

Wissenschaftliche Arbeiten schreiben mit \LaTeX

Workshop 7 – Grafiken

Fadi Semmo

30. September 2010

Institut für Erd- und Umweltwissenschaften
Universität Potsdam

Veranstaltungstermine

27.09.10: Einführung

28.09.10: KOMA-Script, Mathematisatz

29.09.10: Gleitobjekte (Abbildungen, Tabellen), Zusatzpakete

30.09.10: Verzeichnisse (Bibliografie, Index, Glossar), Grafiken

01.10.10: Präsentationen

Letzter Workshop

- Aufbau einer Bibliografie
- Zitierstile mit `biblatex`
- Anlegen einer Bib-Datei
- Eintragstypen (`article`, `book`, `online`)
- Ausgabe des Literaturverzeichnisses
- Erstellen eines Stichwortverzeichnisses mit `makeindex`
- Indexeinträge (Einfache, Querverweise, Untergeordnete)
- Erstellen eines Glossars (Paket `glossaries`)
- Erstellen eines Abkürzungsverzeichnisses (Paket `glossaries`)

Was ist PGF/TikZ?

- Portable Graphic Format
- Entwickelt von Till Tantau (Autor von beamer)
- Makropaket zur Erstellung von Grafiken mit \TeX
- Syntax von METAFONT und PSTricks
- Plattform- und formatunabhängig
- Arbeitet mit den wichtigsten Treibern zusammen (pdf \TeX , dvips)
- Ausgabeformat: PS oder PDF
- Benutzerfreundliche Schnittstelle TikZ
- TikZ ist *kein* Zeichenprogramm

Das Paket PGF/TikZ

- Projektseite: <http://sourceforge.net/projects/pgf/>
- Erstellung hochqualitativer Vektorgrafiken (z. B. für Publikationen)
- Spezielle Syntax für Koordinaten, Pfade, Knoten, Matrizen, Bäume, Funktionen etc.
- Paket mit `\usepackage{tikz}` laden
- Keine Paketoptionen

Grafiken erstellen (Umgebungen)

- Grafiken werden mit der `tikzpicture`-Umgebung erstellt
- Alle TikZ-Makros werden innerhalb der Umgebung angegeben (Ausnahme `\tikzset`)
- Für Inline-Grafiken kann das Makro `\tikz` verwendet werden

```
% Große Grafiken
\begin{tikzpicture} [Optionen]
...
\end{tikzpicture}
```

```
% Inline-Grafiken
\tikz [Optionen] <Text>;
```

Gruppierungen

- Gruppierungen werden mit der `scope`-Umgebung erstellt
- `scope`-Umgebung wird innerhalb der `tikzpicture`-Umgebung definiert

```
\begin{tikzpicture} [Optionen]
...
\begin{scope} [Optionen]
...
\end{scope}
...
\end{tikzpicture}
```

Syntax von Koordinatenangaben

- Einfachste Möglichkeit als x,y-Wertepaar: $(1\text{cm}, 2\text{pt})$
- Polarkoordinaten: $(30 : 1\text{cm}) \rightarrow 1\text{ cm in Richtung } 30\text{ Grad}$
- Ohne Längenangabe: $(1, 1)$ bzw. $(1, 1, 1)$
- Länge des x-, y- bzw. z-Vektors beträgt standardmäßig 1 cm
- Relative Koordinaten: $\text{\texttt{++}}(1, 0) \rightarrow \text{Verschiebung } 1\text{ cm nach rechts}$
- Mit $+(1, 0)$ relative Verschiebung zum Referenzpunkt
- Koordinatensysteme: kartesisch, tangential etc.
- Bestimmung eigener Koordinatensysteme

Syntax von Pfadangaben

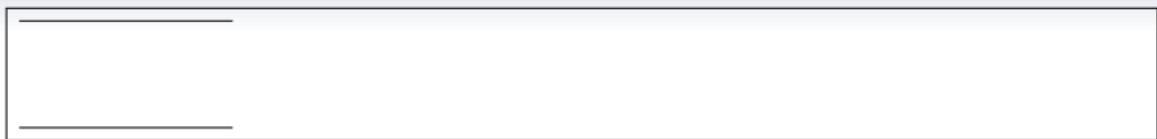
- Pfade bestehen aus einer Serie gerader und gekrümmter Linien, die miteinander verknüpft werden
- Einfache horizontale Linien werden mit `--` erstellt
- Operationen: `cycle`, `circle`, `ellipse`, `arc`, `grid` etc.

```
\path [Optionen] <Spezifikation> oder  
\draw [Optionen] <Spezifikation>
```

Pfad-Operationen

```
\path ... <Koordinate> ...; % Gehe zu
\path ... --<Koordinate> ...; % Linie zu
\path ... --cycle ...; % Zyklus
\path ... rectangle(<Ecken>) ...; % Rechteck
\path ... circle(<Radius>) ...; % Kreis
\path ... ellipse(<Breite> and <Höhe>) ...; %
Ellipse
\path ... arc(<Startwinkel>:<Endwinkel>:<Radius> and
<Höhe>) ...; % Bogen
\path ... grid [Optionen] <Ecken> ...; % Gitter
```

Die Operation „Gehe zu“



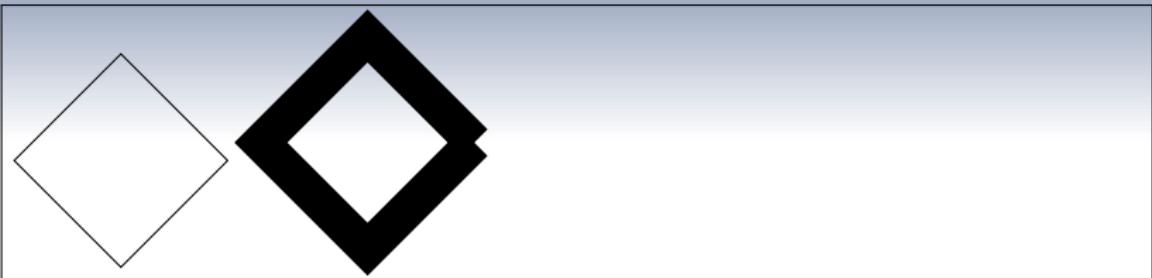
```
1 \tikz \draw (0,0) --(2,0) (0,1) --(2,1);
```

Die Operation „Linie zu“



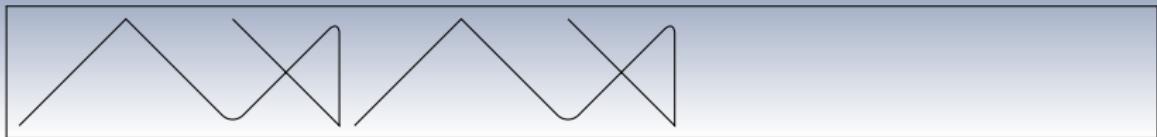
```
1 \begin{tikzpicture}[line width=10pt]
2   \draw (0,0) --(1,1) (1,1) --(2,0);
3   \draw (3,0) -- (4,1) -- (5,0);
4   \useasboundingbox (0,1.5); % make bounding box
      higher
5 \end{tikzpicture}
```

Die Operation „Linie zu“



```
1 \tikz \draw (1,0) -- (0,1) -- (-1,0) -- (0,-1)  
    -- (1,0);  
2 \tikz \draw[line width=10pt] (1,0) -- (0,1) --  
    (-1,0) -- (0,-1) -- (1,0);
```

Richtiger Umgang mit Pfad-Optionen



```
1 \tikz \draw (0,0) -- (1,1)
2 [rounded corners] -- (2,0) -- (3,1)
3 [sharp corners] -- (3,0) -- (2,1);
4 \tikz \draw (0,0) -- (1,1)
5 {[rounded corners] -- (2,0) -- (3,1)} %
6   Gruppierung
-- (3,0) -- (2,1);
```

Richtiger Umgang mit Pfad-Optionen



```
1 \tikz \draw (0,0) -- (1,1)
2 [color=red] -- (2,0) -- (3,1)
3 [color=blue] -- (3,0) -- (2,1);
4 \tikz{\draw (0,0) -- (1,1); % Aufteilung in
      mehrere Pfade
5 \draw [color=red] (2,0) -- (3,1);
6 \draw [color=blue] (3,0) -- (2,1);}
```

Relative Koordinatenangaben



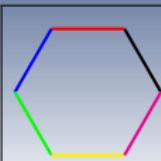
```
1 \tikz \draw[thick,blue] (0,0) -- (1,0) -- (1,1)  
    -- (0,1) -- (0,0);  
2 \tikz \draw[red] (0,0) -- ++(1,0) -- ++(0,1) --  
    ++(-1,0) -- ++(0,-1);  
3 \tikz \draw[thick] (0,0) -- +(1,0) -- +(1,1) --  
    +(0,1) -- +(0,0);
```

Polarcoordinaten



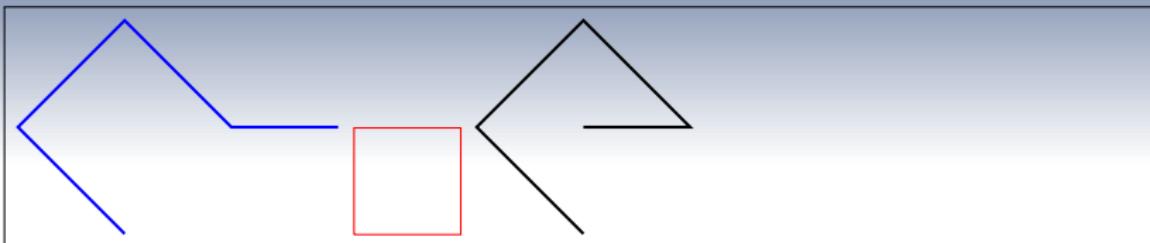
```
1 \tikz [color=red ,thick]
2 \draw (60:4ex) -- (120:4ex) -- (180:4ex) --
      (240:4ex) -- (300:4ex) -- (360:4ex) -- (60:4
      ex);
```

Polarcoordinaten



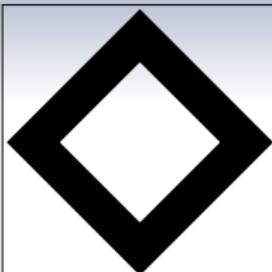
```
1 \begin{tikzpicture}[thick]
2   \draw (0:4ex) -- (60:4ex);
3   \draw [color=red] (4ex,4ex) (60:4ex) -- (120:4ex)
4     ;
5   \draw [color=blue] (4ex,4ex) (120:4ex) -- (180:4
6     ex);
7   \draw [color=green] (180:4ex) -- (240:4ex);
8   \draw [color=yellow] (240:4ex) -- (300:4ex);
9   \draw [color=magenta] (300:4ex) -- (360:4ex);
10  \end{tikzpicture}
```

Relative Polarkoordinatenangaben



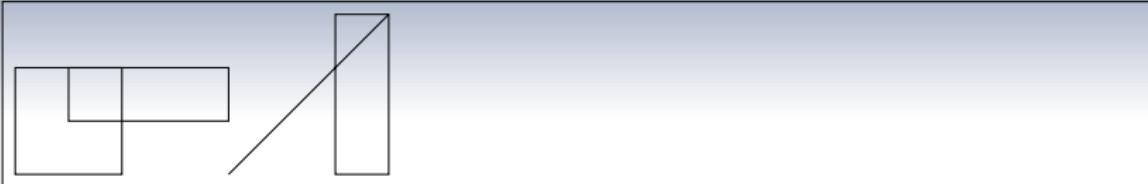
```
1 \tikz \draw[thick,blue] (2cm,0) -- (0:1cm) --
    (90:1cm) -- (180:1cm) -- (270:1cm);
2 \tikz \draw[red] (3cm,0) -- ++(0:1cm) -- ++
(90:1cm) -- ++
(180:1cm) -- ++
(270:1cm);
3 \tikz \draw[thick] (6cm,5) -- +(right:1cm) -- +
(up:1cm) -- +(left:1cm) -- +(down:1cm);
```

Die Operation cycle



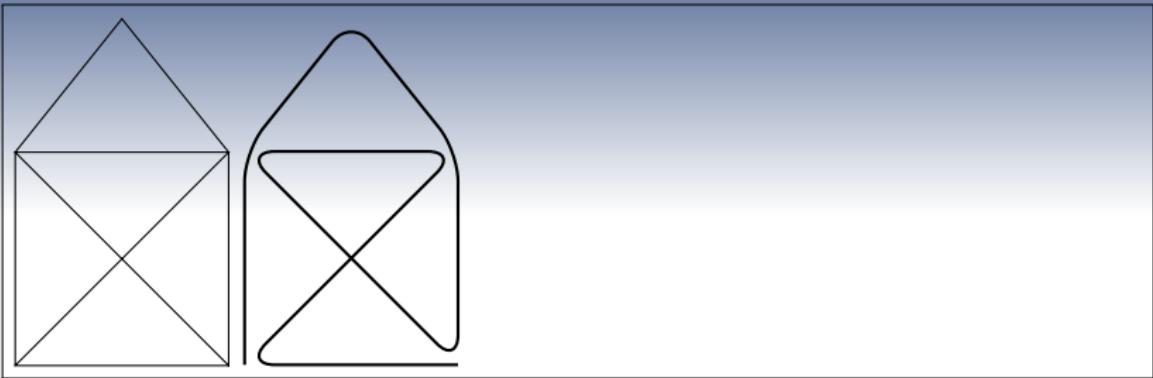
```
1 \tikz \draw [line width=10pt] (1,0) -- (0,1) --  
(-1,0) -- (0,-1) -- cycle;
```

Die Operation rectangle



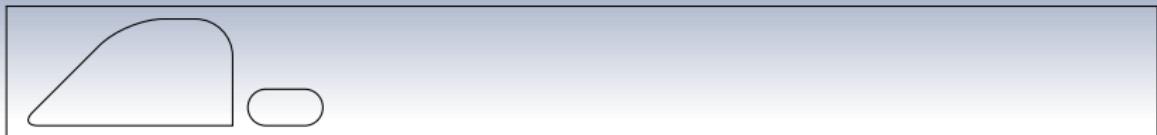
```
1 \begin{tikzpicture}
2 \draw (0,0) rectangle (1,1);
3 \draw (.5,1) rectangle (2,0.5) (3,0) rectangle
   (3.5,1.5) -- (2,0);
4 \end{tikzpicture}
```

Abgerundete Ecken: rounded corners



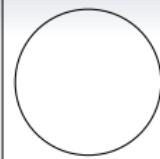
```
1 \tikz \draw
2 (0,0) -- (0,2) -- (1,3.25) -- (2,2) -- (2,0) --
3 (0,2) -- (2,2) -- (0,0) -- (2,0);
4 \tikz \draw[thick,rounded corners=8pt]
5 (0,0) -- (0,2) -- (1,3.25) -- (2,2) -- (2,0) --
6 (0,2) -- (2,2) -- (0,0) -- (2,0);
```

Abgerundete Ecken: rounded corners



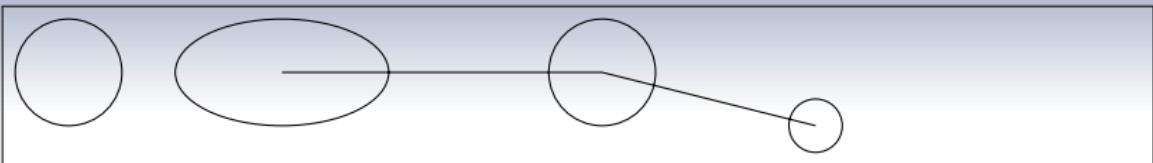
```
1 \tikz \draw (0,0) [rounded corners=10pt] --
2   (1,1) -- (2,1)
3 [sharp corners] -- (2,0)
4 [rounded corners=5pt] -- cycle;
5 \tikz \draw[rounded corners=1ex] (0,0) rectangle
6   (20pt,2ex);
```

Die Operation circle



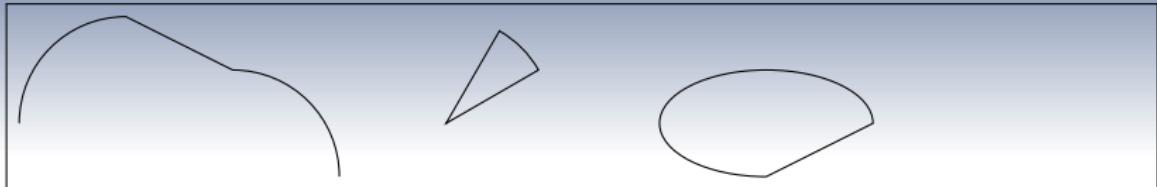
```
1 \tikz \draw (1,0) circle (4ex);
```

Die Operation ellipse



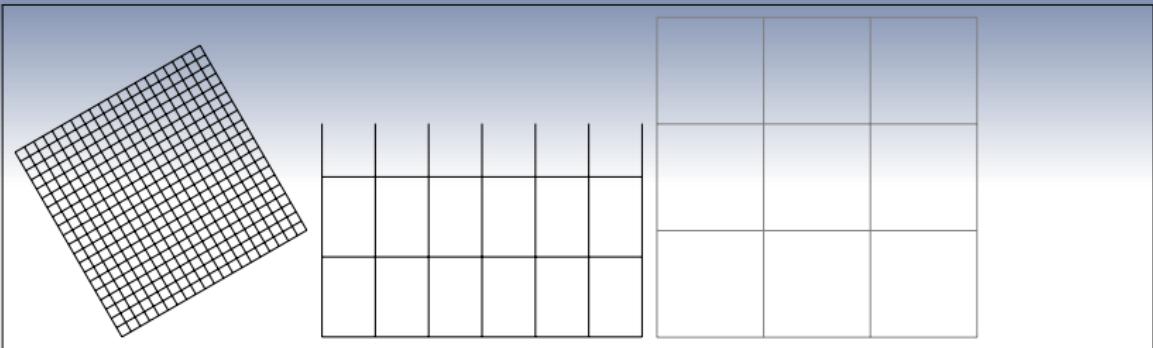
```
1 \begin{tikzpicture}
2   \draw (1,0) circle (.5cm);
3   \draw (3,0) ellipse (1cm and .5cm) -- ++(3,0)
4     circle (.5cm) -- ++(2,-.5) circle (.25cm);
4 \end{tikzpicture}
```

