

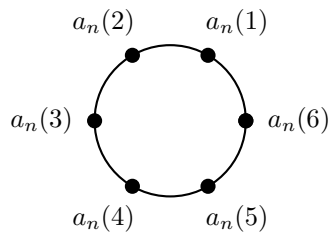
Inhaltsverzeichnis

1	Knoten	1
1.1	Kreis mit Beschriftung über Knoten und multido	1
1.2	Skizze mit pst-node erstellen	2
1.3	Dreieck mit pst-node	3
1.4	Dreieck mit Winkeln und Hilfslinien und pst-node	4
2	Euklidische Geometrie	5
2.1	Gleichschenkliges Dreieck mit \overline{AC} , \overline{AB} und Winkel BAC	5
2.2	Gleichschenkliges Dreieck mit A , C und Winkel ACB	6
2.3	Dreieck mit Winkeln und Hilfslinien und pst-eucl	7
2.4	Gleichseitiges Dreieck mit Innkreis	8
3	Funktionen	9
3.1	Funktion über algebraische Eingabe	9
3.2	Funktion mit pstricks-add	10
3.3	Funktion mit farbiger Fläche	11
3.4	Kurve mit Pfeilen, newpsobj, PS-Code	12
4	Diverses	13
4.1	Text auf Kurve	13

1 Knoten

1.1 Kreis mit Beschriftung über Knoten und multido

Es werden `multido` und `pst-node` benötigt.



Komplette Eingabe für Skizze

```

1 \psset{unit=1mm}
  \begin{pspicture}(-20,-15)(21,15)
3 \SpecialCoor \degrees[6]
  \multido{\i=0+1}{7}{\cnode*(10;\i){1mm}{P\i}}
5 \multido{\i=1+1}{6}{\nput{\i}{P\i}{$a_n(\i)$}}
  \pscircle{10mm}
7 \end{pspicture}

```

Erläuterungen

→ spezielle Koordinatenformen aktivieren, s. **Z.3**

Syntax: `\SpecialCoor`

- ① Knoten können nun als Koordinate verwendet werden

→ Kreis in 6 Teile einteilen, da nur 6 Punkte gebraucht werden, s. **Z.3**

Syntax: `\degrees[6]`

- ① `\degrees` $\hat{=}$ Vollkreis von 360°

→ Knoten festlegen:

Bsp Z.4: `\cnode*(10;n){1mm}{Pn}`

Syntax: `\cnode*(r;\alpha){Radius}{Knotenname}`

- ① `\cnode*` erzeugt ausgefüllten Punkt
- ② Polarkoordinaten werden mit Semikolon eingegeben

→ 6 Knoten festlegen mit `multido`:

Bsp Z.4: `\multido{\i=0+1}{7}{\cnode*(10;\i){1mm}{P\i}}`

Syntax: `\multido{Variable=Start+Schrittweite}{Wiederholungen}{Was soll wiederholt werden}`

- ① mit `\r` wird reeller, mit `\i` wird ganzer Wert als Variable definiert

→ Beschriftung mit `nput` gehört zu Paket `pst-node`

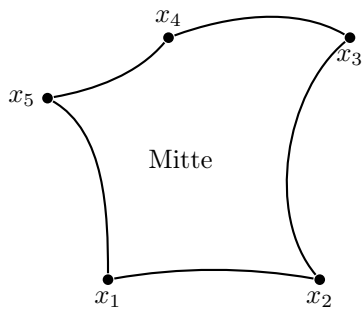
Bsp Z.5: `\nput{\i}{P\i}{$a_n(\i)$}`

Syntax: `\nput{Winkel}{Knotenname}{Bezeichnung}`

- ① `\nput` ist prinzipiell identitisch mit `\uput`, bezieht sich aber auf Knotenlabel

1.2 Skizze mit pst-node erstellen

Pakete `pstricks` und `pst-node` bieten sich an



Komplette Eingabe für Skizze

```

1 \psset{unit=.8cm}
2 \begin{pspicture}(-1,-.8)(5.5,5)\SpecialCoor
3 \cnode*(1,0){2pt}{A}\uput{.2}[-90](A){$x_1$}
4 \cnode*(4.5,0){2pt}{B}\uput{.2}[-90](B){$x_2$}
5 \cnode*(5,4){2pt}{C}\uput{.2}[-90](C){$x_3$}
6 \cnode*(2,4){2pt}{D}\uput{.2}[90](D){$x_4$}
7 \cnode*(0,3){2pt}{E}\uput{.2}[180](E){$x_5$}
8 \nccurve[angleA=10,angleB=170]{A}{B}
9 \nccurve[angleA=130,angleB=220]{B}{C}
10 \nccurve[angleA=150,angleB=20]{C}{D}
11 \nccurve[angleA=230,angleB=10]{D}{E}
12 \nccurve[angleA=330,angleB=90]{E}{A}
13 \rput(2.2,2){Mitte}
14 \end{pspicture}

```

Erläuterungen

→ 5 Knoten festgelegt:

