

L^AT_EX-Beamer-Kurs

Grafiken

Martin Glatz
ig-mathe



1. Pakete

2. Farben

3. Boxen

4. Grafiken

5. Sonstiges

6. Zeichnen



Leseanleitung

Programmcode:

```
abgesetzter Programmcode  
(Leerzeichen notfalls entfernen)  
%Kommentare im Code
```

Normaler Text sowie Programmcode im Text normaler Text

Achtung! Gefahr?

Besonders Wichtiges oder Gefährliches ist so gekennzeichnet! Also eventuelle Unstimmigkeiten, Kompatibilitätsprobleme, ...

Solche Hinterlegungen dienen nur der optischen/inhaltlichen Strukturierung und haben ansonsten keine besondere Bedeutung.

Starke Gliederung

Falls notwendig und sinnvoll, gibt es auch Gliederungen mit Überschriften ...



IATeX-Kurs
Grafiken

Inhalt

Pakete

Farben

Boxen

Grafiken

Sonstiges

Zeichnen

1. Pakete

2. Farben

3. Boxen

4. Grafiken

5. Sonstiges

6. Zeichnen



Pakete für Farben, Grafiken

Farben

- ⊙ `xcolor` Verwendung/Neudefinition von Farben

Grafiken einbinden

- ⊙ `graphicx` Bilder und Grafiken einbinden
- ⊙ `pdfpages` ganze pdf-Seiten einbinden
- ⊙ `wallpaper` Hintergrundgrafiken auf einer Seite
- ⊙ `eso-pic` Hintergrundgrafiken mit vielen Einstellungen

Grafiken zeichnen/überdrucken/beschriften

- ⊙ `overpic` Text/Objekte über Bilder drucken
- ⊙ `tikz` Vektorgrafiken in \LaTeX »programmieren«
- ⊙ `Asymptote` Rechnen, Programmieren, Zeichnen



Pakete (Fortsetzung)

Grafiken anordnen

- ⊙ `wrapfig` Bilder von Text umfließen lassen (float)
- ⊙ `picins` Bilder von Text umfließen lassen (fix)
- ⊙ `rotating` zum Drehen von bel. Objekten oder Tabellen

Abbildungen etc betiteln

- ⊙ `float` für eigene float-Umgebungen (inkl. räuml. Anordnung)
- ⊙ `subfig` für Aufteilung beschrifteter (Unter-)Grafiken
- ⊙ `caption` mehr Einstellungen für Beschriftungen von floats
- ⊙ `subcaption` Einstellungen von Unterbeschriftungen



Farben ermöglichen

```
\usepackage[table]{xcolor}
```

- ⊙ für farbige Linien, Text, Boxen, Tabellen, ...
- ⊙ verschiedene Farbmodelle/Farbbezeichnungen werden unterstützt: cmyk, RGB, Namensdefinierte (x11names)
- ⊙ Vorauswahl möglich:
z. B. `\usepackage[x11names]{xcolor}`
- ⊙ für Druckerzeugnisse cmyk sinnvoll/notwendig:
`\usepackage[cmyk]{xcolor}`
(Farben werden passend konvertiert)



vordefinierte Farben

Standard

- ⦿ black, darkgray, gray, lightgray, white
- ⦿ blue, cyan, teal, green, red, orange, yellow
- ⦿ brown, magenta, pink

mit `x11names`-Option (Beispiele):

- ⦿ Gold1, Gold2, Gold3, Gold4
- ⦿ DodgerBlue1, DodgerBlue2, DodgerBlue3, DodgerBlue4
- ⦿ Red1, Red2, Red3, Red4
- ⦿ ...

Siehe Dokumentation von `xcolor`-Paket



eigene Farben definieren

Eigene Farbdefinitionen:

```
\definecolor{Meinpink}{rgb}{1.0,0.5,0.5}  
\definecolor{Meinpink}{RGB}{1.0,0.5,0.5}  
\definecolor{Meinpink}{cmyk}{1.0,0.5,0.5}  
\colorlet{Meinrot}{red}
```

- ⦿ `\definecolor{Meinpink}{rgb}{1.0,0.5,0.5}`
liefert diese Farbe
- ⦿ `\colorlet{neuefarbe}{farbe}`
weist der Farbe neuefarbe den Farbton farbe zu.