Die Operation arc



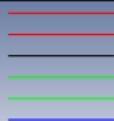
```
1 \begin{tikzpicture}
2   \draw (0,0) arc (180:90:1cm) -- (2,.5) arc
3     (90:0:1cm);
4   \draw (4,0) -- +(30:1cm) arc (30:60:1cm) --
5     cycle;
6   \draw (8,0) arc (0:270:1cm and .5cm) -- cycle;
7 \end{tikzpicture}
```

Die Operation grid



```
1 \tikz [rotate=30] \draw [step=1mm] (0,0) grid  
      (2,2);  
2 \tikz \draw (0,0) grid [xstep=.5,ystep=.75]  
      (3,2);  
3 \tikz \draw [help lines] (0,0) grid (3,3);
```

Gruppierungen: scope



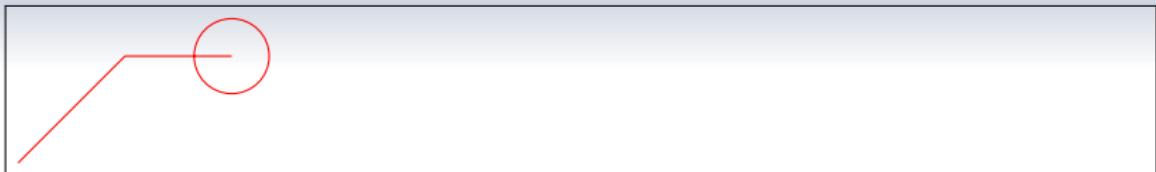
```
1 \begin{tikzpicture}
2   \begin{scope}[color=red]
3     \draw (0mm,10mm) -- (10mm,10mm);
4     \draw (0mm, 8mm) -- (10mm, 8mm);
5   \end{scope}
6   \draw (0mm,6mm) -- (10mm,6mm);
7   \begin{scope}[color=green]
8     \draw (0mm, 4mm) -- (10mm, 4mm);
9     \draw (0mm, 2mm) -- (10mm, 2mm);
10    \draw [color=blue] (0mm, 0mm) -- (10mm, 0mm);
11  \end{scope}
12 \end{tikzpicture}
```

Pfad-Aktionen

- Pfade können unterschiedlich dargestellt werden
- Aktion wird als Option an \path übergeben
- Kombinationen möglich

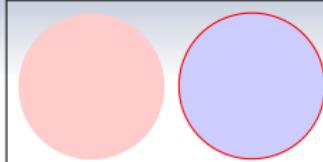
```
\draw % Abk. für \path[draw]
\fill % Abk. für \path[fill]
\filldraw % Abk. für \path[fill,draw]
\pattern % Abk. für \path[pattern]
\shade % Abk. für \path[shade]
\shadedraw % Abk. für \path[shadedraw]
\clip % Abk. für \path[clip]
```

Linien zeichnen: draw



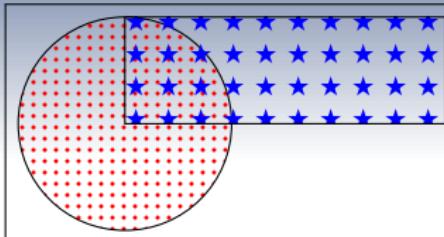
```
1 \tikz \path [draw=red] (0,0) -- (1,1) -- (2,1)  
      circle (10pt);
```

Grafiken füllen: fill



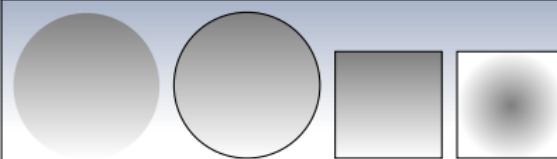
```
1 \tikz \fill [color=red!20] (0,0) circle (4ex);  
2 \tikz \draw [fill=blue!20, draw=red] (0,0) circle  
        (4ex);
```

Muster erstellen: pattern



```
1 \begin{tikzpicture}
2   \draw [pattern color=red,pattern=dots] (0,0)
3     circle (1cm);
4   \draw [pattern color=blue,pattern=fivepointed
5     stars] (0,0) rectangle (3,1);
6 \end{tikzpicture}
```

Schattierungen: shade



```
1 \tikz \shade (0,0) circle (4ex);
2 \tikz \shadedraw (0,0) circle (4ex);
3 \tikz \shadedraw [shading=axis] (0,0) rectangle
    (1,1);
4 \tikz \shadedraw [shading=radial] (0,0)
    rectangle (1,1);
```

Schattierungen: shade



```
1 \tikz \draw[top color=red] (0,0) rectangle (2,1)
  ;
2 \tikz \draw[top color=white,bottom color=black,
  middle color=red] (0,0) rectangle (2,1);
3 \tikz \draw[inner color=red] (0,0) rectangle
  (2,1);
4 \tikz \draw[outer color=red,inner color=white]
  (0,0) rectangle (2,1);
```

Schattierungen: shade



```
1 \begin{tikzpicture}
2 \shade [ball color=white] (0,0) circle (2ex);
3 \shade [ball color=red] (1,0) circle (2ex);
4 \shade [ball color=black] (2,0) circle (2ex);
5 \end{tikzpicture}
```

Verschneidungen: clip



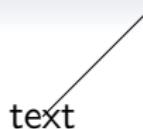
```
1 \begin{tikzpicture}
2 \draw[clip] (0,0) circle (1cm);
3 \fill[red] (1,0) circle (1cm);
4 \end{tikzpicture}
```

Syntax von Knotenpunkten

- Bestehen meist aus Rechtecken, Kreisen oder anderen einfachen Formen und etwas Text
- Knotenpunkte werden Pfaden hinzugefügt, sind aber nicht Teil des Pfades selbst
- Werden an der aktuellen Position des Pfades gesetzt (nach der Koordinate!)
- Bekannte Optionen wie `draw` oder `fill` verwendbar
- Referenz auf einen Knotenpunktes mit (`<Name>`) möglich

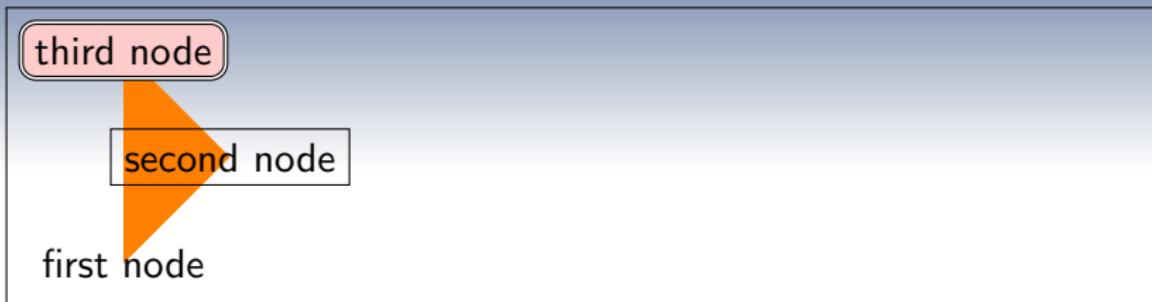
```
node [<Optionen>] (<Name>) at (<Koordinate>){<Text>}
```

Knotenpunkte: node



```
1 \tikz \draw (1,1) node {text} -- (2,2);
```

Mehrere Knotenpunkte



```
1 \tikz \fill[fill=orange]
2 (0,0) node {first node}
3 -- (1,1) node[draw] {second node}
4 -- (0,2) node[fill=red!20,draw,double,rounded
corners] {third node};
```

Innere Abstände: inner sep

default

loose

tight

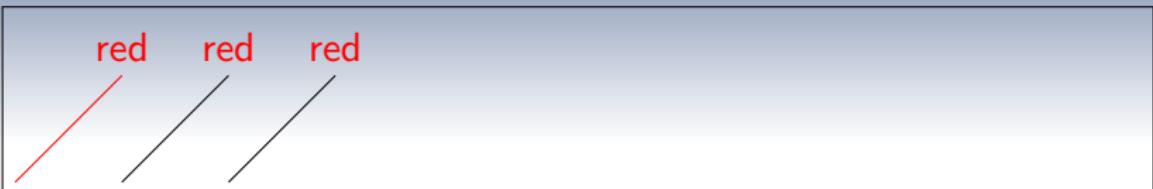
```
1 \begin{tikzpicture}
2   \draw (0,0) node[inner sep=0pt,draw] {tight}
3   (0cm,2em) node[inner sep=5pt,draw] {loose}
4   (0cm,4em) node[fill=red!20] {default};
5 \end{tikzpicture}
```

Höhe und Breite: minimum height/width



```
1 \begin{tikzpicture}
2   \draw (0,0) node [minimum height=1cm, draw] {1cm};
3   (2,0) node [minimum height=0cm, draw] {0cm};
4 \end{tikzpicture}
```

Textanordnung: above



```
1 \begin{tikzpicture}
2   \draw [red] (0,0) -- +(1,1) node[above] {red};
3   \draw [text=red] (1,0) -- +(1,1) node[above] {red};
4   };
5   \draw (2,0) -- +(1,1) node[above,red] {red};
6 \end{tikzpicture}
```

Textanordnung: text justified/centered

This is a demonstration text for showing how line breaking works.

This is a demonstration text for showing how line breaking works.

This is a demonstration text for showing how line breaking works.

```
1 \tikz \draw (0,0) node[fill=red!30, text width=3  
cm] {This is a demonstration text for showing  
how line breaking works.};  
2 \tikz \draw (0,0) node[fill=red!30, text width=3  
cm, text justified] {This is a demonstration  
text for showing how line breaking works.};  
3 \tikz \draw (0,0) node[fill=red!30, text width=3  
cm, text centered] {This is a demonstration  
text for showing how line breaking works.};
```

Positionierung: anchor

x y t

x y t

x y t

```
1 \begin{tikzpicture}[scale=2,transform shape]
2   \draw[anchor=center] (0,1) node{x} -- (0.5,1)
3     node{y} -- (1,1) node{t};
4   \draw[anchor=base] (0,.5) node{x} -- (0.5,.5)
5     node{y} -- (1,.5) node{t};
6   \draw[anchor=mid] (0,0) node{x} -- (0.5,0) node{
7     y} -- (1,0) node{t};
8 \end{tikzpicture}
```

Positionierung: above

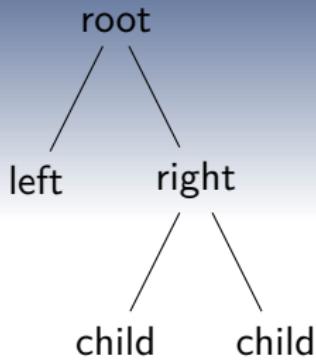
above above above left above right

```
1 \tikz \fill (0,0) circle (2pt) node[above] {
2   above};
3 \tikz \fill (0,0) circle (2pt) node[above=2pt] {
4   above};
5 \tikz \fill (0,0) circle (2pt) node[above left]
6   {above left};
7 \tikz \fill (0,0) circle (2pt) node[above right]
8   {above right};
```

Syntax von Bäumen

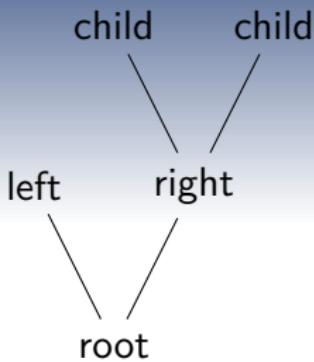
- Erweiterung von Knotenpunkten durch sogenannte „Kinder“ (Schlüsselwort `child`)
- Beliebig viele „Kinder“ können auf einem Knotenpunkt folgen
- „Kinder“ bilden selbst wieder einen Knotenpunkt

Die Operation child



```
1 \begin{tikzpicture}
2   \node {root}
3   child {node {left}}
4   child {node {right}}
5   child {node {child}}
6   child {node {child}}};
7 \end{tikzpicture}
```

Die Operation child



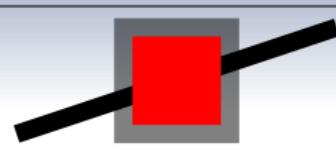
```
1 \begin{tikzpicture}
2   \node {root} [grow'=up]
3   child {node {left}}
4   child {node {right}}
5   child {node {child}}
6   child {node {child}};
7 \end{tikzpicture}
```

Die Operation child



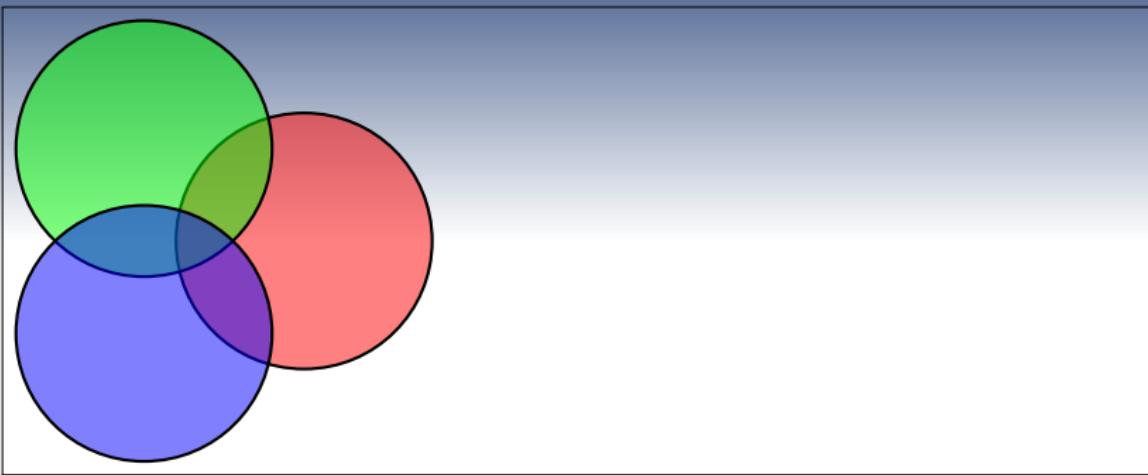
```
1 \begin{tikzpicture}
2   \node [rectangle, draw] {root}
3   child {node [circle, draw] (left_node) {left}}
4   child {node [rectangle, draw] (right_node) {right}}
5   \draw[dashed, ->] (left_node) -- (right_node);
6 \end{tikzpicture}
```

Tranparenz: draw opacity



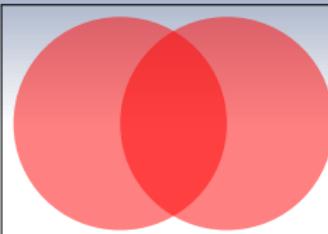
```
1 \begin{tikzpicture}[line width=1ex]
2   \draw (0,0) -- (3,1);
3   \filldraw [fill=red, draw opacity=0.5] (1,0)
4     rectangle (2,1);
5 \end{tikzpicture}
```

Tranparenz: fill opacity



```
1 \begin{tikzpicture}[thick,fill opacity=0.5]
2 \filldraw [fill=red] (0:1cm) circle (12mm);
3 \filldraw [fill=green] (120:1cm) circle (12mm);
4 \filldraw [fill=blue] (-120:1cm) circle (12mm);
5 \end{tikzpicture}
```

Tranparenz: fill opacity



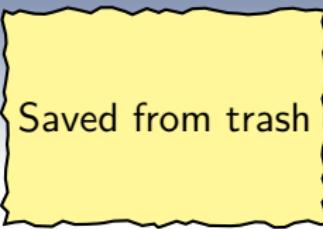
```
1 \begin{tikzpicture}[fill opacity=0.5]
2   \fill[red] (0,0) circle (1);
3   \fill[red] (1,0) circle (1);
4 \end{tikzpicture}
```

Pfade dekorieren: decoration



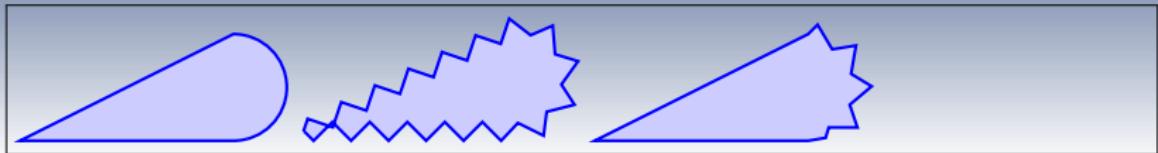
```
1 \begin{tikzpicture}
2 \node [fill=red!20, draw, decorate, decoration={bumps, mirror}, minimum height=1cm] {Bumpy};
3 \end{tikzpicture}
```

Pfade dekorieren: decoration



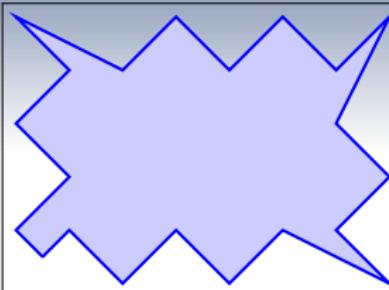
```
1 \begin{tikzpicture}
2 \node [fill=yellow!50, draw, thick, minimum height
      =2cm, minimum width=3cm,
      decorate, decoration={random steps,segment
      length=3pt,amplitude=1pt}]
      {Saved from trash};
5 \end{tikzpicture}
```

Pfade dekorieren: decoration



```
1 \tikz \fill [fill=blue!20,draw=blue,thick] (0,0)
    -- (2,1) arc (90:-90:.5) -- cycle;
2 \tikz \fill [decorate,decoration={zigzag}] [fill
    =blue!20,draw=blue,thick] (0,0) -- (2,1) arc
(90:-90:.5) -- cycle;
3 \tikz \fill [decoration={zigzag}] [fill=blue!20,
    draw=blue,thick] (0,0) -- (2,1) decorate {
    arc (90:-90:.5) } -- cycle;
```

Pfade dekorieren: decoration



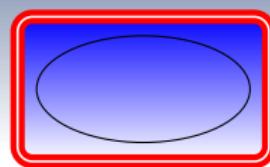
```
1 \tikz \fill [fill=blue!20, draw=blue, thick]
  decorate[decoration={zigzag, segment length=10
  mm, amplitude=2.5mm}] { (0,0) rectangle (3,2)
  };
```

(Ausgewählte) Bibliotheken

- Werden mit `\usetikzlibrary{<Bibliotheken>}` geladen
- Erweitern TikZ um zusätzliche Objekte, Symbole, Muster etc.