Bsp Z.3: `\cnode*(1,0){2pt}{A}`

Syntax: `\cnode(x,y){Radius}{Knotenname}`

① ausgefüllter Punkt mit `\cnode*`

→ Knoten mit Label versehen:

Bsp Z.3: `\uput{.2}[-90](A){x_1}`

Syntax: `\uput{Labelabstand}[Winkel]{Drehung}(x,y){Labelname}`

① wenn Drehung nicht erforderlich, kann Wert entfallen

② wenn `SpecialCoor` verwendet wird, kann statt (x,y) (Knoten) verwendet werden

→ Knoten verbinden:

Bsp Z.8: `\nccurve[angleA=10,angleB=170]{A}{B}`

Syntax: `\nccurve[Winkel von Knoten1,Winkel von Knoten2]{Knoten1}{Knoten2}`

→ restliche Bezeichnung

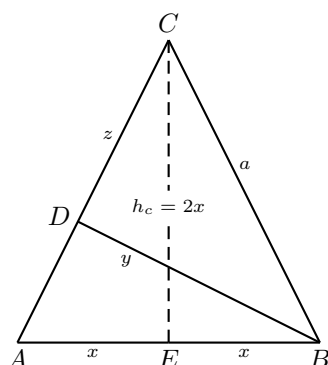
Bsp Z.13: `\rput(3,2){Mitte}`

Syntax: `\rput(x,y){Was}`

1.3 Dreieck mit pst-node

Es wird **pst-node** benötigt. Der Nullpunkt der Grafik ist in Punkt E .

Diese Grafik kann auch einfacher mit dem Paket **pst-eucl** erstellt werden, s. Bsp. 2.1.



Komplette Eingabe für Skizze

```

\psset{unit=20mm,shortput=tablr}
2 \begin{pspicture}(-1.2,-0.2)(1.2,2.2)\SpecialCoor
\psset{labelsep=1mm}
4 \pnode(1;180){A} \nput{-90}{A}{A$}
\pnode(1;0){B} \nput{-90}{B}{B$}
6 \pnode(2;90){C} \nput{90}{C}{C$}
\pnode(1;126.87){D} \nput{160}{D}{D$}
8 \pnode(0,0){E} \nput{-90}{E}{E$}
\psset{labelsep=0.5mm}
10 \ncline{A}{E}_{\scriptsize$x$}
\nclice{E}{B}_{\scriptsize$x$}
12 \ncline{B}{C}>[tpos=0.4]{\scriptsize$a$}
\nclice{C}{A}<[tpos=0.3]{\scriptsize$z$}
14 \ncline{B}{D}_[tpos=0.2]{\scriptsize$y$}
\nclice[linestyle=dashed]{C}{E}\ncput*[npos=.55]{\scriptsize$h_c=2x$}
16 \end{pspicture}

```

Erläuterungen

→ mit **psset** allgemeine Definition für **pspicture** festlegen, beziehungsweise innerhalb von **pspicture** für nachfolgende Segmente

Bsp Z.1: `\psset{unit=20mm,shortput=tablr,labelsep=1mm}`

Syntax: `\psset{allgemeine Anweisungen}`

- ① `unit=20mm` – Einheit für x,y auf 2cm gestellt
- ② `shortput=tablr` – ermöglicht Kurzform der Beschriftung von Verbindungen
- ③ `labelsep=1mm` – Labelabstand

→ spezielle Koordinatenformen aktivieren, um Knoten als Koordinaten zu verwenden, **Z.2:**

Syntax: `\SpecialCoor`

→ Knoten (nicht sichtbar) mit Beschriftung festlegen:

Bsp Z.4: `\pnode(1;180){A} \nput{-90}{A}{A$}`

Syntax: `\pnode(r;α){Knotenname} \nput{α}{Knotenname}{A$}`

- ① Knoten in diesem Fall mit Polarkoordinaten eingegeben (mit Semikolon)

→ Verbindungen mit Beschriftung festlegen:

Bsp Z.13: `\nclice{C}{A}<[tpos=0.3]{\scriptsizez}`

Syntax: `\nclice{Knoten1}{Knoten2}_[Optionen]{Bezeichnung}`

- ① bei t-labeln bezieht sich die Position auf die Zentren der Knoten, keine Sternform
- ② mit Option `tablr` gibt es `_`, `^`, `<` und `>` für unten ($\hat{=}$ `\tbput[Option]{Bez.}`), oben, links und rechts
- ③ Standardbeschriftung ist mittig, mit Option `tpos` kann bel. Position angegeben werden

→ Beschriftung über eine Linie legen

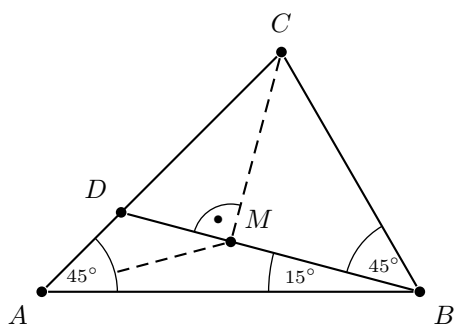
Bsp Z.15: `\nclice[linestyle=dashed]{C}{E}\ncput*[npos=.55]{\scriptsize$h_c=2x$}`