Farben (Sättigung)

```
\colorlet{Hellrot}{red!50}  
\colorlet{Dunkelrot}{red!150}
```

Farben mischen

```
\colorlet{gruengelb}{green!40!yellow}  
%40 % gruen, 60 % gelb
```



Farben verwenden

Text:

- ⊙ `\textcolor{SteelBlue3}{In Blau}` In Blau
- ⊙ oder als Schalter `\color{SteelBlue3}` bis zum nächsten Farbschalter

Seitenhintergrund:

```
\pagecolor{yellow}
```

Linien etc

```
{\color{red}\rule{1cm}{2cm}}
```



Boxen und Farbe

- `\colorbox{Aquamarine2}{Grüne Box}` liefert Grüne Box
- `\fbox{Umrandete Box}` Umrandete Box
- `\fcolorbox{gray}{Gold1}{...}`
Farbige Boxen mit Randlinie

Rahmen ändern:

- `\setlength{\fboxrule}{5pt}` 5pt Randlinienstärke
- `\setlength{\fboxsep}{4pt}` 4pt Randabstand
- diese Befehle einfach vor den gewünschten Boxen anführen.
- Achtung: an sich sind das globale Änderungen!
Nach Bedarf mit `{...}` klammern!

Farbige Boxen mit Randlinie



Minipages

- Die Minipage-Umgebung erzeugt eine kleine Seite mit vorgegebener Breite.
- Im Text wird sie grundsätzlich wie ein einziger, großer Buchstabe behandelt, der vertikal zentriert ist:

Text vorher ... eine kleine Seite mit
automatischer Wort-
trennung Text nachher

- (Der Rahmen dient nur der Illustration. Die Umgebung müsste z. B. von `\fbox{...}` eingeschlossen werden)

```
\begin{minipage}{0.3\textwidth}  
eine kleine Seite ...  
\end{minipage}
```

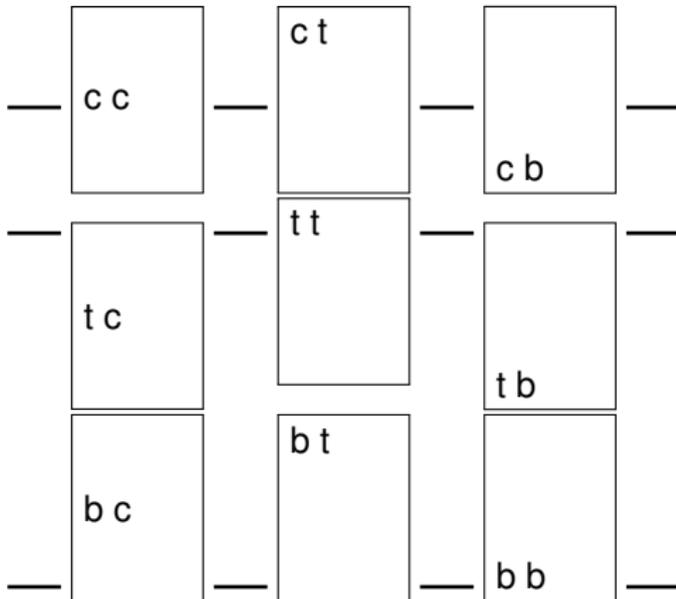
- Achtung: Der Befehl `\textwidth` innerhalb einer Minipage bezieht sich immer auf die Breite der Minipage, nicht auf die Textbreite der großen Seite.



Minipages – Ausrichtungen II

```
\begin{minipage}[x][1.5cm][y]{1cm}  
...  
\end{minipage}
```

(in den Feldern stehen immer die Ausrichtungen x und y)





Farbgestaltung – Gesamtbeispiel

Und so macht man auch in dieser Präsentation farbige Boxen für verschiedenste Formeln wie

$$f(x) = a_3x^3 + a_2x^2 + a_1x + a_0$$

oder Grafiken wie



(händisch) gemacht werden.

```
\setlength{\fboxrule}{2pt}
\setlength{\fboxsep}{4pt}

\fcolorbox{RoyalBlue2}{LightSteelBlue1}{
\begin{minipage}{0.75\textwidth}
Und so macht man auch ...
\end{minipage}}
```