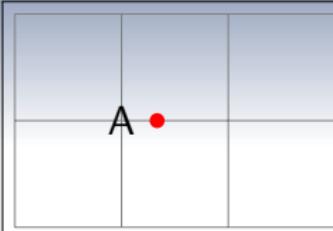
<i>Bibliothek</i>	<i>Beschreibung</i>
arrows	Pfeilspitzen
backgrounds	Rahmen
calc	Mathematische Zeichnungen
calendar	Kalender
matrix	Matrizen
mindmap	Gedankenkarten
patterns	Muster
shadows	Schatten
trees	Bäume

Die Bibliothek backgrounds



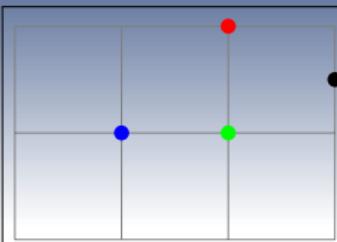
```
1 \begin{tikzpicture}
2 [background rectangle/.style=
3 {double, ultra thick, draw=red, top color=blue,
4 rounded corners},
5 show background rectangle]
6 \draw (0,0) ellipse (10mm and 5mm);
7 \end{tikzpicture}
```

Die Bibliothek calc



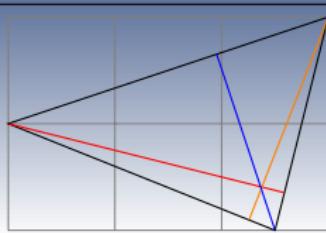
```
1 \begin{tikzpicture}
2   \draw [help lines] (0,0) grid (3,2);
3   \node (a) at (1,1) {A};
4   \fill [red] ($(a) + 1/3*(1cm,0)$) circle (2pt);
5 \end{tikzpicture}
```

Die Bibliothek calc



```
1 \begin{tikzpicture}
2   \draw [help lines] (0,0) grid (3,2);
3   \fill [red] ($2*(1,1)$) circle (2pt);
4   \fill [green] (${1+1}*(1,.5)$) circle (2pt);
5   \fill [blue] ($\cos(0)*\sin(90)*(1,1)$) circle (2
    pt);
6   \fill [black] (${3*(4-3})*(1,0.5)$) circle (2pt)
    ;
7 \end{tikzpicture}
```

Die Bibliothek calc



```
1 \begin{tikzpicture}
2   \draw [help lines] (0,0) grid (3,2);
3   \coordinate (a) at (0,1);
4   \coordinate (b) at (3,2);
5   \coordinate (c) at (2.5,0);
6   \draw (a) -- (b) -- (c) -- cycle;
7   \draw[red] (a) -- ($(b)!(a)!(c)$);
8   \draw[orange] (b) -- ($(a)!(b)!(c)$);
9   \draw[blue] (c) -- ($(a)!(c)!(b)$);
10  \end{tikzpicture}
```

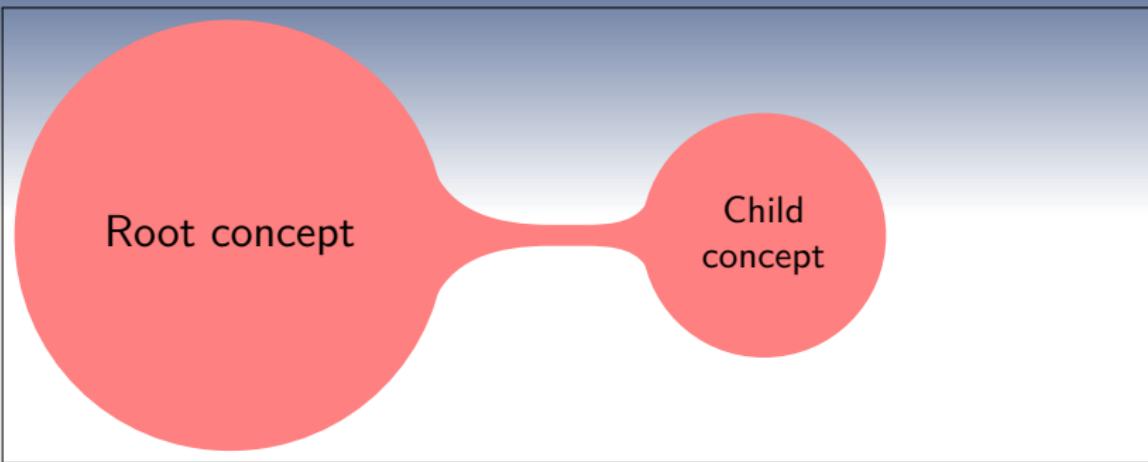
Die Bibliothek calendar

January 2000

		1	2			
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30
31						

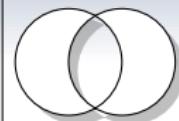
```
1 \tikz \calendar [dates=2000-01-01 to 2000-01-31 ,  
2   week list ,  
3   month label above centered ,  
4   month text=\textcolor{red}{\%mt} \%y-];
```

Die Bibliothek mindmap



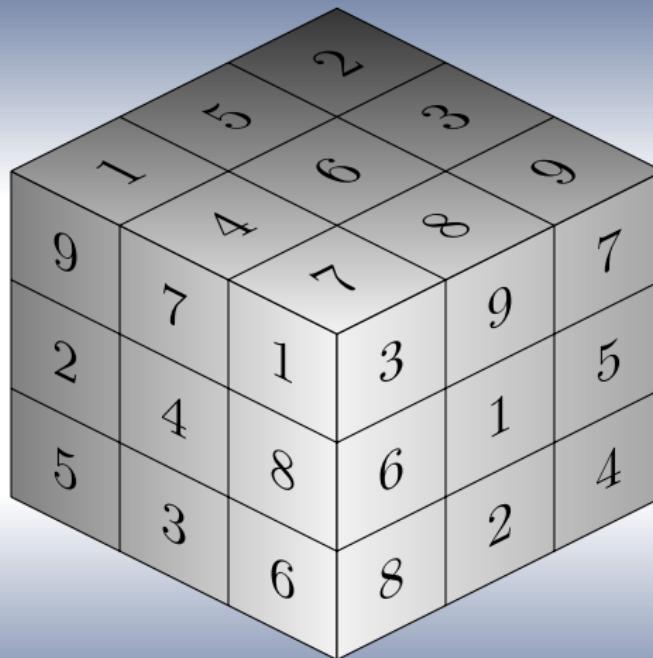
```
1 \tikz[mindmap,concept color=red!50]
2 \node [concept] {Root concept}
3 child[grow=right] {node[concept] {Child concept
}};
```

Die Bibliothek shadows



```
1 \tikz [even odd rule]
2 \filldraw [drop shadow,fill=white] (0,0) circle (.5) (0.5,0) circle (.5);
```

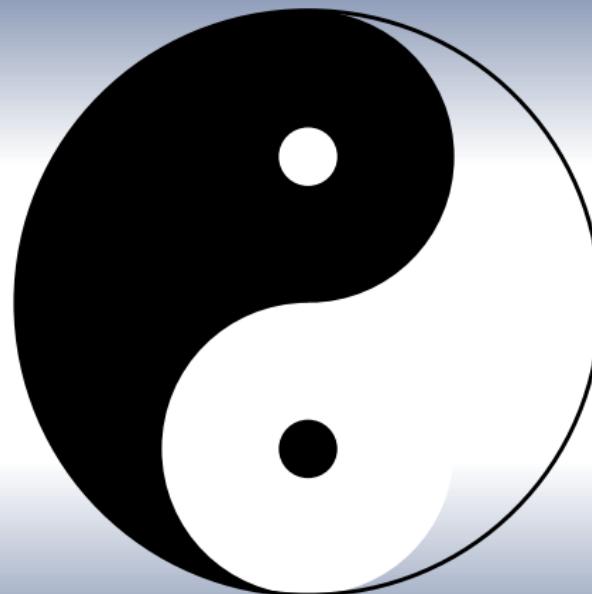
Beispiel 1: shade



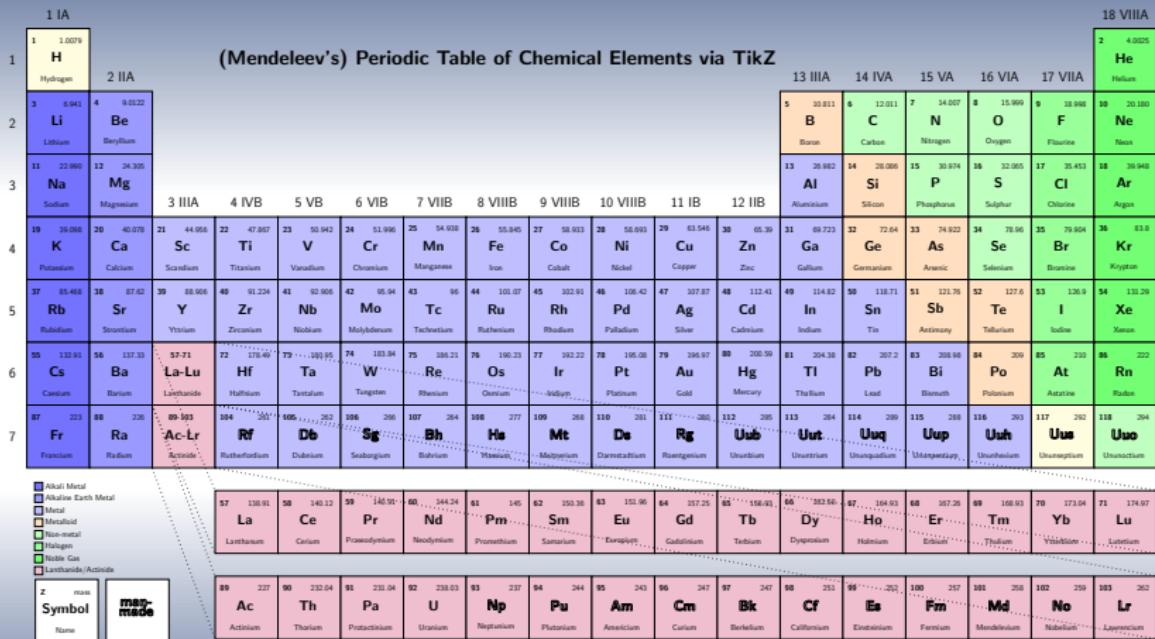
Beispiel 2: clip



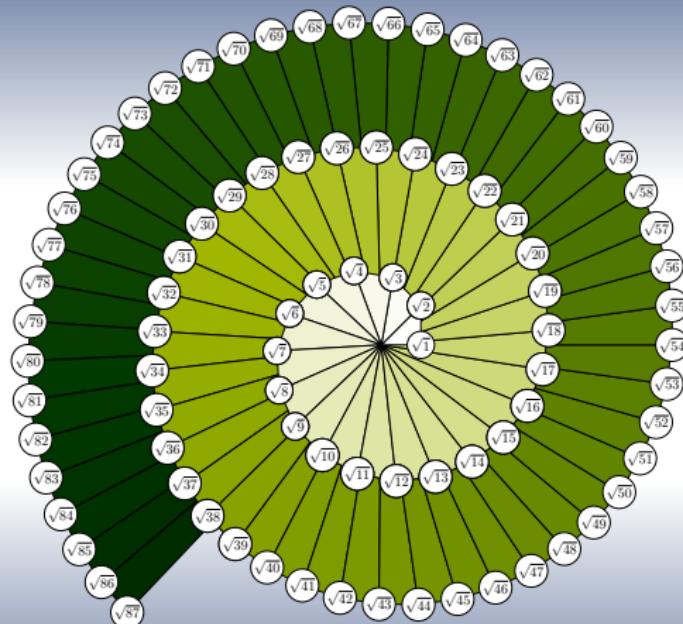
Beispiel 3: clip



Beispiel 4: node



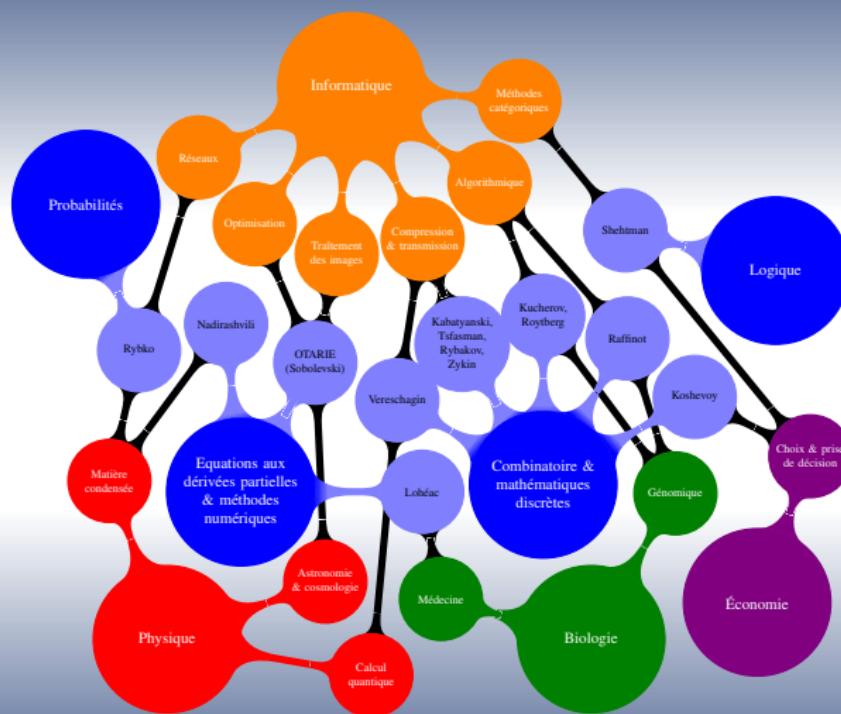
Beispiel 5: node



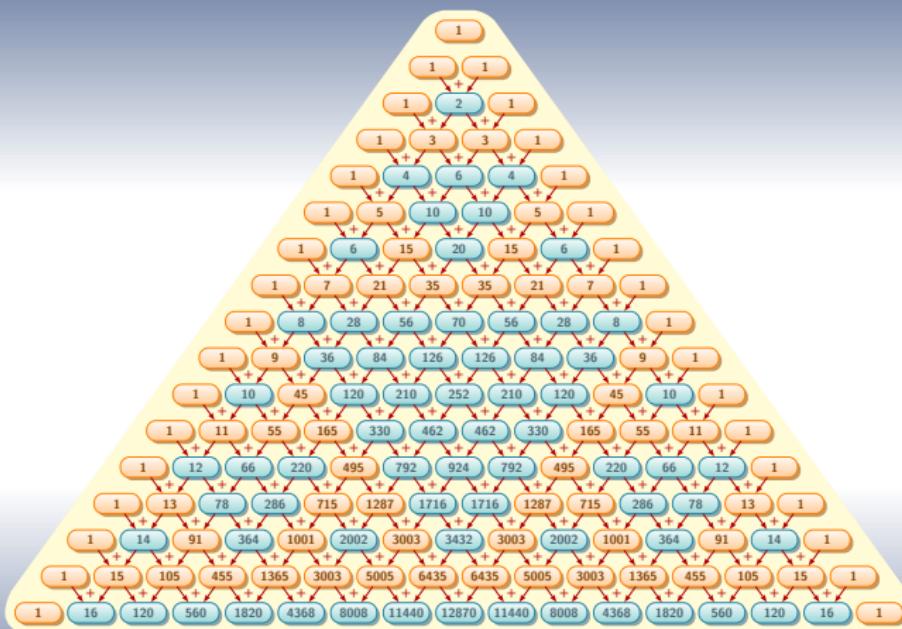
Beispiel 6: child



Beispiel 7: mindmap



Beispiel 8: foreach



Beispiel 9: foreach



Übung 7a

- Konstruktion von Linien
- Relative Koordinatenangaben
- Operationen `rectangle`, `circle`, `ellipse`, `arc` und `grid`
- Transparenz
- Knotenpunkte
- Baumdiagramm
- Gruppierungen

Das Paket pgfplots

- Basiert auf PGF/TikZ
- Ermöglicht hochqualitative 2D- und 3D-Diagramme unter L^AT_EX
- Erstellung von Linien-, Streu- und Balkendiagrammen
- Vorteil: Konsistenz bei Schriftart/Schriftgröße/Grafiken, Mathematisatz von T_EX, Präambel-Einstellungen werden übernommen
- Alternative zu G_NUPL_OT, M_ATHEM_ATICA und M_ATLAB
- Starke Verbesserungen mit Version 1.3
- Paket mit \usepackage{pgfplots} laden
- Daten importieren: Paket pgfplotstable

Achsen erstellen

- Achsen: normal, logarithmisch oder semi-logarithmisch
- Normal: axis
- Logarithmisch: loglogaxis
- Semi-logarithmisch: semilogxaxis oder semilogyaxis

```
\begin{tikzpicture} [Optionen]
\begin{axis} [Optionen]
...
\end{axis}
\end{tikzpicture}
```

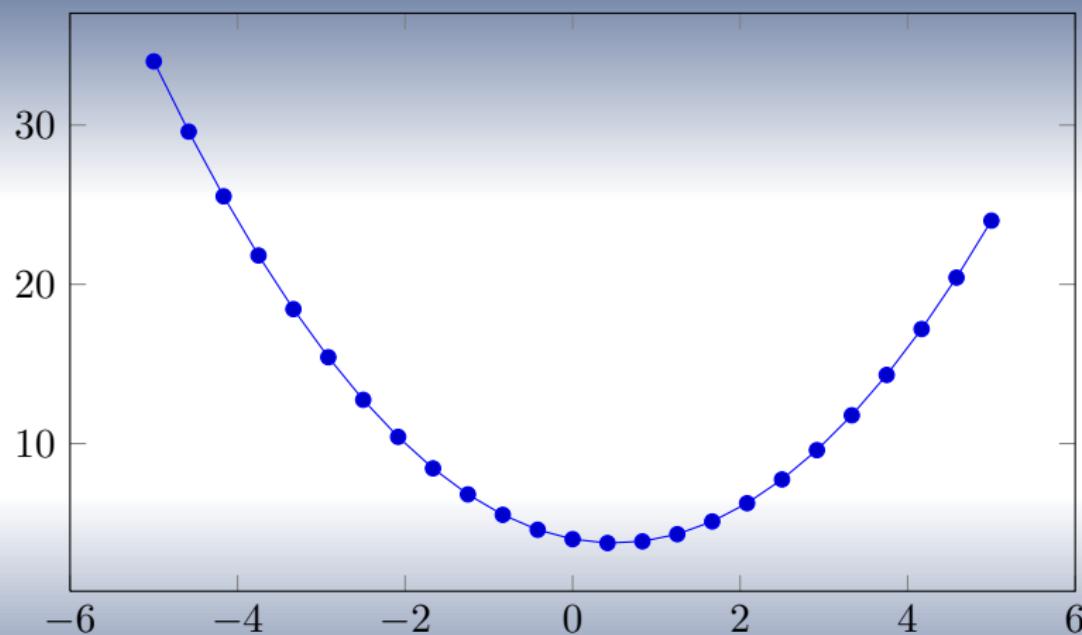
2D-Diagramme: \addplot

```
\begin{tikzpicture} [Optionen]
\begin{axis} [Optionen]
...
\addplot [Optionen] ...;
...
\end{axis}
\end{tikzpicture}
```

