

Aufgabe 25.1

Arbeiten Sie sich (z.B. mithilfe der beigefügten Einführung) in das Computernumeriksystem MATLAB ein und lösen Sie damit dann die folgenden Aufgaben. Protokollieren Sie Ihr Vorgehen in einer `diary`-Datei und speichern Sie erstellte Plots ab.

1. Berechnen Sie $\sin \frac{\pi}{6}$, $\sqrt[3]{42}$ (also $42^{\frac{1}{3}}$) und e^π (also $\exp(\pi)$).
2. Plotten Sie die Funktion $f(x) = -x^3 + 3x^2 + x - 2$ im Intervall $[-1.2, 3.4]$. Geben Sie dem Plot einen Titel und beschriften Sie auch die Achsen entsprechend.
3. Bestimmen Sie unter Verwendung des Befehls `roots` (siehe \gg `help roots`) die Nullstellen des Polynoms $f(x) = -x^3 + 3x^2 + x - 2$. Vergleichen Sie mit dem Plot von oben.
Hinweis: \gg `roots([1, -2, -3])` bestimmt die Nullstellen von $f(x) = x^2 - 2x - 3$.
4. Plotten Sie die Funktion $g(x) = 2\sin(5x)$ in dasselbe Fenster wie $f(x)$. Verwenden Sie dabei eine andere Farbe. Passen Sie den Titel an und erstellen Sie eine Legende (\gg `help legend`).

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.

Einführung in MATLAB (Programmstand 2008)

MATLAB ist eine kommerzielle Software zur Lösung mathematischer Probleme. Sämtliche Beispiele dieser Einführung mit Ausnahme des Befehls `ezplot` funktionieren genauso mit der freien Software **Octave**. Windows-Downloadversionen dafür gibt es unter octave.sourceforge.net. Hier wurde die Version `octave-3.0.1-vs2008-setup.exe` verwendet.

Starten Sie MATLAB und geben Sie die folgenden Befehle ein. Beobachten Sie sorgfältig die Bildschirmausgabe und achten Sie auf die Bedeutung von `,` und `;`.

Tipp: mit den Cursortasten kann man die vorherigen Eingabezeilen „zurückholen“ und ändern.

```
>> 23+19
>> 3*4
>> 3*4,
>> 3*4;
>> a=3*4;
>> a
>> a;
>> sqrt(2)
>> help sqrt
>> pi
>> help pi
>> radius=2, durchmesser=2*radius, umfang=durchmesser*pi
>> radius=2, durchmesser=2*radius; umfang=durchmesser*pi
>> radius=2, durchmesser=2*radius umfang=durchmesser*pi
```

Mit dem Befehl

```
>> pwd
```

(`pwd` steht für „**p**rint **w**orking **d**irectory“) können Sie sich Ihr aktuelles Arbeitsverzeichnis anzeigen lassen. Es empfiehlt sich, für jeden Hausaufgabenkomplex ein eigenes Verzeichnis anzulegen. Dies kann beispielsweise durch die folgenden beiden Befehle realisiert werden:

```
>> mkdir Hausaufgabenkomplex1
>> cd Hausaufgabenkomplex1
>> pwd
```

Geben Sie nun die folgenden Befehle ein:

```
>> help diary
>> diary hal.txt
```

Alle ab jetzt eingegebenen Befehle werden mitsamt ihrer Ausgabe in die Datei `hal.txt` im aktuellen Arbeitsverzeichnis geschrieben. Dies eignet sich besonders, um Ihre Lösung der Hausaufgabe zu dokumentieren.