Anwendungen – Minipage

- ⊙ zwei Objekte/Grafiken an der Oberkante auszurichten
- ⊙ (Die waagrechten Linien symbolisieren die Mitte der Grundlinie)
- ⊙ (Der Rahmen dient wieder der Verdeutlichung)



```
\begin{minipage}[c][1.5cm][t]{0.4\textwidth}  
... Inhalt links ...  
\end{minipage}  
\hfill %Ausrichtung der Boxen am Seitenrand  
\begin{minipage}[c][1.5cm][t]{0.4\textwidth}  
... Inhalt rechts ...  
\end{minipage}
```



Grundlegendes zu Grafiken

Unterstützte Grafik-Formate

- ⊙ Pixel-Grafiken: jpg, bmp, png, ...
- ⊙ Vektorgrafiken: svg, eps, pdf (z. T.), ...
- ⊙ Verlustfreie Grafiken: bmp, png sowie Vektorgrafiken
- ⊙ Verlustbehaftete Grafiken: jpg und ähnliche (Komprimierung)

Standard-Paket: `\usepackage{graphicx}`

Einbindbare Formate (pdf_LA_TE_X): pdf, jpg, png

- ⊙ Vektorgrafiken nützen: beliebig scharf, meist recht klein.
- ⊙ nützliches, frei verfügbares Programm zum Bearbeiten/Erstellen von Vektorgrafiken oder pdf-Dateien: **Inkscape**:
<http://inkscape.org/?lang=de>
- ⊙ Je nach Bedarf: Word/Office-Dokumente als pdf exportieren und als Grafik in $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ einbinden. (z. B. komplizierte Tabellen)



graphicx und includegraphics

```
\includegraphics [Optionen] {./pfad/grafikename}
```

Optionen:

- ⊙ `scale= Faktor` Skalierung um einen Faktor
(z. B. 0.5 Verkleinerung auf die Hälfte)
- ⊙ `height = h` Höhe wird vorgegeben
- ⊙ `width = b` Breite wird vorgegeben
- ⊙ `keepaspectratio = true oder false` (un)verzerrt
- ⊙ `angle = winkel` (Drehung um math. pos. Sinn)
- ⊙ `trim=l b r t, clip` zuschneiden
links, unten, rechts, oben (l,b,r,t Längenangaben)
clip zum Wegschneiden

Grafiken können auch in Tabellen, Kopf- und Fußzeilen, minipages,
... eingebaut werden



Grafik-Einbindung – Beispiele

```
\includegraphics[width= 20mm, angle=30]  
{grafiken/maskottchen.png}
```



```
\includegraphics[trim = 5cm 4cm 0 0,clip,  
height=2cm]{grafiken/maskottchen.png}
```



(Zum Zuschneiden: trial & error oder mit passendem Programm die Koordinaten messen)



Floats und caption

- Gleitobjekte/-umgebung: Objekte/Grafiken werden dort eingefügt, wo die Absätze es zulassen
- auf Position kann nicht exakt zugegriffen werden
- Wünsche können L^AT_EX mitgeteilt werden: `h`, `t`, `b`, `p` (hier, Seitenanfang, Seitenende, eigene Bilderseite)
- voreingestellte float-Umgebungen:
`figure` (für Abbildungen) und `table` (für Tabellen)
- weitere Gleitobjektdefinitionen möglich (z. B. für Diagramme etc) (Paket `float`)
- Betitelung erfolgt mit `\caption{Titel}` in der Umgebung
- labeling muss nach dem `\caption`-Befehl kommen:
`\label{marke}`
- Zugriff auf `marke` wie üblich: `\ref{marke}` etc.



figure-Umgebung

Grafiken als Abbildung x: »Bildtitel« mit fortlaufender Nummerierung einfügen:

```
\begin{figure}[htb]
  \centering
  \includegraphics[...]{...}
  \caption{Bildtitel}
  \label{fig:bildreferenz}
\end{figure}
```

Zusätzliche Befehle:

- ⊙ Abbildungsverzeichnis: `\listoffigures`
- ⊙ Grafikename (»Abbildung«) umbenennen:
`\renewcommand{\figurename}{Abb.}` bzw.
`\renewcaptionname{ngerman}{\figurename}{Abb.}`



L^AT_EX-Kurs
Grafiken

Inhalt

Pakete

Farben

Boxen

Grafiken

einbinden

figure-Umgebung

nebeneinander

textumflossen

Sonstiges

Zeichnen

Grafiken nebeneinander I

ohne gemeinsame Nummerierung, an Grundlinie ausgerichtet



Abbildung: Grafik links



Abbildung: Grafik rechts

```
\begin{figure}[htb]
\begin{minipage}[b] [3cm] [b]{0.4\textwidth}
...
\caption{Grafik links}\label{links}
\end{minipage}
\hfill
\begin{minipage}[b] [3cm] [b]{0.4\textwidth}
...
\caption{Grafik rechts}\label{rechts}
\end{minipage}
```