Funktionenplot

```
1 \begin{tikzpicture}
2 \begin{axis}
3 \addplot {x^2 - x + 4};
4 \end{axis}
5 \end{tikzpicture}
```

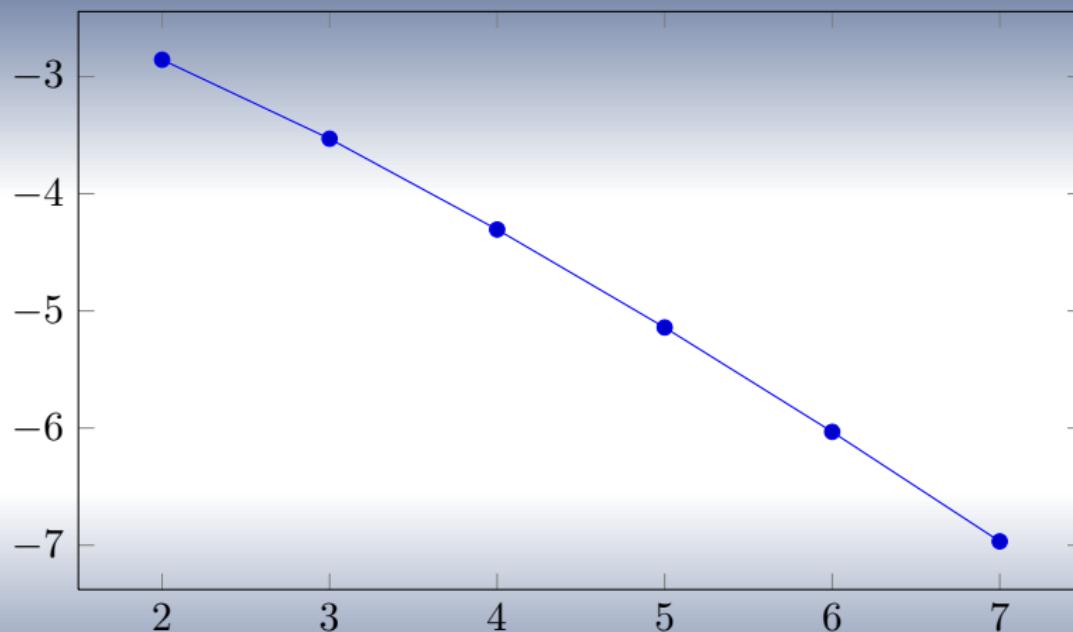
Funktionenplot



Datenplot: coordinates

```
1 \begin{tikzpicture}
2 \begin{axis}
3 \addplot coordinates {
4 (2,-2.8559703)
5 (3,-3.5301677)
6 (4,-4.3050655)
7 (5,-5.1413136)
8 (6,-6.0322865)
9 (7,-6.9675052)
10 };
11 \end{axis}
12 \end{tikzpicture}
```

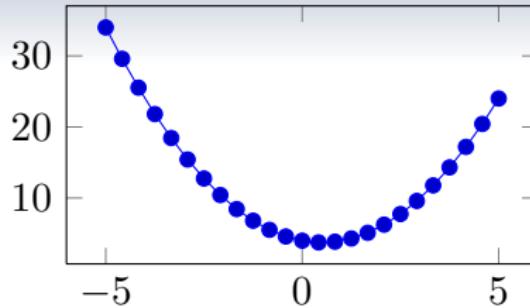
Datenplot: coordinates



Höhe und Breite: height, width

```
1 \begin{tikzpicture}
2 \begin{axis}[height=4cm, width=6cm]
3 \addplot {x^2 - x + 4};
4 \end{axis}
5 \end{tikzpicture}
```

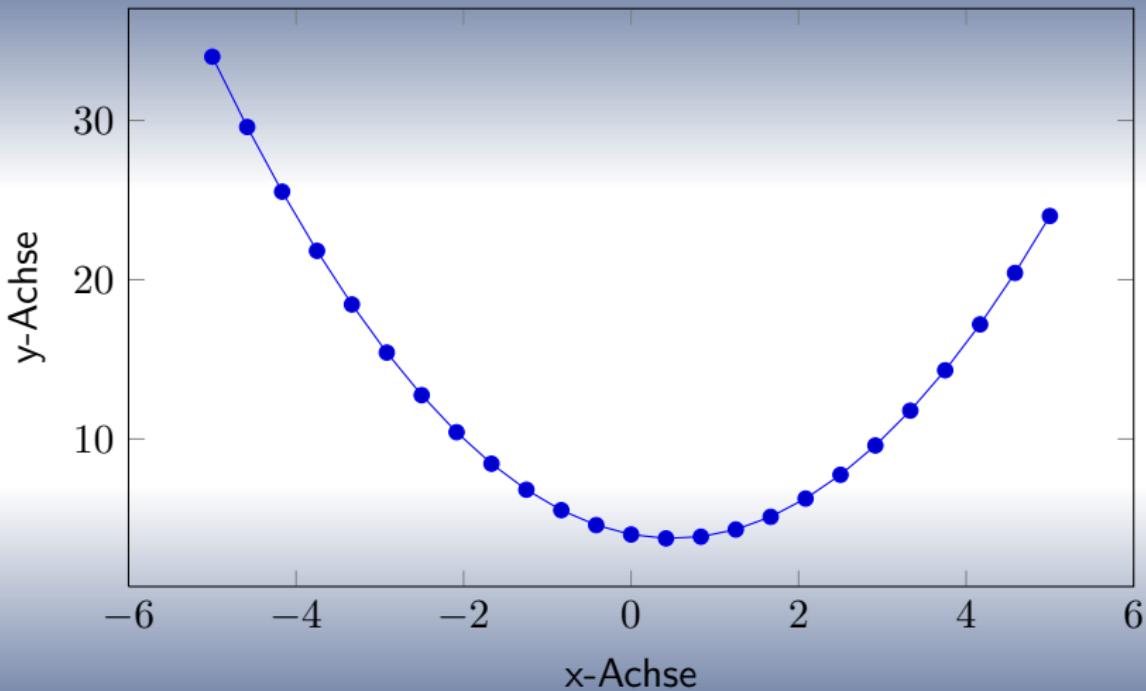
Höhe und Breite: height, width



Achsenbeschriftung: xlabel, ylabel

```
1 \begin{tikzpicture}
2 \begin{axis}[xlabel=x-Achse ,ylabel=y-Achse]
3 \addplot {x^2 - x + 4};
4 \end{axis}
5 \end{tikzpicture}
```

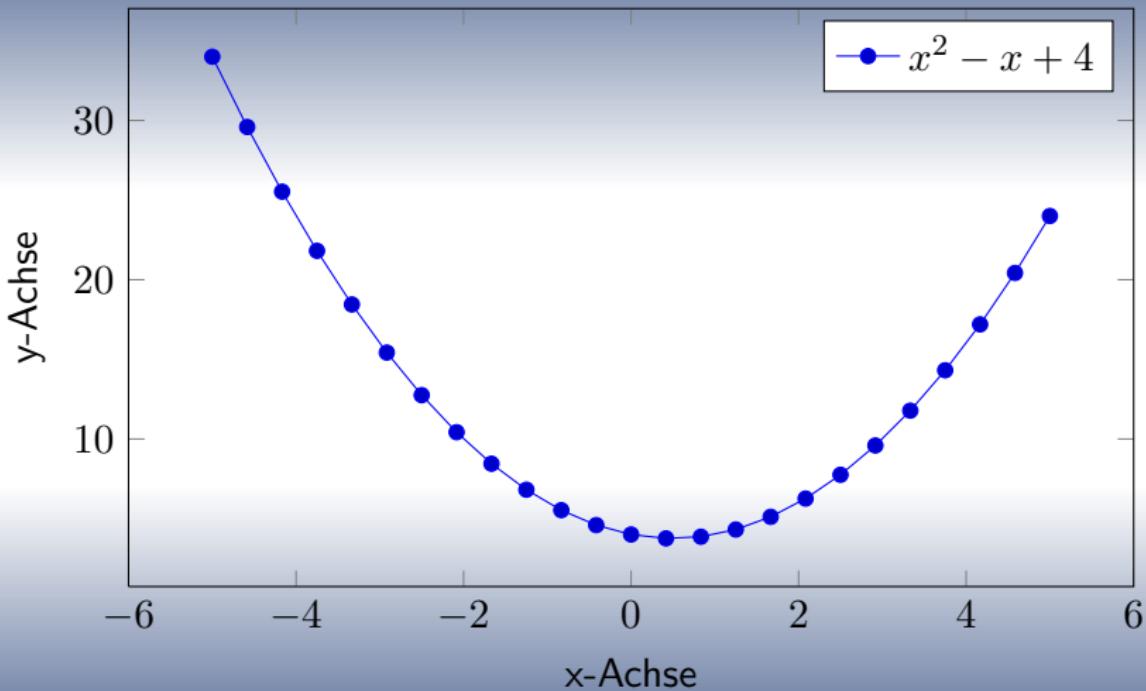
Achsenbeschriftung: xlabel, ylabel



Legende: \addlegendentry, \legend

```
1 \begin{tikzpicture}
2 \begin{axis}[ xlabel=x-Achse , ylabel=y-Achse]
3 \addplot {x^2 - x + 4};
4 \legend{$x^2 - x + 4$}
5 \end{axis}
6 \end{tikzpicture}
```

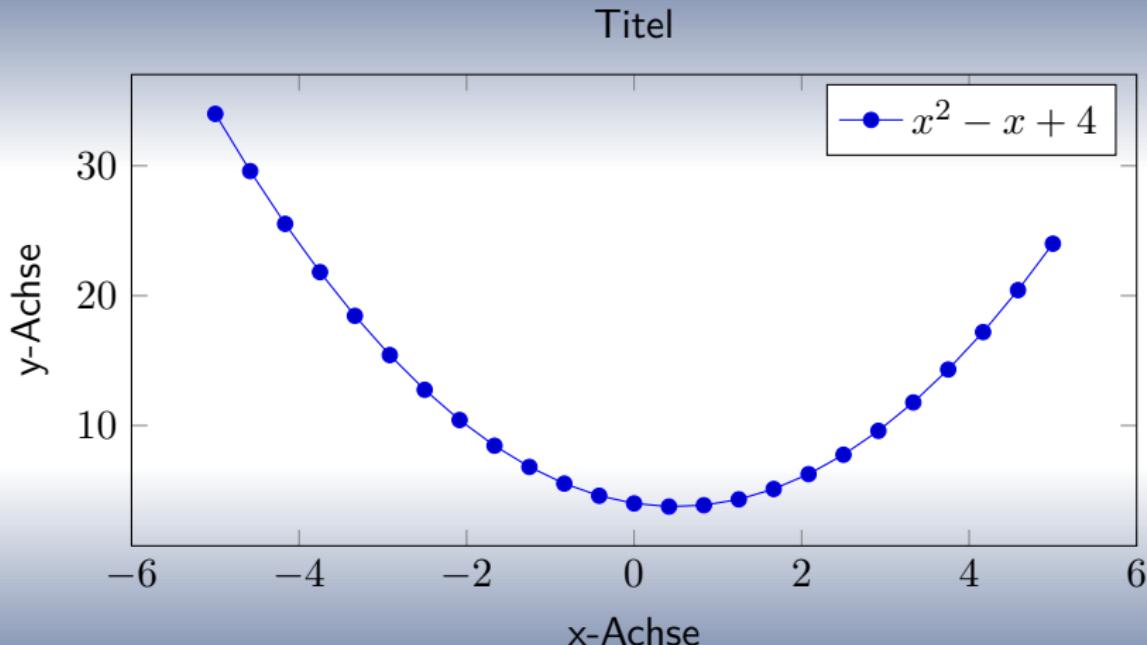
Legende: \addlegendentry, \legend



Titel: title

```
1 \begin{tikzpicture}
2 \begin{axis}[ xlabel=x-Achse , ylabel=y-Achse , title
   =Titel]
3 \addplot {x^2 - x + 4};
4 \legend{$x^2 - x + 4$}
5 \end{axis}
6 \end{tikzpicture}
```

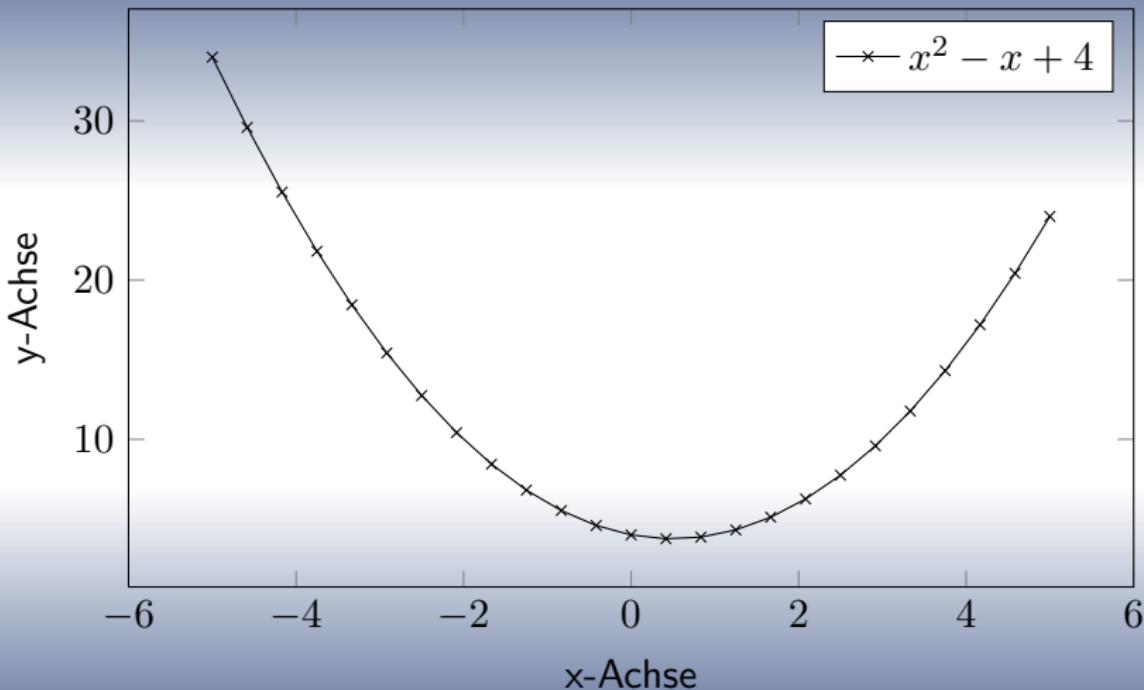
Titel: title



Markierung: mark

```
1 \begin{tikzpicture}
2 \begin{axis}[ xlabel=x-Achse , ylabel=y-Achse]
3 \addplot [mark=x] {x^2 - x + 4};
4 \legend{$x^2 - x + 4$}
5 \end{axis}
6 \end{tikzpicture}
```

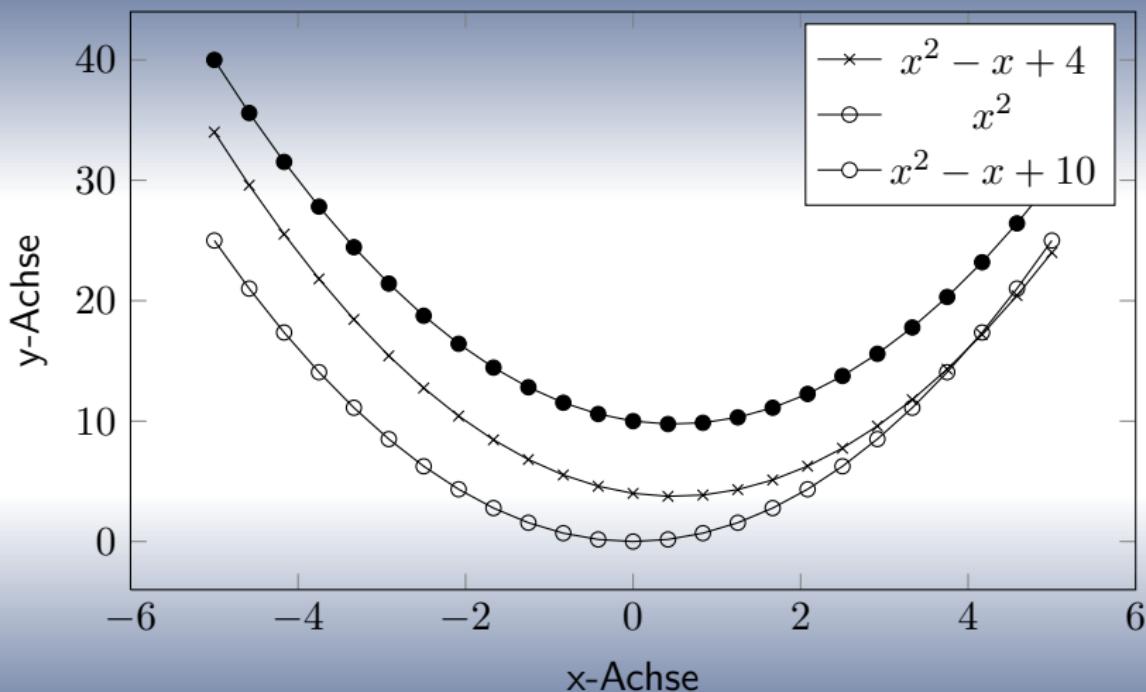
Markierung: mark



Mehrere Funktionen

```
1 \begin{tikzpicture}
2 \begin{axis}[ xlabel=x-Achse , ylabel=y-Achse]
3 \addplot [mark=x] {x^2 - x + 4};
4 \addlegendentry{$x^2 - x + 4$}
5 \addplot [mark=o] {x^2};
6 \addlegendentry{$x^2$}
7 \addplot [mark=**] {x^2 -x + 10};
8 \addlegendentry{$x^2 -x + 10$}
9 \end{axis}
10 \end{tikzpicture}
```