Syntax: `\nclice[Optionen]{Knoten1}{Knoten2}\ncput*[Optionen]{Bezeichnung}`

- ① bei n-Labeln bezieht sich die Position auf das Zentrum der sichtbaren Verbindung, mit Option `npos` kann bel. Position angegeben werden,
- ② `c` (aus `nput`) heißt vertikal mittig auf Verbindung
- ③ Sternversion überschreibt Verbindung

1.4 Dreieck mit Winkeln und Hilfslinien und pst-node

Das Paket **pstricks-add** oder **pst-node** wird benötigt. Die Werte von A , B , C , D und M wurden berechnet. Diese Grafik kann auch einfacher mit dem Paket **pst-eucl** erstellt werden, s. Bsp. 2.3



Komplette Eingabe für Skizze

```

\psset{unit=1mm,radius=2pt}
2 \begin{pspicture}(-5,-5)(55,40) \SpecialCoor
  \Cnode*( 0.0, 0.0){A} \nput{225}{A}{\scriptstyle A}
4  \Cnode*(50.0, 0.0){B} \nput{-45}{B}{\scriptstyle B}
  \Cnode*(31.7,31.7){C} \nput{ 90}{C}{\scriptstyle C}
6  \Cnode*(10.5,10.5){D} \nput{135}{D}{\scriptstyle D}
  \Cnode*(25.0, 6.6){M} \nput{ 45}{M}{\scriptstyle M}
8  \ncline{A}{B}
  \ncline{B}{C}
10 \ncline{C}{A}
  \ncline{B}{D}
12 \ncline[linestyle=dashed]{C}{M}
  \ncline[linestyle=dashed, nodesepA=9.5]{A}{M}
14 \psarc[linewidth=.5pt]{-}(A){10}{0}{45}
  \rput([nodesep=5,angle=22.5]A){\scriptstyle 45^\circ}
16 \psarc[linewidth=.5pt]{-}(B){10}{120}{165}
  \rput([nodesep=5,angle=142.5]B){\scriptstyle 45^\circ}
18 \psarc[linewidth=.5pt]{-}(B){20}{165}{180}
  \rput([nodesep=15,angle=172.5]B){\scriptstyle 15^\circ}
20 \psarc[linewidth=.5pt]{-}(M){5}{75}{165}
  \rput([nodesep=2.5,angle=120]M){\scriptstyle \bullet}
22 \end{pspicture}

```

Erläuterungen

SpecialCoor, nput, ncline siehe Bsp. 1.3

→ Knoten setzen

Bsp Z.3: `\Cnode*(0.0, 0.0){A}`

Syntax: `\Cnode*(x,y){Knotenname}`

① *Cnode* erstellt kreisförmigen Knoten, der Radius wird über die Option *radius* (Z.1) gesetzt

→ Knoten verbinden

Bsp Z.13: `\ncline[linestyle=dashed, nodesepA=9.5]{A}{M}`

Syntax: `\ncline[Optionen]{Knoten1}{Knoten2}`

① *nodesepA* Linienabstand von Knoten

→ Winkel zeichnen

Bsp Z.14: `\psarc[linewidth=.5pt]{-}(A){10}{0}{45}`

Syntax: `\psarc[Optionen]{Strichart}(xM,yM bzw. Knoten){Radius}{Winkel1}{Winkel2}`

① ohne Mittelpunkt wird Koordinatenursprung benutzt

→ Winkel beschriften

Bsp Z.15: `\rput([nodesep=5,angle=22.5]A){\scriptstyle 45^\circ}`

Syntax: `\rput([Optionen]A){Beschriftung}`

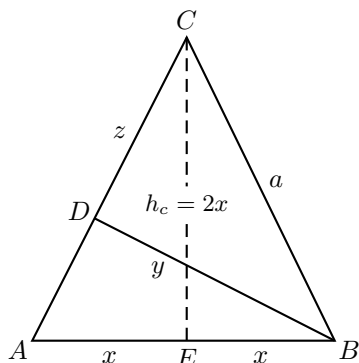
① Winkelbeschriftung bei Knoten A mit folgenden Optionen durch SpecialCoor möglich:

- *nodesep* Abstand horizontal
- *offset* Abstand vertikal
- *angle* Winkel

2 Euklidische Geometrie

2.1 Gleichschenkliges Dreieck mit \overline{AC} , \overline{AB} und Winkel BAC

Es wird `pst-eucl` benötigt. Der Nullpunkt der Grafik ist in Punkt A . Man benötigt \overline{AC} , \overline{AB} und Winkel BAC .



