------------------------------------------------

% Punkte eingeben
A=[2; 0; 0]
A =
     2
     0
     0
B=[0; 3; 0]
B =
     0
     3
     0
C=[24; 16; 14]
C =
    24
    16
    14

% Normalenvektor der Ebene eingeben
n=[3; 2; 1]
n =
     3
     2
     1

figure; hold on;
% Zeichne Ebene 3x+2y+z=6 durch Punkte [6 -3 -6], [-2 9 -6] und [-2 -3 18]
patch([6 -2 -2],[-3 9 -3],[-6 -6 18],[1 0 0],'FaceAlpha',0.5);
%
% Dieser Befehl funktioniert unter Octave-3.0.1 mit Jhandle nicht korrekt.
% Statt dessen sollte dort der Befehl
%
% patch([6 -2 -2],[-3 9 -3],[-6 -6 18], 'cdata', reshape([1 0 0],1,1,3),
% 'FaceColor', 'flat', 'FaceAlpha',0.5);
%
% verwendet werden.
%

% Achsen beschriften
xlabel('x')
ylabel('y')
zlabel('z')

% Zeichne Punkte A, B, C ein
plot3(A(1),A(2),A(3),'Marker','.','MarkerSize',20,'Color','blue')
plot3(B(1),B(2),B(3),'Marker','.','MarkerSize',20,'Color','blue')
plot3(C(1),C(2),C(3),'Marker','.','MarkerSize',20,'Color','blue')

% Stelle das Dreieck durch Zeichnen der Verbindungsstrecken dar
p=A; q=B;
plot3([p(1),q(1)], [p(2),q(2)], [p(3),q(3)], 'LineWidth',2,'Color','blue')
p=A; q=C;
plot3([p(1),q(1)], [p(2),q(2)], [p(3),q(3)], 'LineWidth',2,'Color','blue')
p=C; q=B;
plot3([p(1),q(1)], [p(2),q(2)], [p(3),q(3)], 'LineWidth',2,'Color','blue')
```

```
% b) ----------------------------------------------------------------------

% Projiziere AC in die Ebene (verwende Formel aus der Vorlesung)
Proj_AC = 1/norm(n)^2*cross(n,cross(C-A,n))
Proj_AC =
    -2
     0
     6

% Zeichne Projektion in die Ebene ein
p=A; q=A+Proj_AC;
plot3([p(1),q(1)], [p(2),q(2)], [p(3),q(3)], 'LineWidth',2,'Color','green')

% Projiziere BC in die Ebene (verwende Formel aus der Vorlesung)
Proj_BC = 1/norm(n)^2*cross(n,cross(C-B,n))
Proj_BC =
     0
    -3
     6

% Zeichne Projektion in die Ebene ein
p=B; q=B+Proj_BC;
plot3([p(1),q(1)], [p(2),q(2)], [p(3),q(3)], 'LineWidth',2,'Color','green')

% c) ----------------------------------------------------------------------

% Berechne den  Lotfußpunkt
PC=C+1/norm(n)^2*dot(A-C,n)*n;
% (Es gilt PC = A+Proj_AC = B+Proj_BC .)

% Zeichne Lotfußpunkt PC ein
plot3(PC(1),PC(2),PC(3),'Color','g',
      'Marker','.','MarkerSize',20,'Color','red')

% Zeichne Verbindungsstrecke
p=C; q=PC;
plot3([p(1),q(1)], [p(2),q(2)], [p(3),q(3)], 'LineWidth',2,'Color','red')

% Berechne Abstand
norm(C-PC)
ans =
   29.9333

% ggf. von Hand in gewünschte Lage drehen,
% mit der folgenden Anweisung wird Betrachterstandpunkt vorgegeben:
view(-51.3,56)

% Titel setzen
viewparam=get(gca(),'view');
title(['Projektion des blauen auf das grüne Dreieck, Darstellung mit view(',
num2str(viewparam(1)),',',num2str(viewparam(2)),')']);

print -depsc ak5_zusatz_4.eps

% d) ----------------------------------------------------------------------

% Berechne Flächeninhalt des projizierten Dreiecks
Flaecheninhalt=1/2*norm(cross(A-PC, B-PC))
Flaecheninhalt =
   11.2250
```

Darstellung des von den Vektoren a, b und a x b aufgespannten Paralellepipedes mit view(−50,50)



Projektion des blauen auf das grüne Dreieck, Darstellung mit view(-51.3,56)