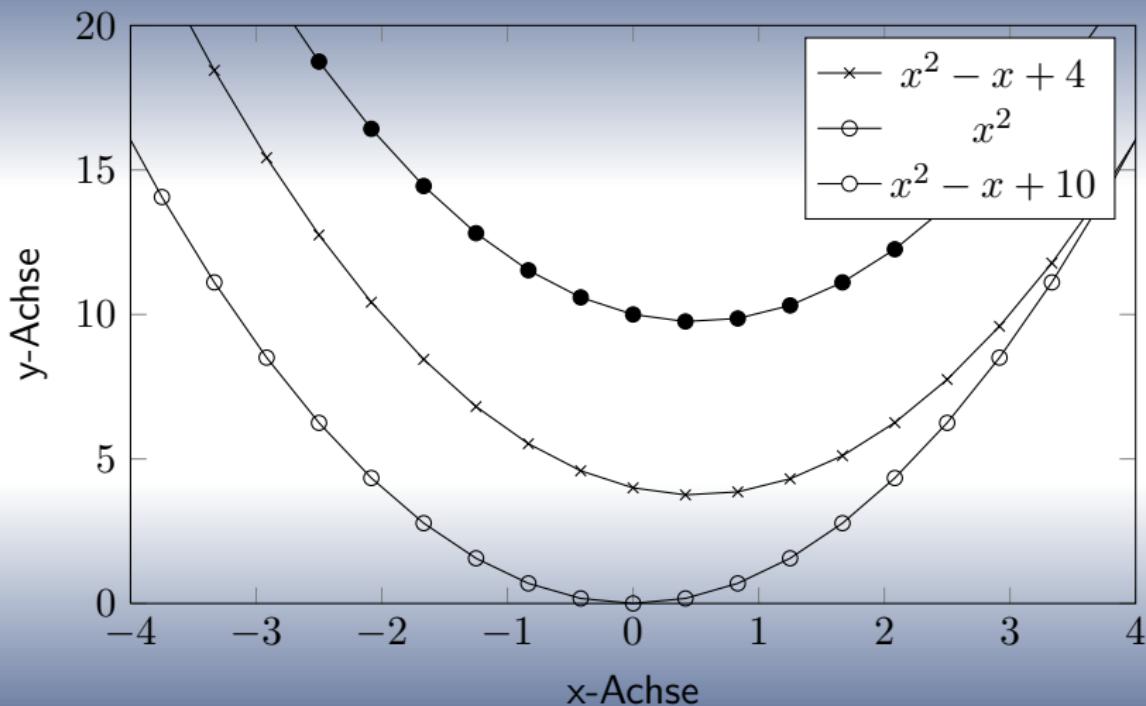
Mehrere Funktionen



Achsenformatierung: ymin, ymax, xmin, xmax

```
1 \begin{tikzpicture}
2 \begin{axis}[ xlabel=x-Achse , ylabel=y-Achse , ymin
3   =0, ymax=20, xmin=-4, xmax=4]
4 \addplot [mark=x] {x^2 - x + 4};
5 \addlegendentry{$x^2 - x + 4$}
6 \addplot [mark=o] {x^2};
7 \addlegendentry{$x^2$}
8 \addplot [mark=*, dashed] {x^2 - x + 10};
9 \addlegendentry{$x^2 - x + 10$}
10 \end{axis}
11 \end{tikzpicture}
```

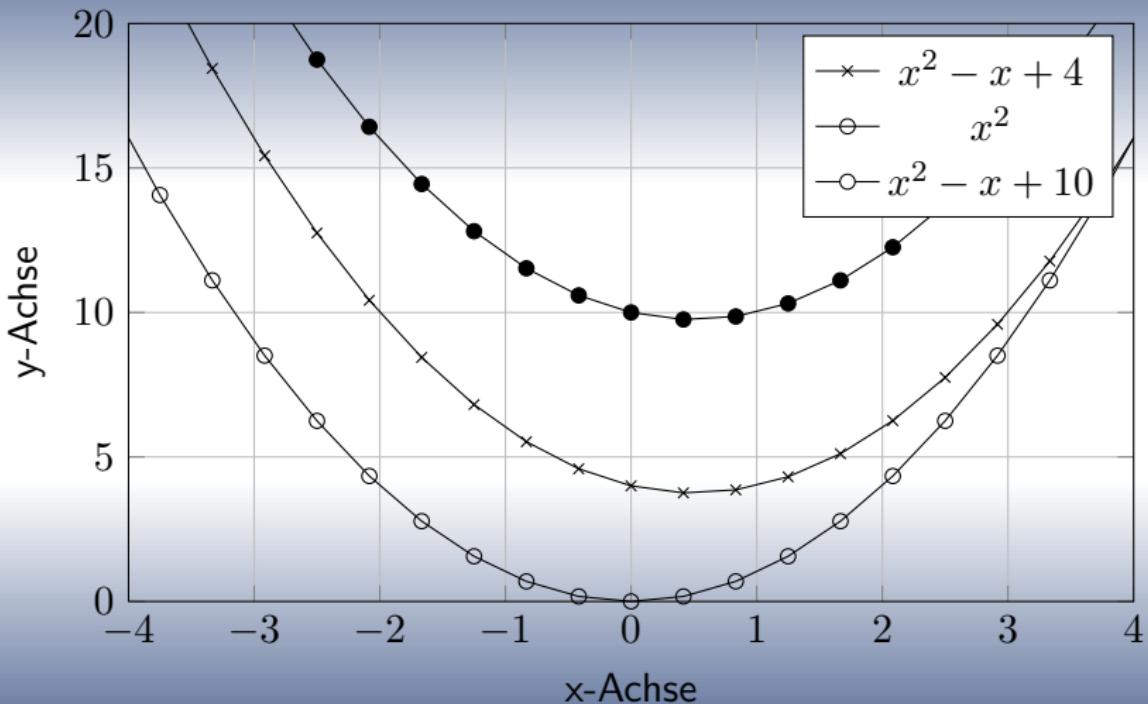
Achsenformatierung: ymin, ymax, xmin, xmax



Gitter anzeigen: grid

```
1 \begin{tikzpicture}
2 \begin{axis}[ xlabel=x-Achse , ylabel=y-Achse , ymin
=0 , ymax=20 , xmin=-4 , xmax=4 , grid=major]
3 \addplot [mark=x] {x^2 - x + 4};
4 \addlegendentry{$x^2 - x + 4$}
5 \addplot [mark=o] {x^2};
6 \addlegendentry{$x^2$}
7 \addplot [mark=*, dashed] {x^2 - x + 10};
8 \addlegendentry{$x^2 - x + 10$}
9 \end{axis}
10 \end{tikzpicture}
```

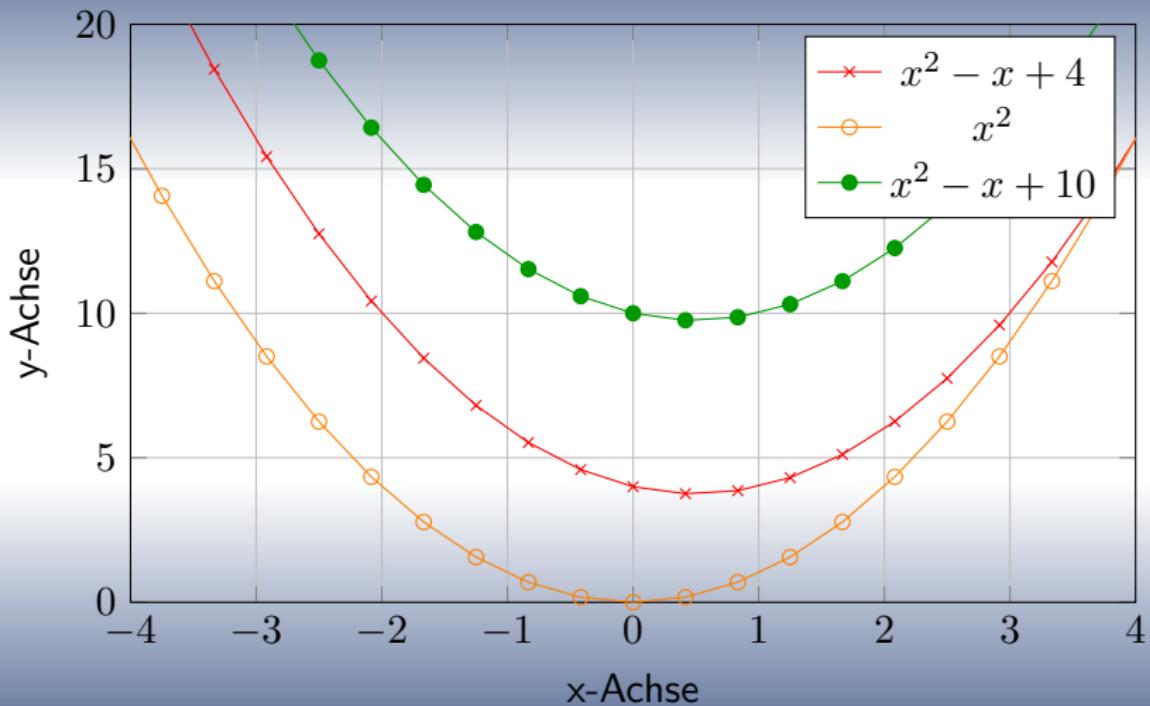
Gitter anzeigen: grid



Farbige Diagramme: color

```
1 \begin{tikzpicture}
2   \begin{axis}[ xlabel=x-Achse , ylabel=y-Achse , ymin =0 , ymax=20 , xmin=-4 , xmax=4 , grid=major]
3     \addplot [mark=x, color=red] {x^2 - x + 4};
4     \addlegendentry{$x^2 - x + 4$}
5     \addplot [mark=o, color=orange] {x^2};
6     \addlegendentry{$x^2$}
7     \addplot [mark=*, color=green!60!black] {x^2 -x + 10};
8     \addlegendentry{$x^2 -x + 10$}
9   \end{axis}
10 \end{tikzpicture}
```

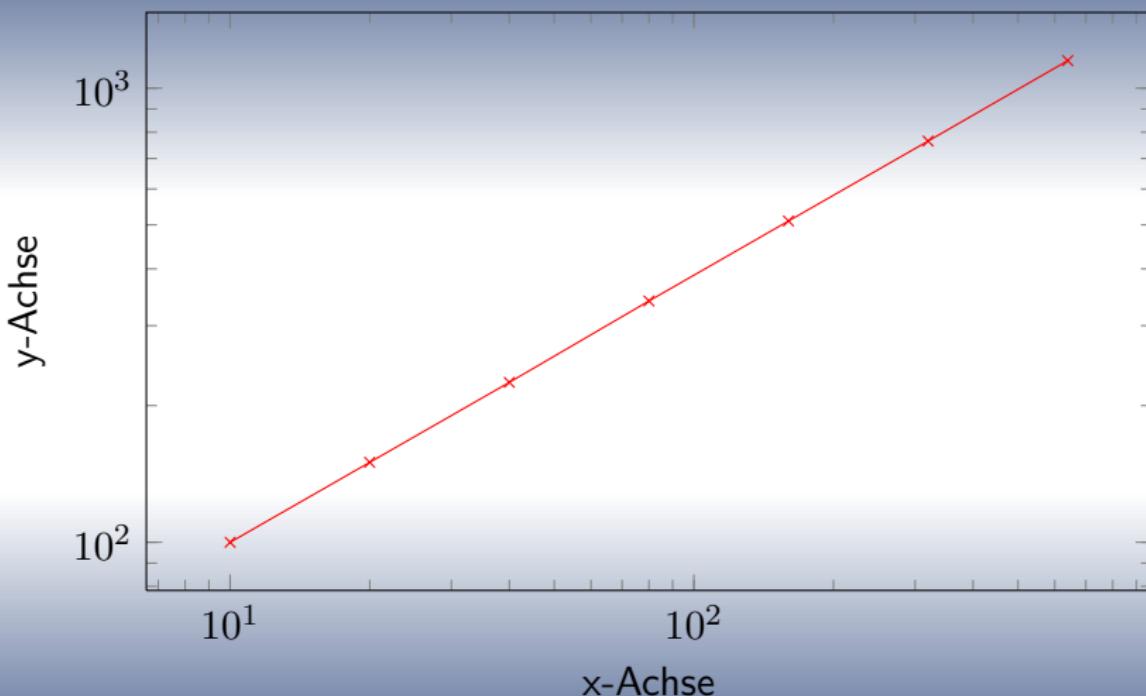
Farbige Diagramme: color



Logarithmische Diagramme

```
1 \begin{tikzpicture}
2   \begin{loglogaxis}[ xlabel=x-Achse , ylabel=y-Achse
3     ]
4     \addplot [color=red,mark=x] coordinates {
5       (10 ,100)
6       (20 ,150)
7       (40 ,225)
8       (80 ,340)
9       (160 ,510)
10      (320 ,765)
11      (640 ,1150)
12    };
13  \end{loglogaxis}
14 \end{tikzpicture}
```

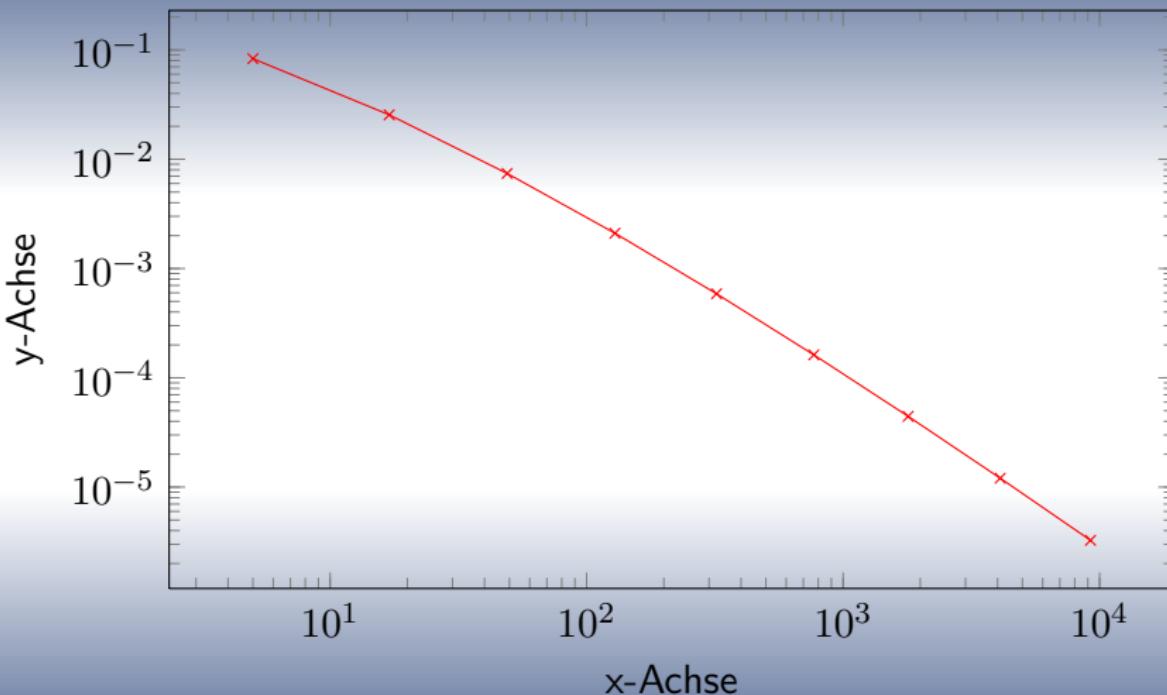
Logarithmische Diagramme



Logarithmische Diagramme

```
1 \begin{tikzpicture}
2 \begin{loglogaxis}[ xlabel=x-Achse , ylabel=y-Achse
    ]
3 \addplot [color=red,mark=x] coordinates {
4 (5, 8.31160034e-02)
5 (17, 2.54685628e-02)
6 (49, 7.40715288e-03)
7 (129, 2.10192154e-03)
8 (321, 5.87352989e-04)
9 (769, 1.62269942e-04)
10 (1793, 4.44248889e-05)
11 (4097, 1.20714122e-05)
12 (9217, 3.26101452e-06)
13 };
14 \end{loglogaxis}
15 \end{tikzpicture}
```

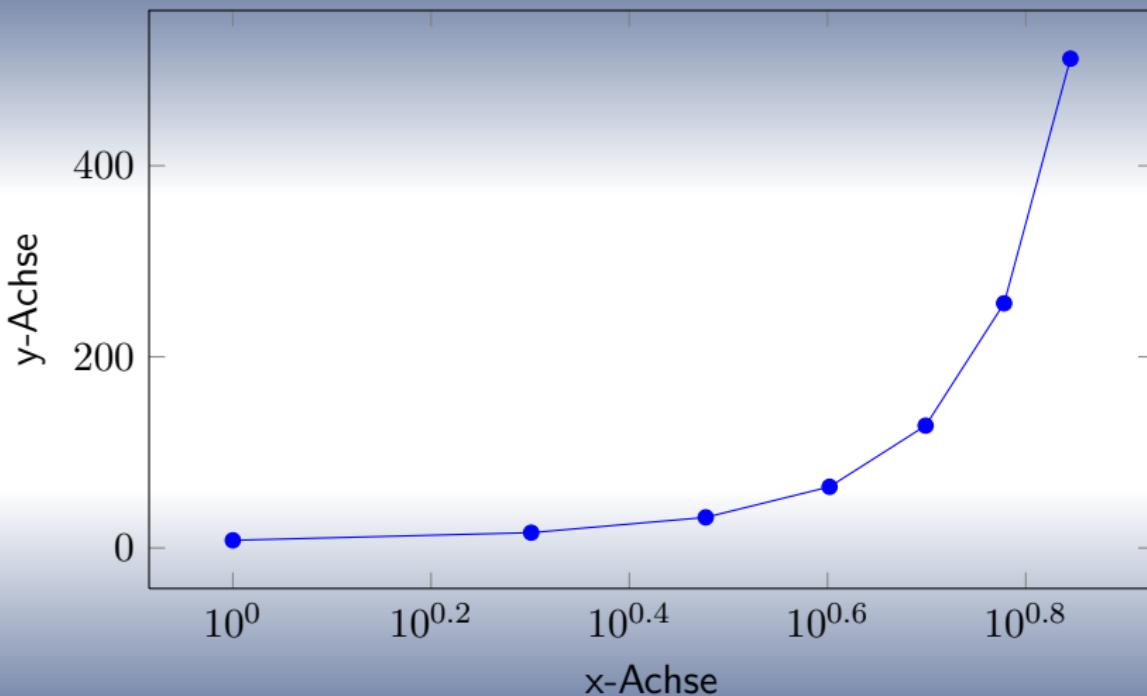
Logarithmische Diagramme



Semi-logarithmische Diagramme

```
1 \begin{tikzpicture}
2   \begin{semilogxaxis}[ xlabel=x-Achse , ylabel=y-
    Achse]
3     \addplot [color=blue,mark=*] coordinates {
4       (1,8)
5       (2,16)
6       (3,32)
7       (4,64)
8       (5,128)
9       (6,256)
10      (7,512)
11    };
12   \end{semilogxaxis}
13 \end{tikzpicture}
```