Komplette Eingabe für Skizze

```

\psset{unit=20mm}
2 \begin{pspicture}(-.2,-0.2)(2.2,2.2)
\psset{PointNameSep=1.5ex,
4 CodeFigColor=black,RightAngleType=none,PointSymbol=none}
\pstTriangle(0;0){A}(2;0){B}(2.25;63){C}
6 \pstProjection[CodeFig=true,CodeFigStyle=solid]{C}{A}{B}[D]
\pstProjection[CodeFig=true]{A}{B}{C}[E]
8 \ncput*[npos=.55]{\small$h_c=2x$}
\pstMiddleAB{E}{A}{x}
10 \pstMiddleAB{B}{E}{x}
\pstMiddleAB{C}{B}{a}
12 \pstMiddleAB[PointNameSep=5ex,PosAngle=80]{A}{C}{z}
\pstMiddleAB[PointNameSep=5ex,PosAngle=170]{B}{D}{y}
14 \end{pspicture}

```

Erläuterungen

→ Allgemeine Einstellungen

Bsp Z.3-4: `PointNameSep=1.5ex,CodeFigColor=black,RightAngleType=none,PointSymbol=none`

- ① `PointNameSep=1.5ex` Labelabstand setzen, `CodeFigColor=black` Standard cyan, `RightAngleType=none` kein Symbol für rechten Winkel, `PointSymbol=none` kein Punktsymbol

→ Dreieck zeichnen

Bsp Z.5: `\pstTriangle(0;0){A}(2;0){B}(2.25;63){C}`

Syntax: `\pstTriangle[Optionen](r;α){A}(r;α){B}(r;α){C}`

- ① Dreieck über Polarkoordinaten angegeben (`SpecialCoor` ist automatisch aktiv), Nullpunkt liegt in A

→ Lot auf \overline{AC}

Bsp Z.6: `\pstProjection[CodeFig=true,CodeFigStyle=solid]{C}{A}{B}[D]`

Syntax: `\pstProjection[Optionen]{C}{A}{B}[Knotenname]`

- ① `CodeFig=true` Linie wird sichtbar, Standardvorgabe ist `false`
- ② `CodeFigStyle=solid` durchgehende Linie, Standardvorgabe ist gestrichelte Linie

→ Beschriftung von Lot auf \overline{AB}

Bsp Z.8: `\ncput*[npos=.55]{\small$h_c=2x$}`

Syntax: `\ncput*[Optionen]{Bezeichnung}`

- ① c (aus `ncput`) heißt vertikal mittig auf Verbindung, Sternversion überschreibt Verbindung
- ② die Befehle aus dem `pst-node` Paket können verwendet werden, da es sich auch hier um Knoten handelt

→ Mittelpunkt der Strecke ermitteln, für Seitenbeschriftung

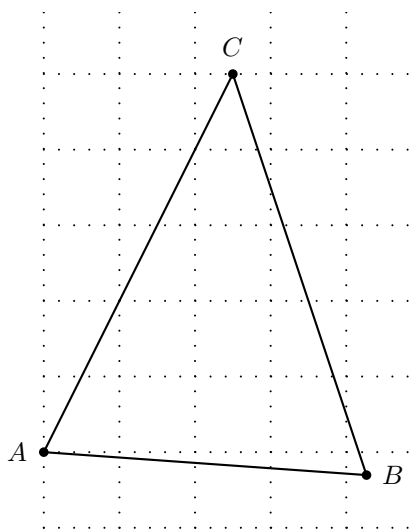
Bsp Z.12: `\pstMiddleAB[PointNameSep=5ex,PosAngle=80]{A}{C}{z}`

Syntax: `\pstMiddleAB[Optionen]{A}{B}{Knotenname}`

- ① Linie nicht sichtbar, da `CodeFig` standardmäßig auf `false` steht
- ② Knotenbeschriftung mittig, `PointNameSep=5ex` ändert Labelabstand und `PosAngle=80` den Winkel

2.2 Gleichschenkliges Dreieck mit A , C und Winkel ACB

Es wird `pst-eucl` benötigt. Der Nullpunkt der Grafik ist in Punkt A . Man benötigt C und Winkel ACB .



Komplette Eingabe für Skizze

```

\begin{pspicture}(-.5,-1)(4.8,5.8)
2 \psgrid[subgriddiv=0,griddots=5,gridlabels=0pt]
\pstGeonode[PosAngle={180,90}](0,0){A}(2.5,5){C}
4 \pstRotation[RotAngle=45]{C}{A}[B]
\pspolygon(A)(C)(B)
6 \end{pspicture}

```

Erläuterungen

→ Gitter erstellt

Bsp Z.2: `\psgrid[subgriddiv=0,griddots=5,gridlabels=0pt]`

Syntax: `\psgrid[Optionen](x,y)`

① wenn das Gitter nicht der Größe des `pspicture`s entsprechen soll, wird es über x - und y -Werte gesetzt

→ Punkte A und C mit Beschriftung erstellt

Bsp Z.3: `\pstGeonode[PosAngle={180,90}](0,0){A}(2.5,5){C}`

Syntax: `\pstGeonode[Optionen](x,y){A}`

① in diesem Fall ist Knotenname = Bezeichnung, sonst über `PointName=` wählbar

② `PosAngle` gibt den Winkel des Labels zum Knoten an ($A = 180^\circ$, $C = 90^\circ$),
`PointNameSep` Abstand wählbar