Grafiken nebeneinander I

ohne gemeinsame Nummerierung, gemeinsame Oberkante

Abbildung: Grafik links



Abbildung: Grafik rechts



```
\begin{figure}[htb]
\begin{minipage}[t][3cm][t]{0.4\textwidth}
\caption{Grafik links}\label{links}
...
\end{minipage}
\hfill
\begin{minipage}[t][3cm][t]{0.4\textwidth}
\caption{Grafik rechts}\label{rechts}
...
\end{minipage}
```



L^AT_EX-Kurs
Grafiken

Inhalt

Pakete

Farben

Boxen

Grafiken

einbinden

figure-Umgebung

nebeneinander

textumflossen

Sonstiges

Zeichnen

Grafiken nebeneinander II

mit gemeinsamer Nummerierung



(a) Grafik links



(b) Grafik rechts

Abbildung: Zwei Grafiken

```
\begin{figure}
\begin{minipage}[b]{.5\linewidth}
...
\subcaption{Grafik links}\label{abb:1a}
\end{minipage}%
\begin{minipage}[b]{.5\linewidth}
...
\subcaption{Grafik rechts}\label{abb:1b}
\end{minipage}
\caption{Zwei Grafiken}\label{abb:1}
\end{figure}
```



Textumflossene Grafiken // wrapfig

- ⊙ `\usepackage{wrapfig}` laden
- ⊙ Grafiken können neben Text eingebaut werden
- ⊙ captions sind möglich
- ⊙ z. T. große Abstände (manuell mit `\vspace{...}` korrigieren)
- ⊙ breite Breite des Platzes, der für die Grafik reserviert wird
- ⊙ ausrichtung
 - l am linken Seitenrand, L zusätzlich float
 - r am rechten Seitenrand, R zusätzlich float
- ⊙ zeilenanz (optional): Anzahl der Zeilen, die für die Grafik reserviert werden

```
\begin{wrapfigure}[zeilenanz]{ausrichtung}{breite}  
...  
\end{wrapfigure}
```

andere Pakete: picins oder floatflt



Gesamtbeispiel

Die `\vspace{...}`-Befehle machen kleinere Abstände

... letzter Textabsatz vorher.

```
\begin{wrapfigure}{r}{0.5\textwidth}
\vspace{-\baselineskip}
\begin{center}
\includegraphics[width=0.42\textwidth]{grafik}
\end{center}
\vspace{-0.5\baselineskip}
\caption{Bildunterschrift}
\vspace{-0.5\baselineskip}
\end{wrapfigure}
```

Mit diesem Text wird neben dem Bild begonnen ...



Ganze Seiten einbinden

```
%Paket einbinden  
\usepackage{pdfpages}
```

```
%an gewollter Stelle  
\includepdf [optionen] {pfad/pdfseite.pdf}
```

- ⊙ Paket ermöglicht das Einbinden von ganzen pdf-Seiten
- ⊙ Option `pages={2,3, {}, 5-7}` bindet die entsprechenden Seiten ein (`{}` bindet eine leere Seite ein)
- ⊙ `nup=nxm` druckt mehrere pdf-Seiten auf eine Dokument-Seite (`n` in x-Richtung, `m` in y-Richtung)



Hintergrundgrafiken // wallpaper

```
\usepackage{wallpaper}
```

Befehle:

```
\TileWallPaper{\paperwidth}{\paperheight}{bild}
```

```
\ClearWallPaper
```

```
\ThisTileWallPaper{\paperwidth}{\paperheight}{bild}
```

- 1 fügt das bild als Hintergrundgrafik ein
- 2 löscht die aktuelle Hintergrundgrafik
- 3 für eine Hintergrundgrafik nur auf einer Seite.

weitere Befehle vorhanden ...