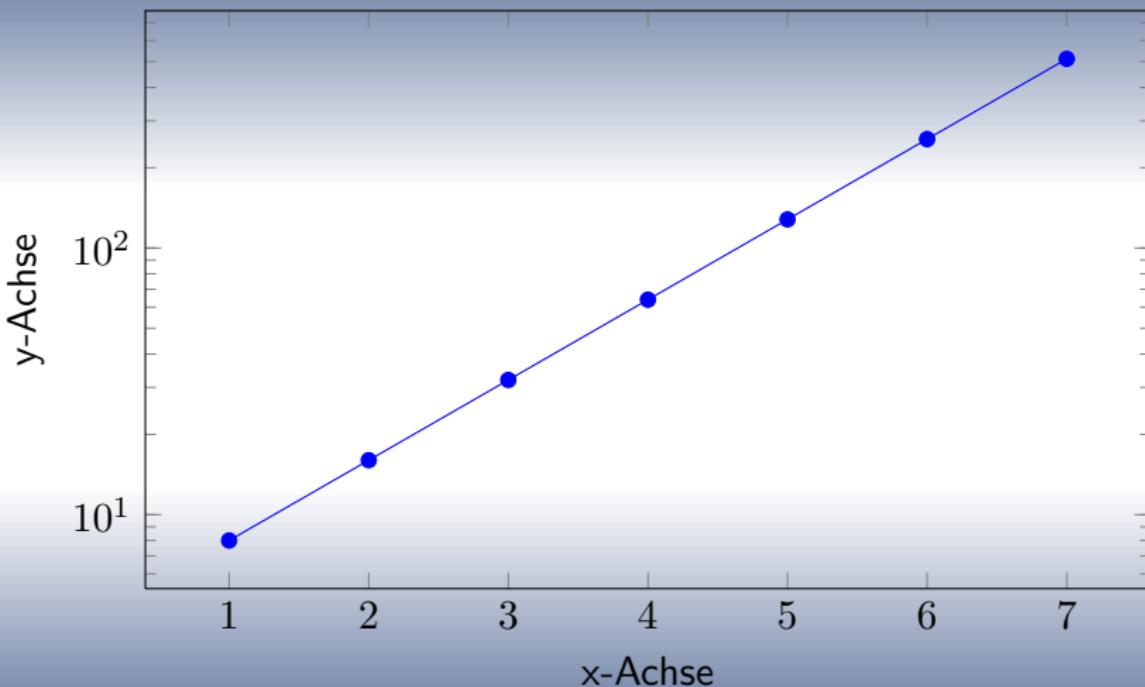
Semi-logarithmische Diagramme



Semi-logarithmische Diagramme

```
1 \begin{tikzpicture}
2   \begin{semilogyaxis}[ xlabel=x-Achse , ylabel=y-
    Achse]
3     \addplot [color=blue,mark=*] coordinates {
4       (1,8)
5       (2,16)
6       (3,32)
7       (4,64)
8       (5,128)
9       (6,256)
10      (7,512)
11    };
12   \end{semilogyaxis}
13 \end{tikzpicture}
```

Semi-logarithmische Diagramme



Externe Daten plotten: table

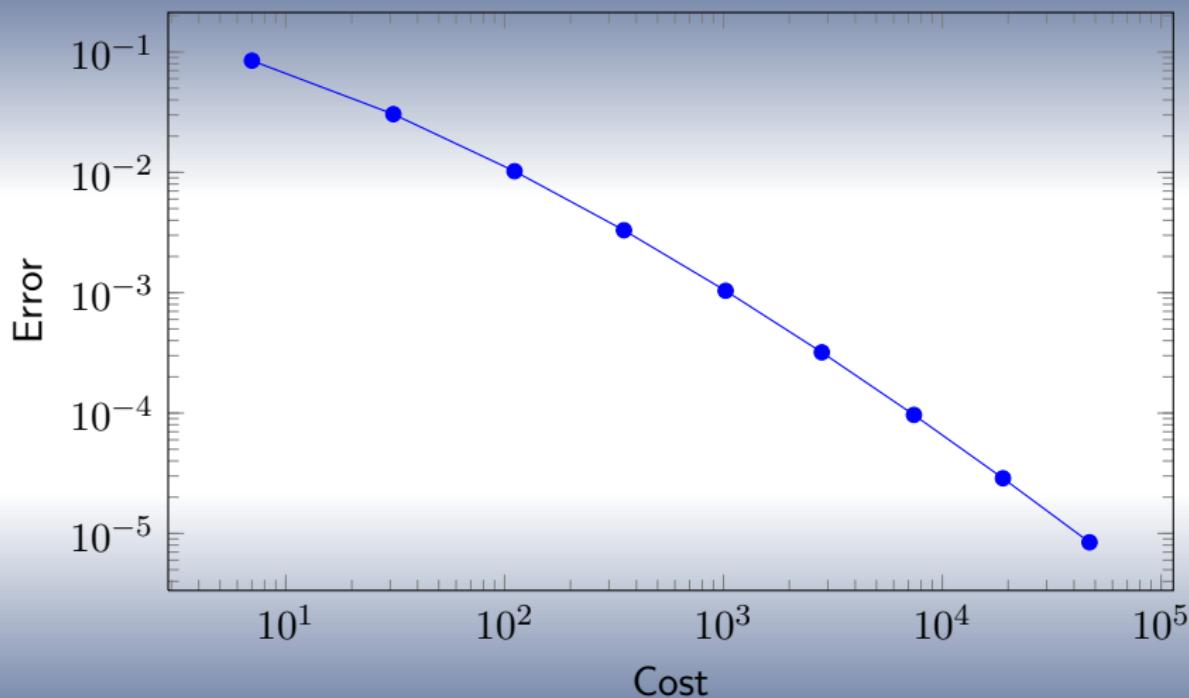
```
1 \begin{tikzpicture}
2 \begin{loglogaxis}[ xlabel=Cost , ylabel=Error]
3 \addplot [color=blue,mark=*] table [x=Cost ,y=Error
    ] {plotdata/table.txt};
4 \end{loglogaxis}
5 \end{tikzpicture}
```

Externe Daten plotten: table

Inhalt der Datei table.txt:

Level	Cost	Error
1	7	8.471e-02
2	31	3.044e-02
3	111	1.022e-02
4	351	3.303e-03
5	1023	1.038e-03
6	2815	3.196e-04
7	7423	9.657e-05
8	18943	2.873e-05
9	47103	8.437e-06

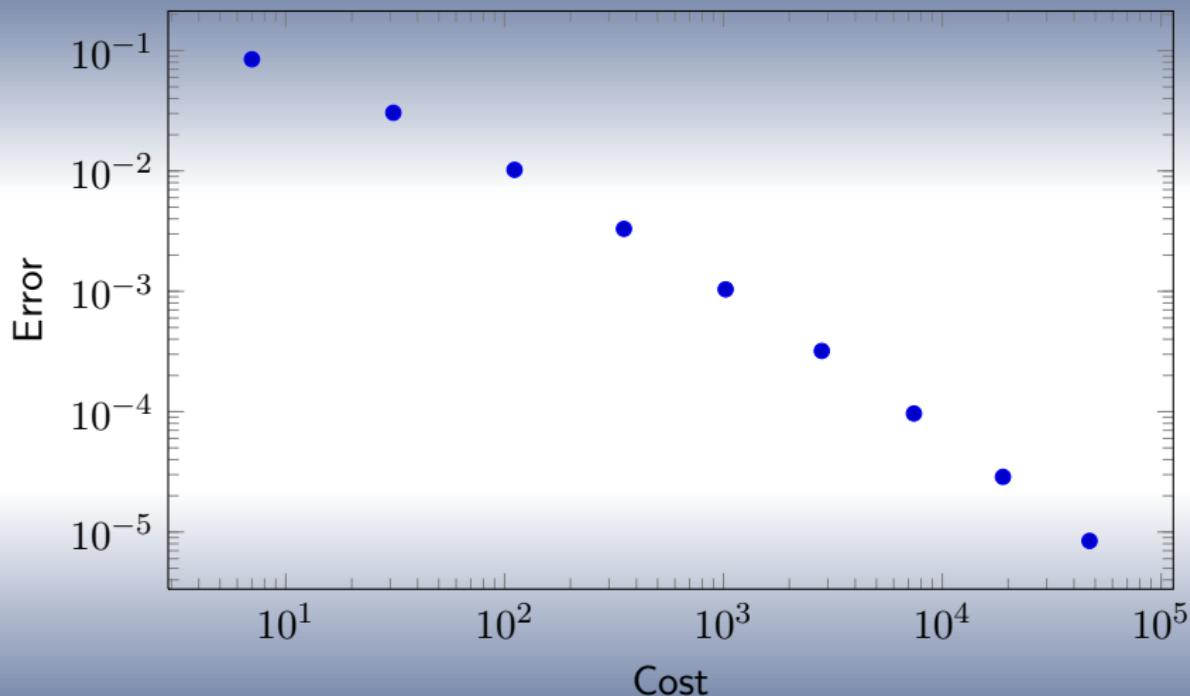
Externe Daten plotten: table



Datenpunkte darstellen: only marks

```
1 \begin{tikzpicture}
2 \begin{loglogaxis}[ xlabel=Cost , ylabel=Error]
3 \addplot+[color=blue,only marks] table [x=Cost ,y=
    Error] {plotdata/table.txt};
4 \end{loglogaxis}
5 \end{tikzpicture}
```

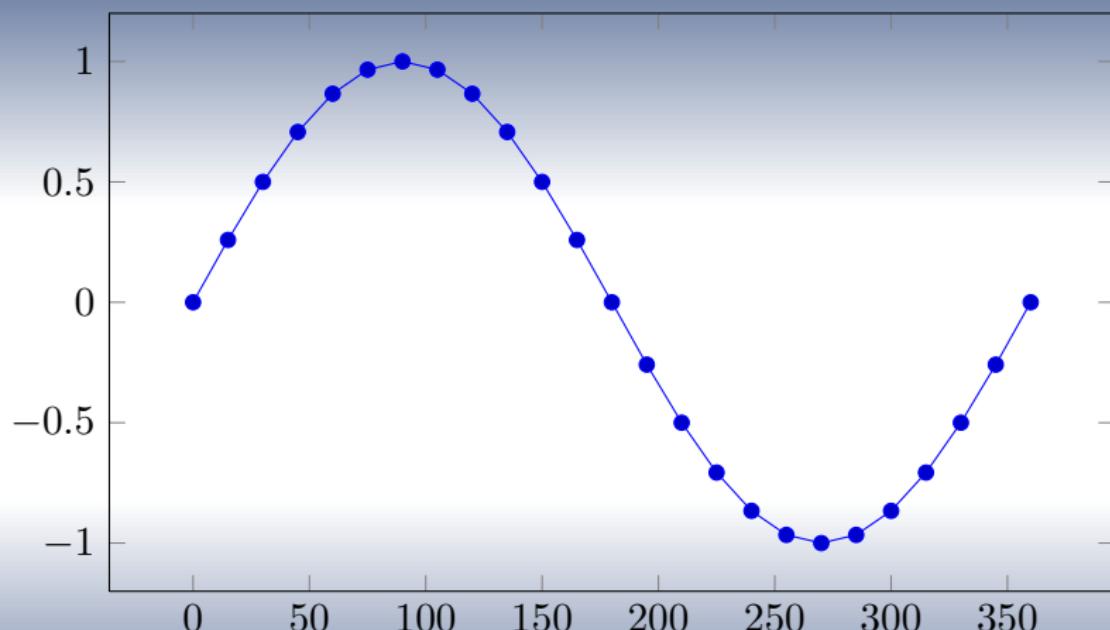
Datenpunkte darstellen: only marks



Bereich ändern: domain

```
1 \begin{tikzpicture}
2 \begin{axis}
3 \addplot+[domain=0:360] {sin(x)};
4 \end{axis}
5 \end{tikzpicture}
```

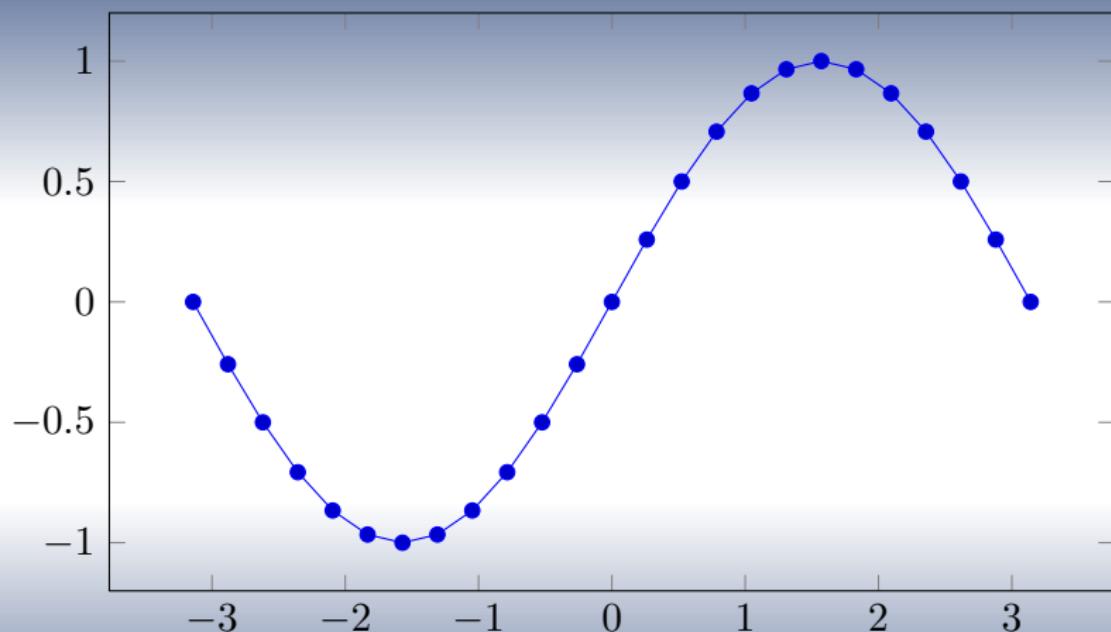
Bereich ändern: domain



Bereich ändern: domain

```
1 \begin{tikzpicture}
2 \begin{axis}
3 \addplot+[domain=-pi:pi] {sin(deg(x))};
4 \end{axis}
5 \end{tikzpicture}
```

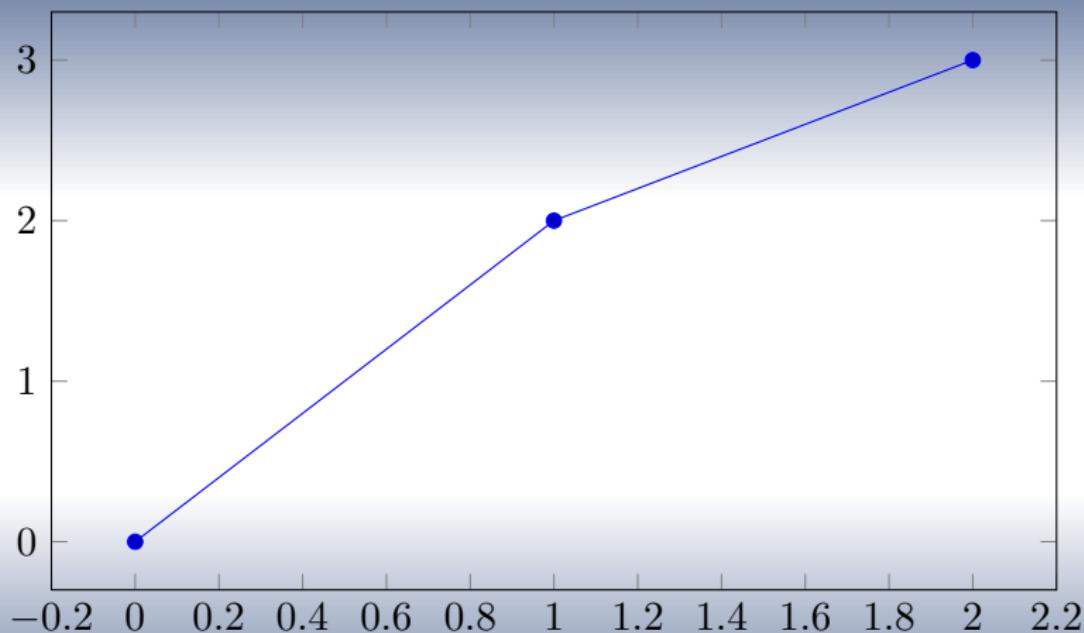
Bereich ändern: domain



Lineare Diagramme: sharp plot

```
1 \begin{tikzpicture}
2 \begin{axis}
3 \addplot+[sharp plot] coordinates {(0,0) (1,2)
4 (2,3)};
5 \end{axis}
6 \end{tikzpicture}
```

Lineare Diagramme: sharp plot



Weiche Diagramme: smooth

```
1 \begin{tikzpicture}
2 \begin{axis}
3 \addplot+[smooth] coordinates {(0,0) (1,2) (2,3)}
4   ;
5 \end{axis}
6 \end{tikzpicture}
```