→ B erzeugen, durch Angabe des Winkels in C

Bsp Z.4: `\pstRotation[RotAngle=45]{C}{A}[B]`

Syntax: `\pstRotation[Optionen]{C}{A}[Knotenname]`

① in $[B]$ steht für den berechneten Knoten und gleichzeitig als Bezeichnung,
wenn eine andere Bezeichnung gewünscht wird `PointName` anwenden

② `RotAngle` Winkel setzen

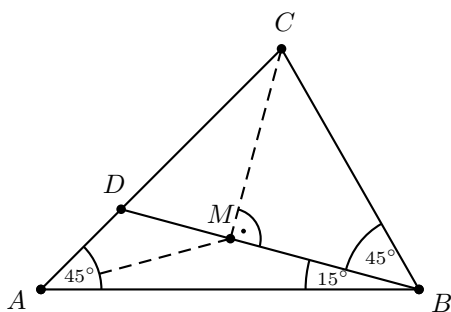
→ Dreieck zeichnen (Knoten miteinander verbinden)

Bsp Z.5: `\pspolygon(A)(C)(B)`

Syntax: `\pspolygon[Optionen](A)(C)(B)`

2.3 Dreieck mit Winkeln und Hilfslinien und pst-eucl

Das Paket `pst-eucl` wird benötigt. Man benötigt, wenn A im Nullpunkt liegt, \overline{AC} , \overline{AB} und $\overline{AD} +$ Winkel.



Komplette Eingabe für Skizze

```
\begin{pspicture}(-.5,-.5)(5.5,4.5)
2 \psset{CodeFig=true,CodeFigColor=black,RightAngleType=german}
\pstTriangle(0;0){A}(5;0){B}(4.5;4.5){C}
4 \pstTriangle[linestyle=none,PointNameA=,
PointNameB=](0;0){A}(5;0){B}(1.5;4.5){D}
6 \pstLineAB{B}{D} %\ncline{B}{D}
\pstProjection[PosAngle=110]{D}{B}{C}[M]
8 \ncline[linestyle=dashed,nodesepA=.8]{A}{M}
\pstMarkAngle[MarkAngleRadius=.8,LabelSep=.5]{B}{A}{C}{
10 $\scriptstyle45^\circ$}
\pstMarkAngle[MarkAngleRadius=1.5,LabelSep=1.2]{D}{B}{A}{
12 $\scriptstyle15^\circ$}
\pstMarkAngle[MarkAngleRadius=1,LabelSep=.7]{C}{B}{D}{
14 $\scriptstyle45^\circ$}
\end{pspicture}
```

Erläuterungen

→ Optionen für Lot auf \overline{DB} einstellen

Bsp Z.2: `\psset{CodeFig=true,CodeFigColor=black,RightAngleType=german}`

Syntax: `\psset{Einstellungen}`

- ① `CodeFigColor` Standard hellblau,
`CodeFig=true` erlaubt Zeichnen vom Lot (sichtbare Translation)
- `RightAngleType=german` rechter Winkel in dt. Ansicht

→ unsichtbares Dreieck für Knoten D erstellen (Dreieck ABC äquivalent nur ohne Optionen)

Bsp Z.4-5: `\pstTriangle[linestyle=none,PointNameA=,PointNameB=](0;0){A}(5;0){B}(1.5;4.5){D}`

Syntax: `\pstTriangle[Optionen](r;α){A}(r;α){B}(r;α){D}`

- ① `linestyle=none` schaltet die Linien aus
- ② `PointNameA=,PointNameB=` Punkte A und B erhalten kein Label
- ③ Dreieck über Polarkoordinaten angegeben (`SpecialCoor` ist automatisch aktiv), Nullpunkt liegt in A

→ Lot auf \overline{DB}

Bsp Z.7: `\pstProjection[PosAngle=110]{D}{B}{C}[M]`

Syntax: `\pstProjection[Optionen]{D}{B}{C}[Bezeichnung]`

- ① `PosAngle=110` Winkel, wo Label erscheinen soll

→ Winkel einzeichnen und beschriften

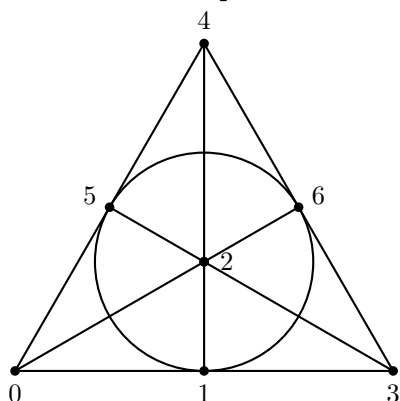
Bsp Z.9-10: `\pstMarkAngle[MarkAngleRadius=.8,LabelSep=.5]{B}{A}{C}{\scriptstyle45^\circ}`

Syntax: `\pstMarkAngle[Optionen]{B}{A}{C}Label`

- ① `MarkAngleRadius=.8` Radius des Winkelbogens
`LabelSep=.5` Labelabstand

2.4 Gleichseitiges Dreieck mit Innkreis

Es wird das Paket **pst-eucl** benötigt.