Hintergrundgrafiken // eso-pic

```
\usepackage{eso-pic}
```

- ⊙ `\AddToShipoutPictureBG{...}` im Hintergrund (jede Seite)
- ⊙ `\AddToShipoutPictureBG*{...}` im Hintergrund (diese Seite)
- ⊙ `\AddToShipoutPictureFG{...}` im Vordergrund
- ⊙ `\ClearShipoutPictureBG` löscht die Bilder im Hintergrund
- ⊙ `\ClearShipoutPictureFG` löscht die Bilder im Vordergrund

Beispiel (Quadrat in rechter Ecke):

```
\AddToShipoutPictureBG{%  
\begin{minipage}[b][\paperheight][t]{\paperwidth}  
\hfill \rule{5cm}{5cm}  
\end{minipage}  
}
```



Grafiken nachträglich beschriften

- ⊙ Paketeinbindung (z. B.)

```
\usepackage[percent]{overpic}
```

- ⊙ stellt die Umgebung `overpic` bereit
- ⊙ mit `\put(x,y){...}` wird der gewünschte Inhalt an den (relativen) Koordinaten `x` und `y` eingebunden
- ⊙ bei `optionen` sind die üblichen `includegraphics`-Optionen möglich

Gesamtbeispiel:

```
\begin{overpic}[optionen]{bild.jpg}  
  \put(20,30){\textcolor{white}{etwas Text}}  
  \put(20,40){... $ f(x) = x^2$ ... }  
\end{overpic}
```



Grafiken in LaTeX »programmieren«

Vorteile:

- ⊙ Vektorgrafiken nach Maß
- ⊙ Quellcode direkt in tex-Dateien einbindbar
- ⊙ Für das perfekte, konsistente Ergebnis (Verwendung der selben Schriften in Text und Grafiken, ...)
- ⊙ manche Programme (z. B. GeoGebra) bieten Export in tikz-Format für L^AT_EX

Nachteile:

- ⊙ eigenständiges Programmieren mühsam
- ⊙ Jedes »Programm« hat eigene Sprache mit eigenen Befehlen
- ⊙ viel Einarbeitungszeit, wenn man selbst alles können möchte

Empfehlung

Meist einfacher: Mit externen Programmen Grafiken erzeugen und (z. B. als pdf) einbinden.



Grafiken selbst zeichnen: tikz und pgf

- `\usepackage{tikz}`
`\usepackage{pgf}`
`\usepackage{pgfplots}`

stellt die Befehle etc zur Verfügung (frontend)

- evtl. sogenannte Libraries nötig, z. B.

```
\usetikzlibrary{arrows,backgrounds,intersections}
```

- Vektorgrafiken auf höchstem Niveau
- Grafiken werden per Code programmiert (siehe unten)
- Befehle für Linien, Kreise, Kurven, Beschriftungen, Farben ...
- Einarbeitungszeit nötig :-)
- viele Tutorials vorhanden :-)

```
\begin{tikzpicture}  
  \fill[orange] (1ex,1ex) circle (1ex);  
\end{tikzpicture}
```