Weiche Diagramme: smooth

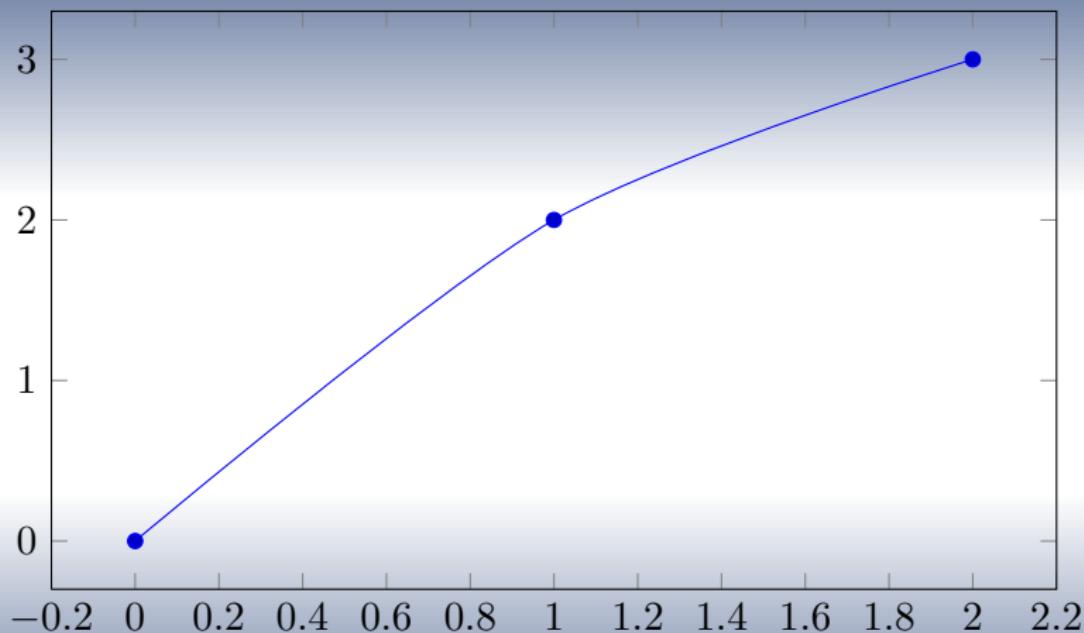
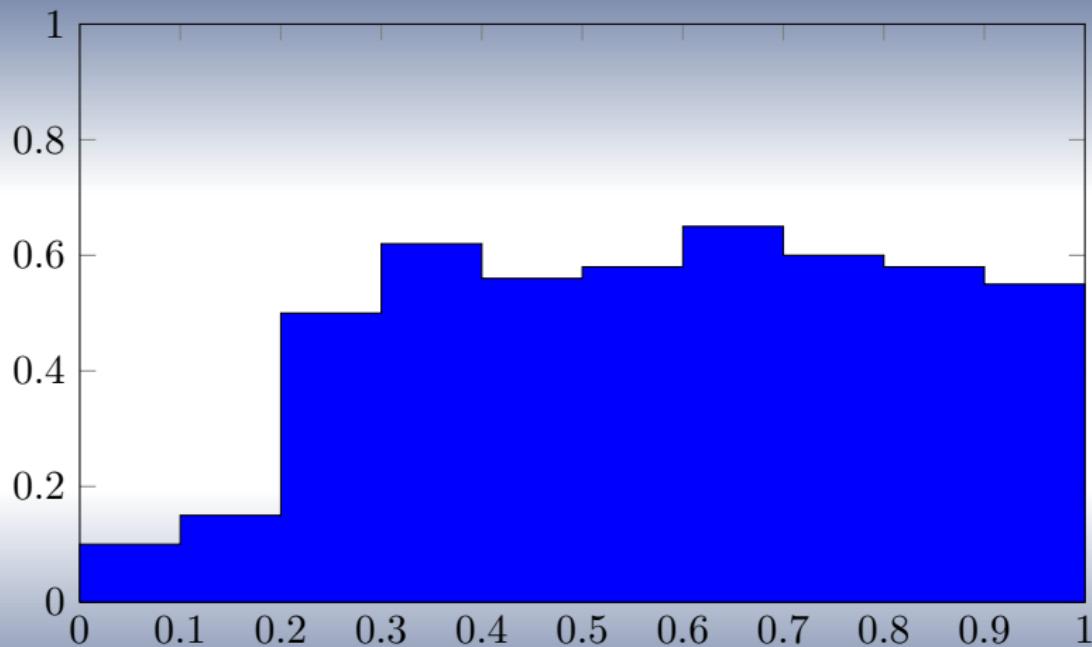


Diagramme füllen: fill

```
1 \begin{tikzpicture}
2 \begin{axis}[ymin=0,ymax=1,enlargelimits=false]
3 \addplot
4 [const plot,fill=blue,draw=black]
5 coordinates
6 {(0,0.1) (0.1,0.15) (0.2,0.5) (0.3,0.62)
7 (0.4,0.56) (0.5,0.58) (0.6,0.65) (0.7,0.6)
8 (0.8,0.58) (0.9,0.55) (1,0.52)}
9 \closedcycle;
10 \end{axis}
11 \end{tikzpicture}
```

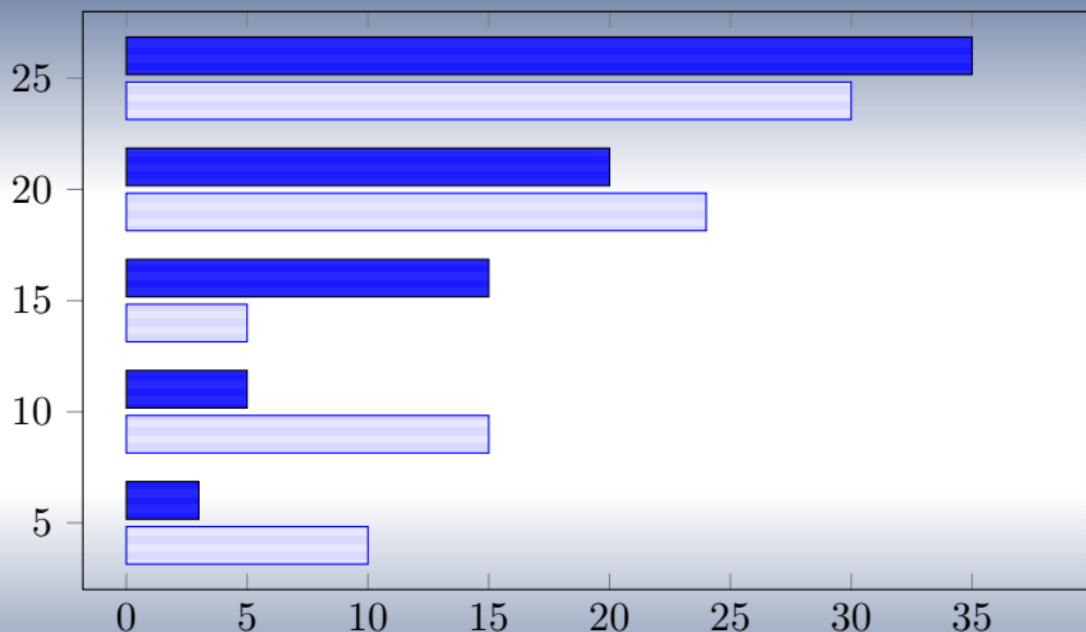
Diagramme füllen: fill



Balkendiagramme: xbar

```
1 \begin{tikzpicture}
2   \begin{axis}[xbar, enlargelimits=0.15]
3     \addplot
4       [draw=blue, pattern=horizontal lines light blue]
5       coordinates
6       {(10,5) (15,10) (5,15) (24,20) (30,25)};
7     \addplot
8       [draw=black, pattern=horizontal lines dark blue]
9       coordinates
10      {(3,5) (5,10) (15,15) (20,20) (35,25)};
11   \end{axis}
12 \end{tikzpicture}
```

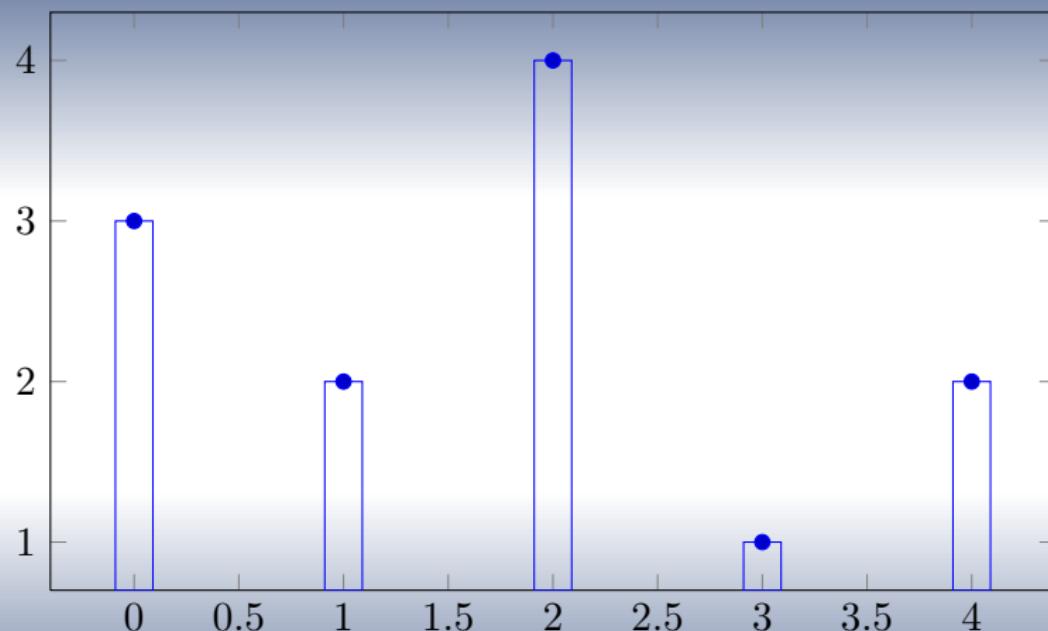
Balkendiagramme: xbar



Balkendiagramme: ybar

```
1 \begin{tikzpicture}
2   \begin{axis}
3     \addplot+[ybar] plot coordinates {(0,3) (1,2)
4                                     (2,4) (3,1) (4,2)};
5   \end{axis}
6 \end{tikzpicture}
```

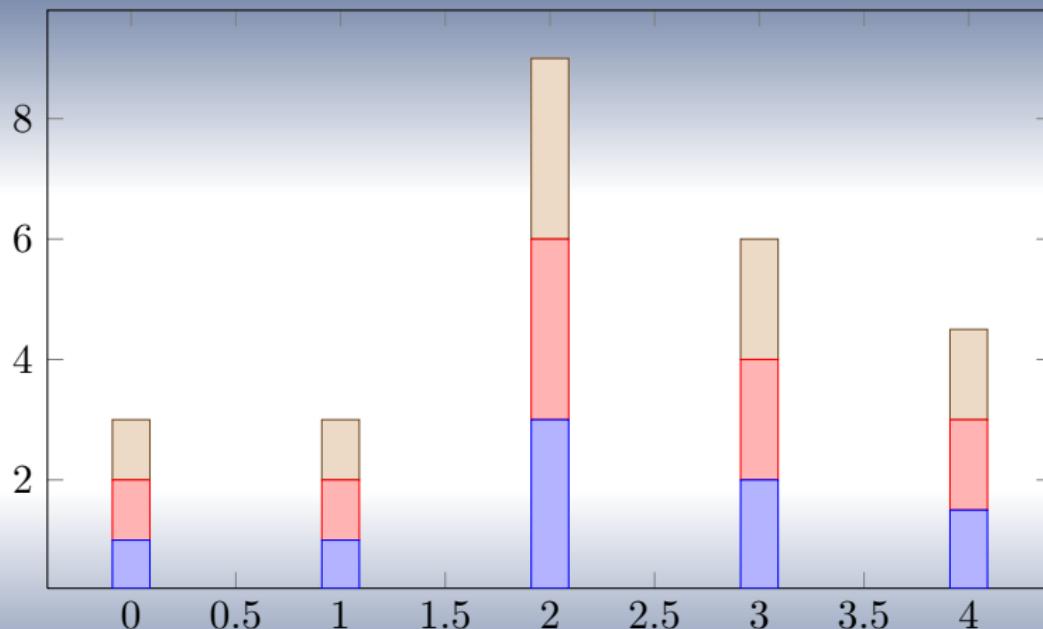
Balkendiagramme: ybar



Stapeldiagramme: ybar stacked

```
1 \begin{tikzpicture}
2   \begin{axis}[ybar stacked]
3     \addplot coordinates
4     {(0,1) (1,1) (2,3) (3,2) (4,1.5)};
5     \addplot coordinates
6     {(0,1) (1,1) (2,3) (3,2) (4,1.5)};
7     \addplot coordinates
8     {(0,1) (1,1) (2,3) (3,2) (4,1.5)};
9   \end{axis}
10 \end{tikzpicture}
```

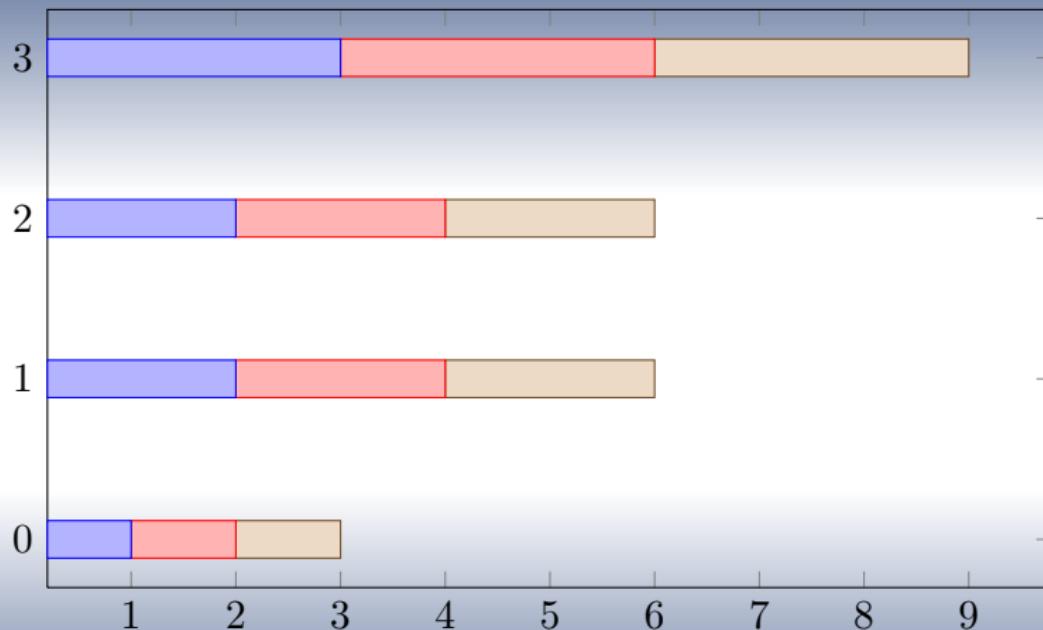
Stapeldiagramme: `ybar stacked`



Stapeldiagramme: xbar stacked

```
1 \begin{tikzpicture}
2   \begin{axis}[xbar stacked]
3     \addplot coordinates
4     {(1,0) (2,1) (2,2) (3,3)};
5     \addplot coordinates
6     {(1,0) (2,1) (2,2) (3,3)};
7     \addplot coordinates
8     {(1,0) (2,1) (2,2) (3,3)};
9   \end{axis}
10 \end{tikzpicture}
```

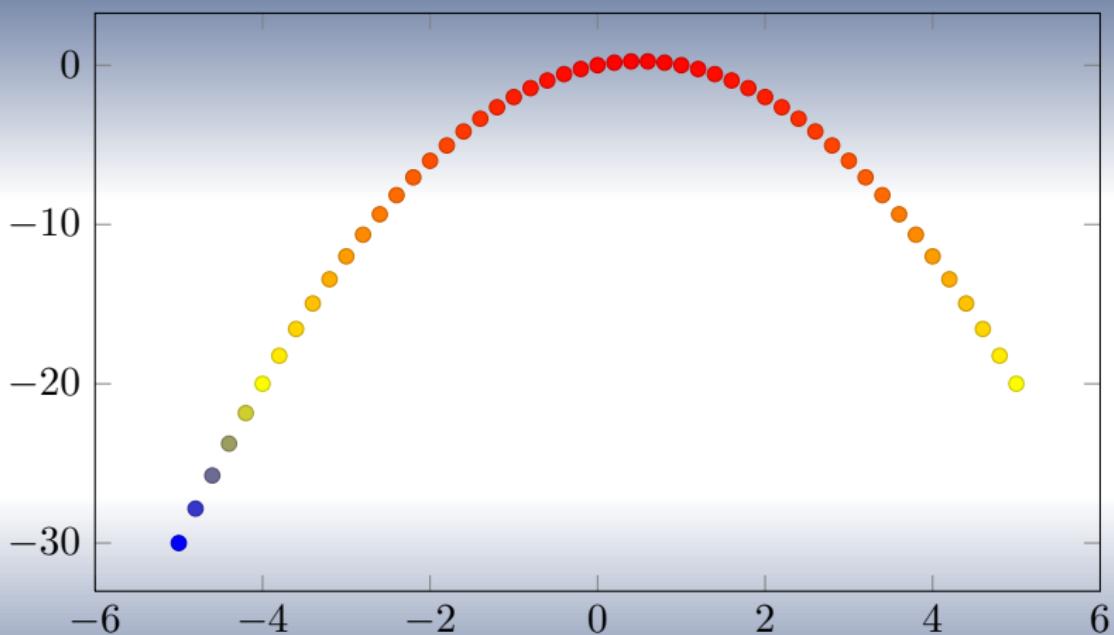
Stapeldiagramme: xbar stacked



Streudiagramme: scatter

```
1 \begin{tikzpicture}
2 \begin{axis}
3 \addplot+[scatter,only marks,
4 samples=50,scatter src=y]
5 {x-x^2};
6 \end{axis}
7 \end{tikzpicture}
```