Komplette Eingabe für Skizze

```

1 \begin{pspicture}(-.2,-.5)(5.2,4.8)
  \psset{PointNameSep=2ex,CodeFigColor=black,RightAngleType=none}
3 \pstGeonode[PosAngle={270,270},PointName={0,3}](0,0){A}(5,0){B}
  \pstRotation[RotAngle=60,PosAngle=90,PointName=4]{A}{B}{C}
5 \pspolygon(A)(C)(B)
  \pstCircleABC[DrawCirABC=false,PointName=2]{A}{B}{C}{M}
7 {\psset{CodeFig=true,CodeFigStyle=solid}
  \pstProjection[PointName=5]{C}{A}{B}{D}
9 \pstProjection[PointName=1]{A}{B}{C}{E}
  \pstProjection[PointName=6]{B}{C}{A}{F}
11 \pstCircleABC[PointName=none]{D}{E}{F}{M}
  \end{pspicture}

```

Erläuterungen

→ Grundeinstellungen

Bsp Z.2: `\psset{PointNameSep=2ex,CodeFigColor=black,RightAngleType=none}`

Syntax: `\psset{Einstellung}`

- ① `PointNameSep=2ex` – Labelabstand setzen, `CodeFigColor=black` – Standard cyan, `RightAngleType=none` – kein Symbol für rechten Winkel

→ Punkte 0 und 3 über Knoten *A* und *B* festlegen mit `\pstGeonode`

Bsp Z.3: `\pstGeonode[PosAngle={270,270},PointName={0,3}](0,0){A}(5,0){B}`

Syntax: `\pstGeonode[Optionen](x,y){Knoten1}(x,y){Knoten2}`

- ① `PosAngle={270,270}` – Label von *A* liegt bei 270 Grad und Label von *B* auch
- ② `PointName=0,3` – Knoten1 soll Labelnamen 0 haben, Knoten2 soll 3 erhalten, als Knotennamen können keine Zahlenwerte vergeben werden, (Standard Knoten1 = Labelname *A*)

→ Punkt 4 (Knoten *C*) anlegen

Bsp Z.4: `\pstRotation[RotAngle=60,PosAngle=90,PointName=4]{A}{B}{C}`

Syntax: `\pstRotation[Optionen]Knoten1Knoten2[Knoten2']`

- ① `RotAngle=60` Knoten2 rotiert 60 Grad um Knoten1 (Bezugspunkt)

→ Knoten *A*, *B* und *C* verbinden

Bsp Z.5: `\pspolygon(A)(C)(B)`

Syntax: `\pspolygon[Optionen](Knoten1)(Knoten2')(Knoten2)`

→ Mittelpunkt festlegen lassen

Bsp Z.6: `\pstCircleABC[DrawCirABC=false,PointName=2]{A}{B}{C}{M}`

Syntax: `\pstCircleABC[Optionen]{Knoten1}{Knoten2}{Knoten2'}{M}`

- ① `\pstCircleABC` – zeichnet Umkreis, sowie den Mittelpunkt
- ② `DrawCirABC=false` – Darstellung des Umkreises ausgeschaltet, da nur Mittelpunkt benötigt

→ Punkte 1, 5 und 6 erstellen (Parallelprojektion eines Punktes auf eine Strecke)

Bsp Z.8: `\pstProjection[PointName=5]{A}{B}{C}{E}`

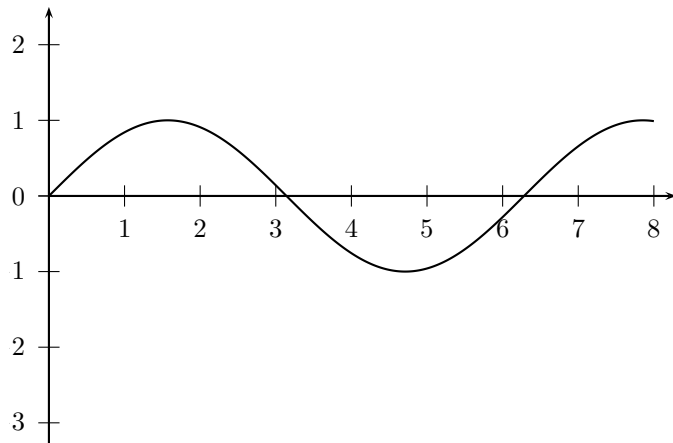
Syntax: `\pstProjection[Optionen]Knoten1Knoten2Knoten2'[Knoten2']`

- ① `CodeFig=true` Linie wird sichtbar, Standardvorgabe ist `false`
- ② `CodeFigStyle=solid` durchgehende Linie, Standardvorgabe ist gestrichelte Linie

3 Funktionen

3.1 Funktion über algebraische Eingabe

Es ist `pstricks-add` erforderlich.