liefert ●



IATeX-Kurs
Grafiken

Inhalt

Pakete

Farben

Boxen

Grafiken

Sonstiges

Zeichnen

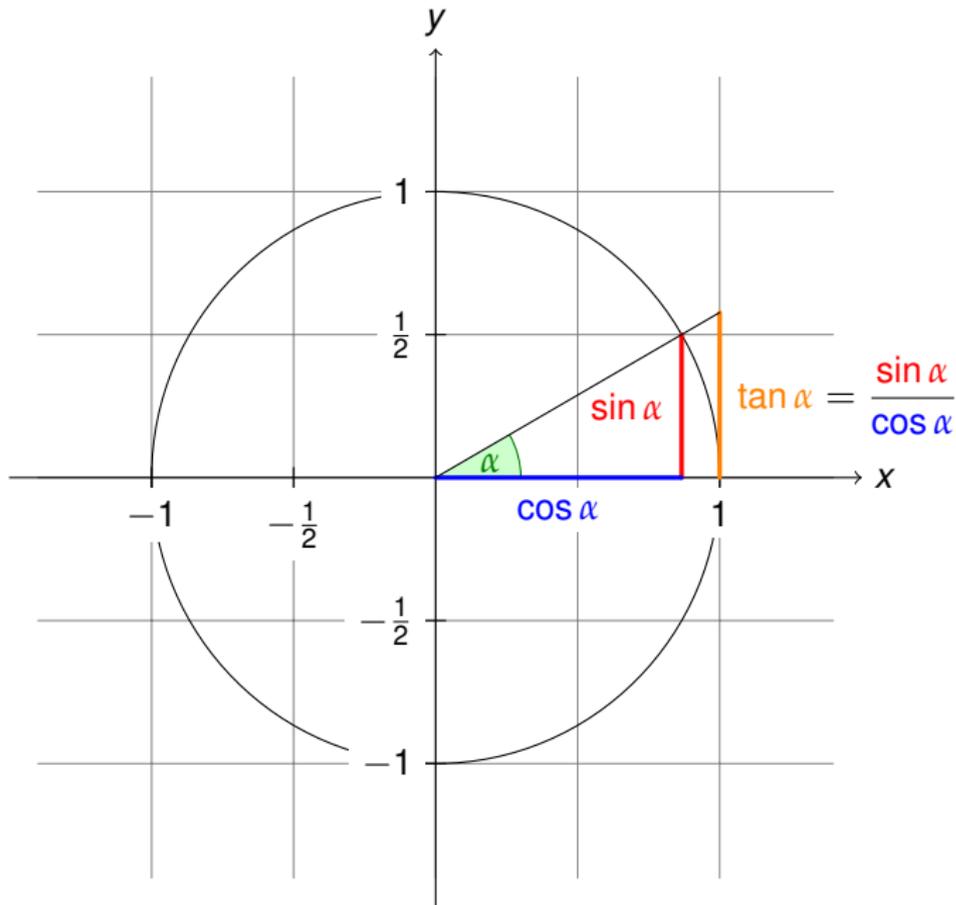
Tikz

Nutzung externer

Programme

fragile

Ein großes Beispiel mit tikz

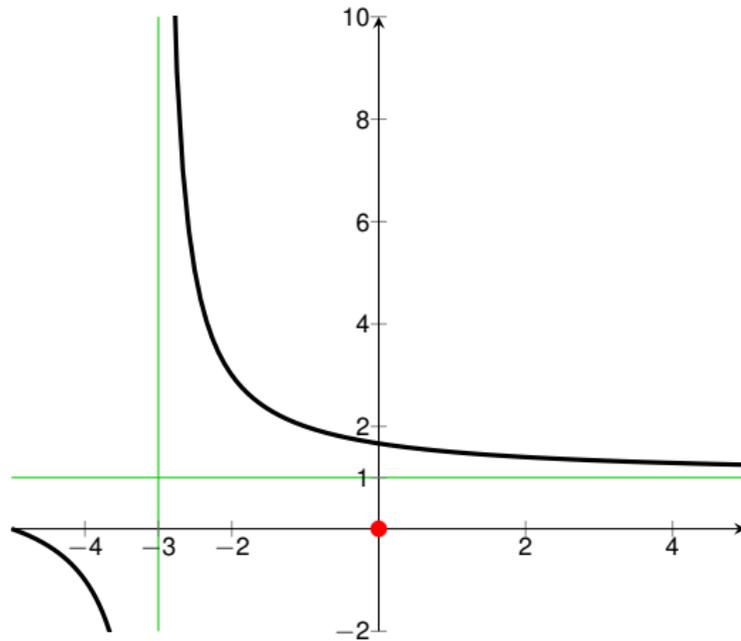




IATeX-Kurs
Grafiken

Inhalt
Pakete
Farben
Boxen
Grafiken
Sonstiges
Zeichnen

Tikz
Nutzung externer
Programme
fragile





```
\pgfplotsset{every tick label/.style=  
{inner sep=0pt,font=\scriptsize}}
```

```
\begin{tikzpicture}
```

```
\begin{axis}[
```

```
  ymin=-2,
```

```
  ymax=10,
```

```
  xmin=-5,
```

```
  xmax=5,
```

```
  axis on top=true,
```

```
  axis x line=middle,
```

```
  axis y line=middle,
```

```
  extra x ticks = {-3},
```

```
  extra y ticks = {1}
```

```
]
```

```
\addplot [forget plot, samples=100, very thick,  
domain=-5:-3.01] {2/(x+3)+1};
```

```
\addplot [forget plot, samples=100, very thick,  
domain=-2.99:5] {2/(x+3)+1};
```

```
\addplot[red, mark=*, only marks] coordinates {(0,0)};
```

```
\draw [color=Green3] (axis cs:-3,-2) -- (axis cs:-3,10);
```

```
\draw [color=Green3] (axis cs:-5,1) -- (axis cs:5,1);
```

```
\end{axis}
```

```
\end{tikzpicture}
```