Streudiagramme: scatter



3D-Diagramme mit pgfplots

- Erstellung von Linien-, Streu-, Netz- und Oberflächendiagrammen
- Vorteil: Konsistenz bei Schriftart/Schriftgröße/Grafiken, Mathematisksatz von \TeX , Präambel-Einstellungen werden übernommen
- Nachteil: 3D-Funktion von pgfplots sehr langsam, Auflösung beschränkt

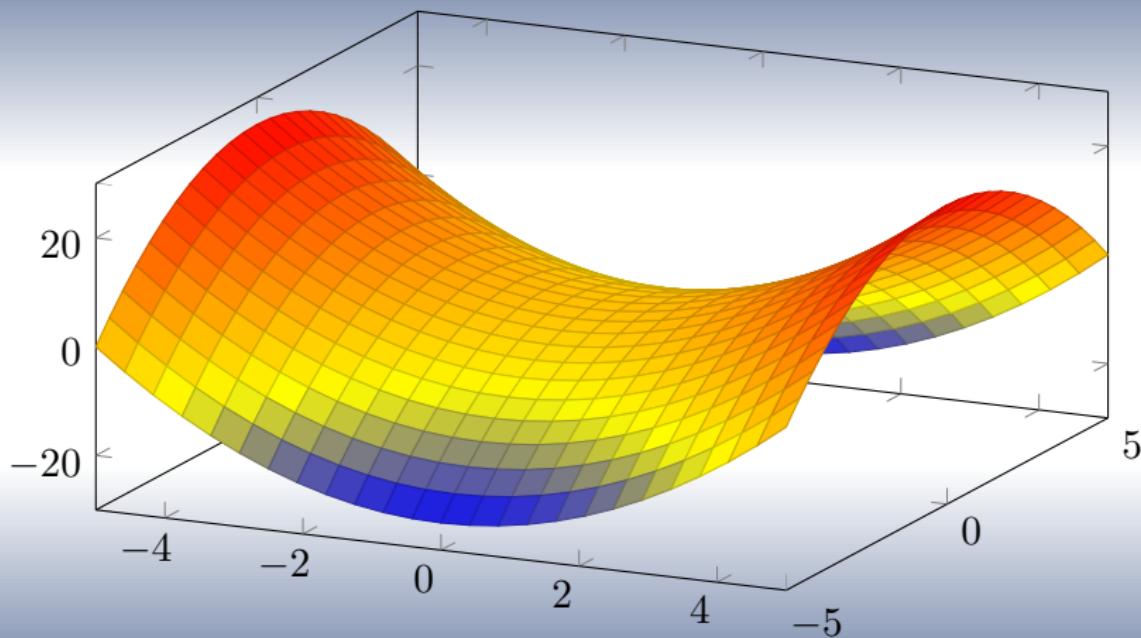
3D-Diagramme: \addplot3

```
\begin{tikzpicture} [Optionen]
\begin{axis} [Optionen]
...
\addplot3 [Optionen] ...;
...
\end{axis}
\end{tikzpicture}
```

Funktionenplot

```
1 \begin{tikzpicture}
2 \begin{axis}
3 \addplot3[surf] {x^2 - y^2};
4 \end{axis}
5 \end{tikzpicture}
```

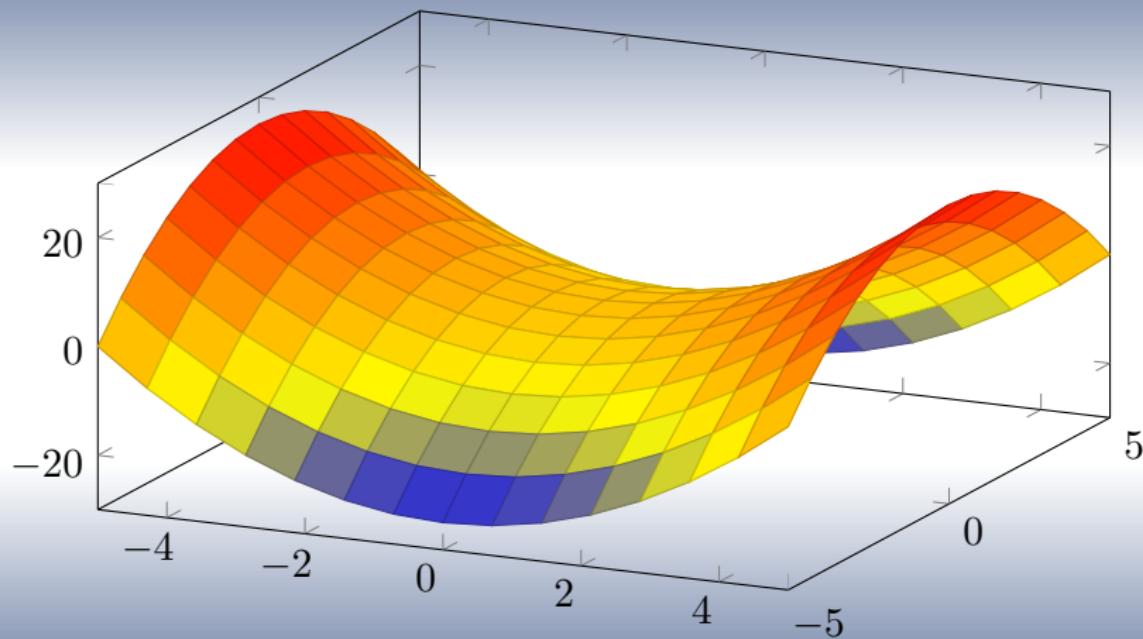
Funktionenplot



Auflösung: samples

```
1 \begin{tikzpicture}
2 \begin{axis}
3 \addplot3[surf, samples=15] {x^2 - y^2};
4 \end{axis}
5 \end{tikzpicture}
```

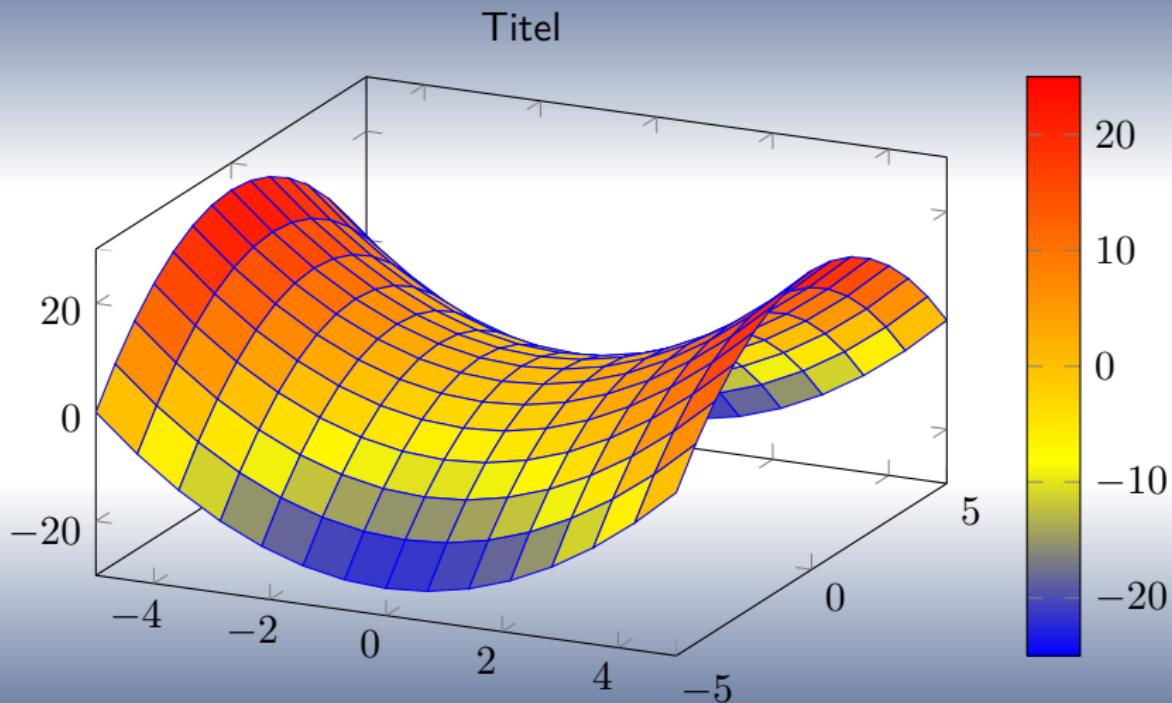
Auflösung: samples



Legende und Titel: colorbar,title

```
1 \begin{tikzpicture}
2   \begin{axis}[colorbar,title=Titel]
3     \addplot3[surf,samples=15,faceted color=blue] {x
4       ^2 - y^2};
5   \end{axis}
6 \end{tikzpicture}
```

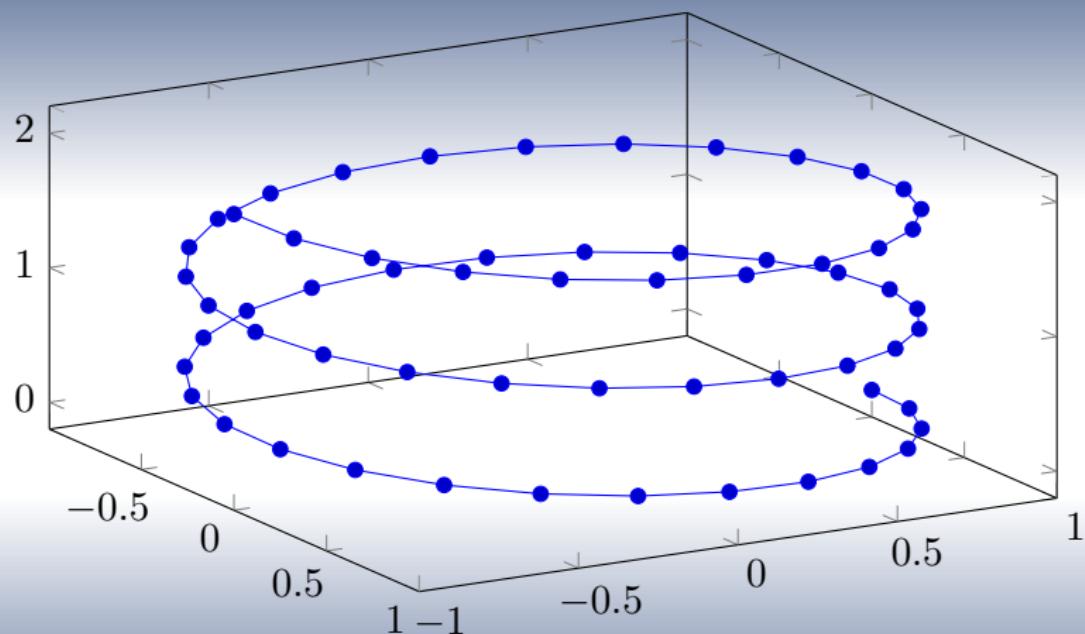
Legende und Titel: colorbar, title



Betrachtungswinkel: view

```
1 \begin{tikzpicture}
2   \begin{axis}[view={60}{30}]
3     \addplot {sin(deg(x))};
4     \addplot {cos(deg(x))};
5     \addplot {2*x/(5*pi)};
6   \end{axis}
7 \end{tikzpicture}
```

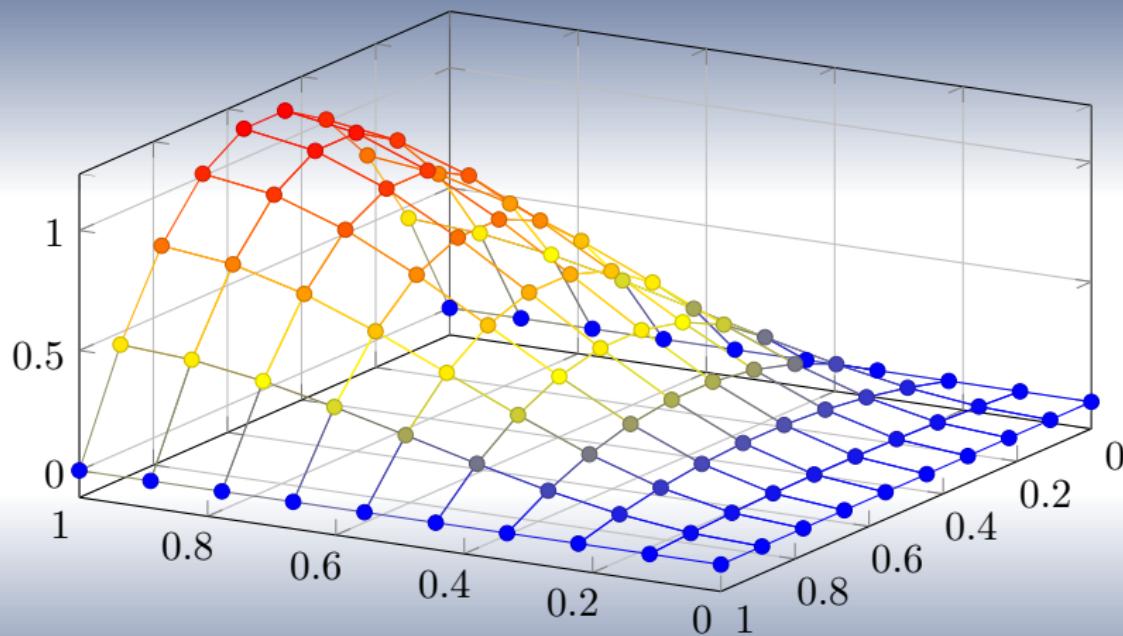
Betrachtungswinkel: view



Netzdiagramme: mesh

```
1 \begin{tikzpicture}
2 \begin{axis}[grid=major ,view={210}{30}]
3 \addplot3+[mesh,scatter,samples=10,domain=0:1]
4 {5*x*sin(2*deg(x)) * y*(1-y)};
5 \end{axis}
6 \end{tikzpicture}
```

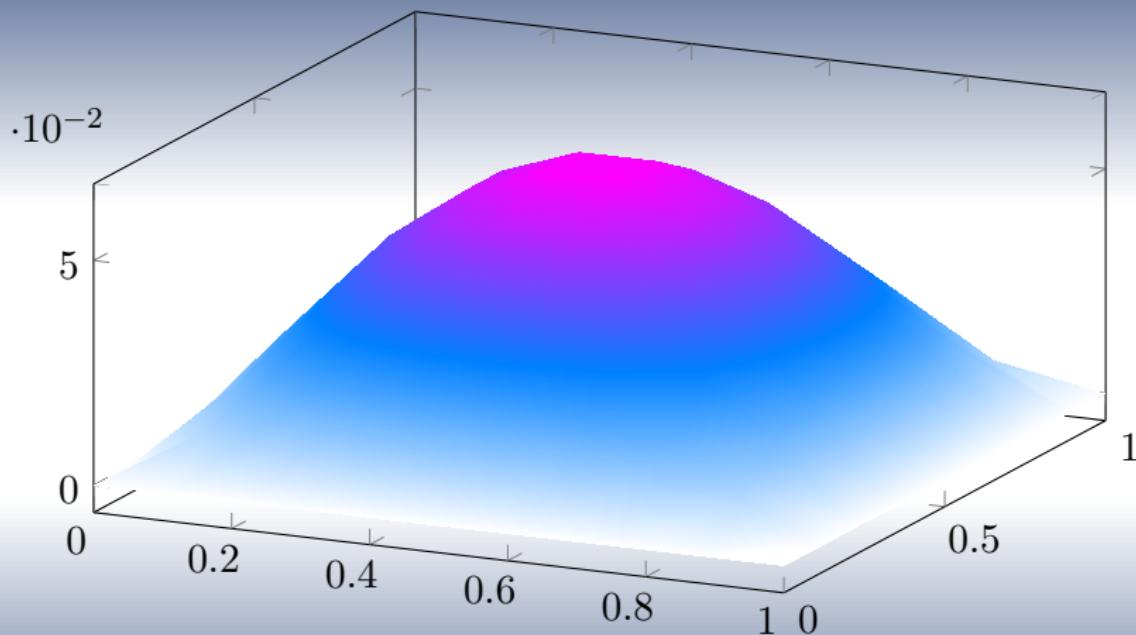
Netzdiagramme: mesh



Shader: shader

```
1 \begin{tikzpicture}
2 \begin{axis}[colormap/cool]
3 \addplot3[surf, samples=10, domain=0:1,
4 shader=interp]
5 {x*(1-x)*y*(1-y)};
6 \end{axis}
7 \end{tikzpicture}
```

Shader: shader



Was ist PSTricks?

- Projektseite: <http://tug.org/PSTricks/main.cgi/>
- Basiert auf PostScript (→ Programmiersprache)
- Ansammlung von Paketen
- Vorteile: Pfade (z. B. 3D-Buchstaben), Knoten verbinden, Berechnungen werden an PS Interpreter übergeben
- Nachteile: Syntax meist komplizierter, keine gemeinsame Dokumentation

Ausgewählte PSTricks Pakete

- `pst-3dplot`: 3D-Diagramme und 3D-Körper
- `pst-circ`: Schaltungen
- `pst-func`: Funktionenplots
- `pst-map2d`: 2D-Projektionen der Erde
- `pst-map3d`: 3D-Projektionen der Erde
- `pst-node`: Knotenpunkte
- `pst-plot`: Funktionenplots
- `pst-solides3d`: 3D-Körper
- `pst-text`: Textgestaltung

Übung 7b

- Funktionenplot
- Breite, Titel, Legende, Markierung, Linienfarbe, Gitter
- Achsenformatierung
- Mehrere Funktionen
- Semi-logarithmisches Diagramm
- Logarithmischer Datenplot
- 3D-Diagramm: Oberflächendiagramm

Nächster Termin

27.09.10: Einführung

28.09.10: KOMA-Script, Mathematisatz

29.09.10: Gleitobjekte (Abbildungen, Tabellen), Zusatzpakete

30.09.10: Verzeichnisse (Bibliografie, Index, Glossar), Grafiken

01.10.10: Präsentationen

Literatur

Feuersänger, Christian (2010). **Manual for Package pgfplots**.

Version 1.31. URL:

<http://tug.ctan.org/tex-archive/graphics/pgf/contrib/pgfplots/doc/latex/pgfplots/pgfplots.pdf>.

Tantau, Till (2007). **The TikZ and PGF Packages**. Version 2.00. URL:

<http://tug.ctan.org/tex-archive/graphics/pgf/base/doc/generic/pgf/pgfmanual.pdf>.