Komplette Eingabe für Skizze

```
\begin{pspicture}(-.5,-3.3)(8.7,3)
2 \psaxes{->}(0,0)(0,-3.3)(8.3,2.5)
\psset{algebraic,plotpoints=501}
4 \psplot{0.01}{8}{sin(x)}
\end{pspicture}
```

Erläuterungen

→ Achsen erstellen

Bsp Z.2: `\psaxes{->}(0,0)(0,-3.3)(8.3,2.5)`

Syntax: `\psaxes[Optionen]{Linienart}(Ursprung)(x_1,y_1)(x_2,y_2)`

→ auf algebraische Funktionseingabe schalten

Bsp Z.3: `\psset{algebraic}`

① die möglichen Funktionen findet man in `pstricks-add-doc.pdf`

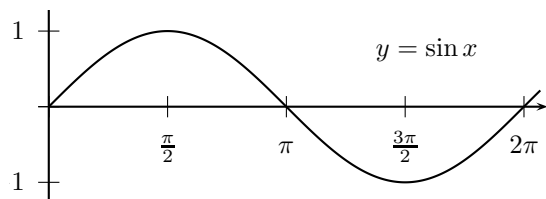
→ Funktion zeichnen

Bsp Z.4: `\psplot{0.01}{8}{sin(x)}`

Syntax: `\psplot[Optionen]{ x_{min} }{ x_{max} }{Funktion}`

3.2 Funktion mit pstricks-add

Es ist das Paket **pstricks-add** für die Achsenbeschriftung und die Kurve erforderlich.



Komplette Eingabe für Skizze

```

1 \begin{pspicture}(-.5,-1.25)(8,1.25)
  \psaxes[showorigin=false,trigLabels,xunit=1.571,
3     trigLabelBase=2]{->}(0,0)(0,-1.25)(4.2,1.5)
  \psplot[plotstyle=curve,plotpoints=500]{0}{6.5}{x RadtoDeg sin}
5 \rput(5,0.75){$y=\sin x$}
  \end{pspicture}

```

Erläuterungen

→ Achsen zeichnen

Bsp Z.2: `\psaxes[showorigin=false,trigLabels,xunit=1.571,
trigLabelBase=2]{->}(0,0)(0,-1.25)(4.2,1.5)`

Syntax: `\psaxes[Optionen]{Linienart}(Ursprung)(x1,y1)(x2,y2)`

- ① `showorigin=false` der Nullpunkt entfällt in der Achsenbeschriftung
- ② `trigLabels` Einheiten werden automatisch im Bogenmaß gesetzt, Achsenbeschriftung in π dabei muss `xunit` ein Teil oder Vielfaches von π sein

→ Kurve zeichnen

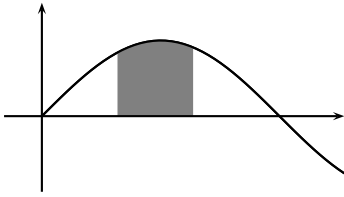
Bsp Z.4: `\psplot[plotstyle=curve,plotpoints=500]{0}{6.5}{x RadtoDeg sin}`

Syntax: `\psplot[Optionen]{xmin}{xmax}{Funktion}`

- ① `RadtoDeg` x von Rad nach Grad wechseln

3.3 Funktion mit farbiger Fläche

Es wird pst-plot oder pstricks-add benötigt.



Komplette Eingabe für Skizze

```

1 \begin{pspicture}(-.5,-1)(4,1.5)
2 \begin{psclip}{\psplot{0}{4}{x RadtoDeg sin}}
3 \psframe*[linecolor=gray](1,0)(2,1.5)
4 \end{psclip}
5 \psaxes[labels=none,ticks=none]{->}(0,0)(-0.5,-1)(4,1.5)
6 \psplot{0}{4}{x RadtoDeg sin}
7 \end{pspicture}

```

Erläuterungen

→ farbige Fläche mit psclip definieren

Bsp Z.2-4: `\begin{psclip}{x RadtoDeg sin}`
`\psframe*[linecolor=gray](1,0)(2,1.5)`
`\end{psclip}`

Syntax: `\begin{psclip}{Randkurve} Material \end{psclip}`

- ① durch die Clipbox kann beliebiges Material entlang jeder Kurve beschnitten werden, hier ist als Randkurve die Sinusfunktion angegeben, die im letzten Punkt gezeichnet wird
als Fläche ist ein Rechteck mit grauer Färbung definiert welches durch die Sinusfunktion beschnitten wird
- ② falls die Randkurve keine Begrenzung durch die Achsen hat muss mit Hilfe von pscustom (Objekte definieren) eine derartige Fläche erzeugt werden, siehe Beispiel unten