- 1 Grafik erstellen, dann Datei → Export → PGF/Tikz.
- 2 Das Programm erstellt eine funktionsfähige .tex-Datei:

```
\documentclass[10pt]{article}
\usepackage[utf8]{inputenc}
\usepackage{pgf,tikz}
\usetikzlibrary{arrows}
\pagestyle{empty}

\begin{document}

\begin{tikzpicture}[...]
...
%eigentlicher Grafik-Programmcode
...
\end{tikzpicture}

\end{document}
```



GeoGebra (Fortsetzung)

- 3 Zur Einbindung in ein bestehendes Dokument müssen nur die verwendeten Pakete in die Präambel übernommen werden

```
%... eigener Vorspann  
%hineinkopiert:  
\usepackage{pgf,tikz}  
\usetikzlibrary{arrows}  
\definecolor{qqqqff}{rgb}{0.0,0.0,1.0}
```

Der eigentliche Grafik-Programmcode kann an die jeweilige Stelle eingebunden werden oder wegen der Übersicht per `\input{...}`-Befehl eingebunden werden.

```
%eigentliche Grafik  
\begin{tikzpicture}[...]  
...  
\end{tikzpicture}
```

- 4 Nachträgliche Änderungen können durch das Umschreiben des Codes durchgeführt werden. (Vorhandener Code ist auch ohne Vorkenntnisse gut nachvollziehbar. Ansonsten: Werte Verändern und Reaktion betrachten)



L^AT_EX-Kurs
Grafiken

Inhalt

Pakete

Farben

Boxen

Grafiken

Sonstiges

Zeichnen

Tikz

Nutzung externer
Programme

fragile

Inkscape

<http://www.inkscape.org/de/>

- ⊙ Grafik (svg) (Skalierbare Vektor-Grafik) erstellen, dann Datei → Kopie speichern unter → pdf
- ⊙ Grafik als Bild per `\includegraphics{...}` einbinden
- ⊙ bei Änderungen sollte am besten gleich die svg-Datei überarbeitet werden und ein neues pdf exportiert werden

Gut zu wissen

- ⊙ Inkscape kann auch beliebige pdf-Dokumente bearbeiten
- ⊙ Exportmöglichkeit nach Bitmap vorhanden
- ⊙ Weitere Funktionen ähnlich zu CorelDraw etc.



L^AT_EX-Kurs
Grafiken

Inhalt

Pakete

Farben

Boxen

Grafiken

Sonstiges

Zeichnen

Tikz

Nutzung externer
Programme

fragile

Matlab-Grafiken einbinden

matlab2tikz

Skript, um MatLab-Grafiken zu tikz-Code zu übersetzen:

<http://win.ua.ac.be/~nschloe/content/matlab2tikz>

Matfig2PGF

Skript, um MatLab-Grafiken zu pdg-Code zu übersetzen:

[http:](http://www.mathworks.com/matlabcentral/fileexchange/12962)

[//www.mathworks.com/matlabcentral/fileexchange/12962](http://www.mathworks.com/matlabcentral/fileexchange/12962)



L^AT_EX und Asymptote

- ⊙ stellt Programmiersprache zur Verfügung
- ⊙ Rechnen, Zeichnen, Array und Variablen, ...
- ⊙ `\usepackage{asymptote}` Paket laden
- ⊙ Komplizierteres Mehrfachübersetzen nötig.
- ⊙ Unsichtbares Programm Asymptote muss aufgerufen werden
- ...

```
\begin{figure}  
\centering  
\begin{asy}  
...befehle...  
\end{asy}  
\caption{Eingebettete Grafik}  
\end{figure}
```

Siehe auch <http://www.math.tugraz.at/Orientierung/>



Asymptote mit Mac

Mac: Verwende den mitgelieferten Editor TeXShop

Erstmalige Aktivierung von Asymptote beim Editor:

- ⦿ gehe in den Ordner `/Library/TeXShop/Engines/` (evtl. Finder → Gehe zu → Gehe zum Ordner)
- ⦿ Verschiebe die Datei `Asymptote.engine` vom `Inactive/Asymptote`-Ordner in den `Engines`-Ordner.
- ⦿ TeXShop erneut öffnen.
- ⦿ Die Datei `sample.tex` kann zum Testen verwendet werden.

Kompilieren:

- 1 Mit z. B. »LaTeX« setzen
- 2 Statt »LaTeX« etc neben »Setzen« den Wert »Asymptote« auswählen
- 3 Auf »Setzen« klicken