Ein Plot der Funktion $f(x) = x^3 - 2x^2 + 1$ auf dem Intervall $[-1, 2]$ kann wie folgt angefertigt werden. Beobachten Sie den Einfluss des Parameters n .

```
>> n=5
>> help linspace
>> x=linspace(-1,2,n)
>> F=x.^3-2*x.^2+1
```

Die komponentenweise Ausführung von Operationen für Vektoren, wie sie z.B. hier zur Berechnung von Werten einer Funktion an mehreren Stellen benötigt wird, wird in MATLAB durch einen Punkt vor dem Operationszeichen veranlasst. $[3 \ 7] \cdot [5 \ 8]$ ergibt also z.B. $[15 \ 56]$. Dabei handelt es sich um keine der bekannten Produktbildungen aus der Vektorrechnung.

```
>> plot(x,F);
>> title('Plot von f(x)=x^3-2*x^2+1 mit n=5');
>> xlabel('x'); ylabel('f(x)'); grid on;
>> n=100
>> x=linspace(-1,2,n)
>> F=x.^3-2*x.^2+1
>> help hold
>> hold on
>> plot(x,F,'o-r');
>> title('Plot von f(x)=x^3-2*x^2+1 mit n=5 und n=100');
>> print -dpng hal.png
>> print -depsc hal.eps
```

Im Gegensatz zu der verpixelten png-Grafik wurde mit dem letzten Befehl eine Vektorgrafik erzeugt. Allerdings wird für deren Anzeige ein Postscript-Interpreter (z.B. [Ghostscript](#)) benötigt. Dieser ist auf den URZ-Rechnern installiert.

```
>> hold off
>> plot(x,F,'-.k');
>> clf
```

Einfache Plots lassen sich auch mit der Funktion `ezplot` (sprich: „easy plot“) erstellen:

```
>> help ezplot
>> ezplot('x.^3-2*x.^2+1',[-1,2])
```

Auch Summen lassen sich leicht berechnen:

```
>> x=[-2, 2, 3, 1, 0, 4]
>> sum(x)
>> x+2
>> sum(x+2)
```

Die folgenden Befehlen müssen nicht vollständig verstanden werden, sie sollen lediglich aufzeigen, dass mit wenig Aufwand komplizierte Funktionen in eine ansprechende Darstellung gebracht werden können.

```
>> x=-8:0.5:8; y=x;
>> [X,Y]=meshgrid(x,y);
>> R=sqrt(X.^2+Y.^2)+eps;
>> Z=sin(R)./R;
>> mesh(Z);
>> title('Mesh-Plot von sin(sqrt(x^2+y^2))/sqrt(x^2+y^2)')
>> surf(Z);
>> title('Surf-Plot von sin(sqrt(x^2+y^2))/sqrt(x^2+y^2)')
```

Um das Protokollieren in die Datei `hal.txt` zu beenden, verwenden Sie den folgenden Befehl.

```
>> diary off
```

Schauen Sie sich den Inhalt der Datei `hal.txt` an, um die Funktionsweise des `diary`-Befehls genau zu verstehen.

Lösung:

Ausdruck siehe nächste Seite

Die bei Aufgabe 3 ermittelten Nullstellen 3.1149, -0.8608 und 0.7459 sind tatsächlich die der blauen Kurven in den Plots.

```

% -----
% Aufgabe 1
% -----
sin(pi/6)

ans =

    0.5000

42^(1/3)

ans =

    3.4760

exp(pi)

ans =

    23.1407

% -----
% Aufgabe 2
% -----
n=100;
x=linspace(-1.2,3.4,n);
F=-x.^3+3*x.^2+x-2;
plot(x,F);
title('Plot von f(x)=-x^3+3*x^2+x-2');
xlabel('x'); ylabel('f(x)'); grid on;
print -depsc akl_zusatz_2.eps

% -----
% Aufgabe 3
% -----
roots([-1,3,1,-2])

ans =

    3.1149
   -0.8608
    0.7459

% -----
% Aufgabe 4
% -----
hold on;
G=2*sin(5*x);
plot(x,G,'r');
title('Plot von f(x) und g(x)');
ylabel('y');
legend('f(x)=-x^3+3*x^2+x-2','g(x)=2*sin(5*x)');
print -depsc akl_zusatz_4.eps
diary off

```