→ Achsen zeichnen

Bsp Z.5: `\psaxes[labels=none,ticks=none]{->}(0,0)(-0.5,-1)(4,1.5)`

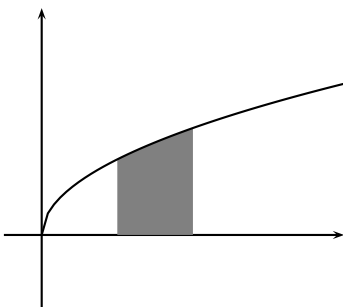
Syntax: `\psaxes[Optionen]{Linienart}(Ursprung)(x1,y1)(x2,y2)`

- ① `labels=none` Label werden ausgeschaltet
- ② `ticks=none` Achseneinteilung wird ausgeschaltet

→ Kurve zeichnen

Bsp Z.6: `\psplot{0}{4}{x RadtoDeg sin}`

Syntax: `\psplot[Optionen]{xmin}{xmax}{Funktion}`



Komplette Eingabe für Skizze

```

1 \begin{pspicture}(-1,-1)(4,3)
2 \psaxes[labels=none,ticks=none]{->}(0,0)(-0.5,-1)(4,3)
3 \begin{psclip}{\pscustom[linestyle=none]{\psplot{0}{4}{x sqrt}\lineto(4,0)}}
4 \psframe*[linecolor=gray](1,0)(2,4)
5 \end{psclip}
6 \psplot{0}{4}{x sqrt}
7 \end{pspicture}

```

→ pscustom definieren

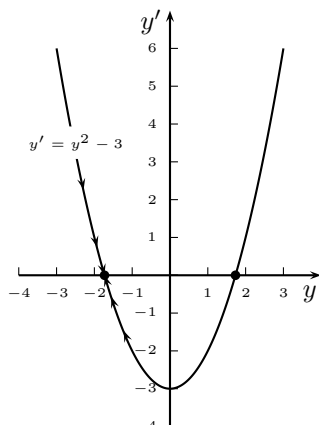
Bsp Z.3: `\pscustom[linestyle=none]{\psplot{0}{4}{x sqrt}\lineto(4,0)}`

Syntax: `\pscustom[Optionen]{Objekt}`

- ① durch `\lineto(4,0)` wird für Objekt eine Begrenzung definiert, die durch Option `[linestyle=none]` nicht sichtbar ist

3.4 Kurve mit Pfeilen, newpsobj, PS-Code

Es werden die Pakete *pstricks-add* und *pst-plot* benötigt.



Komplette Eingabe für Skizze

```

1 \psset{unit=.5cm}
2 \def\pshlabel#1{\tiny $#1$}\def\psvlabel#1{\tiny $#1$}
3 \newpsobject{sdplot}{psplot}{plotpoints=3,linestyle=none,
   arrows=->,ArrowInside=->}
4
5 \begin{pspicture}(-4.5,-4)(4,7)
6 \SpecialCoor
7 \psaxes[ticksize=3pt]{->}(0,0)(-4,-4)(4,7)
8 \rput(3.7,-.5){$y$}
9 \rput(-.5,6.7){$y'$}
10 \psset{algebraic,plotpoints=500}
11 \sdplot{-2.5}{-1.732050808}{x^2-3}
12 \sdplot{-1}{-1.732050808}{x^2-3}
13 \psplot[plotstyle=curve]{-3}{3}{x^2-3}
14 \rput*(-2.5,3.5){\tiny$y'=y^2-3$}
15 \psdot(! 3 sqrt 0)
16 \psdot(! 3 sqrt neg 0)
17 \end{pspicture}

```

Erläuterungen

→ Achsen festlegen

Bsp Z.7: `\psaxes[ticksize=3pt]{->}(0,0)(-4,-4)(4,7)`

Syntax: `\psaxes[Optionen]{Linienart}(Ursprung)(x_1,y_1)(x_2,y_2)`

- ① `ticksize=3pt` Größe der Achseneinteilungen
- ② Schriftgröße wurde in Zeile 2 geändert

→ Kurve zeichnen

Bsp Z.13: `\psplot[plotstyle=curve]{-3}{3}{x^2-3}`

Syntax: `\psplot[Optionen]{ x_{min} }{ x_{max} }{Funktion}`

- ① in Zeile 10 wurden die Optionen für die Kurve gesetzt

→ Punkte auf x-Achse zeichnen

Bsp Z.15: `\psdot(! 3 sqrt 0)`

Syntax: `\psdot[Optionen](x,y)`

- ① durch die Verwendung von *SpecialCoor* in Z. 6 kann auch PostscriptCode benutzt werden (eingeleitet durch !)

→ Optionen definieren, die für bestimmtes Objekt mehrmals benötigt werden

Bsp Z.3: `\newpsobject{sdplot}{psplot}{plotpoints=3,linestyle=none,arrows=->,ArrowInside=->}`

Syntax: `\newpsobject{myObjekt}{Objekt}{Optionen}`

- ① hier wird die Anzahl der Knoten definiert, die Kurve unsichtbar, und Pfeile für die Ansicht gewählt

→ Pfeile auf Kurve setzen

Bsp Z.11: `\sdplot{-2.5}{-1.732050808}{x^2-3}`

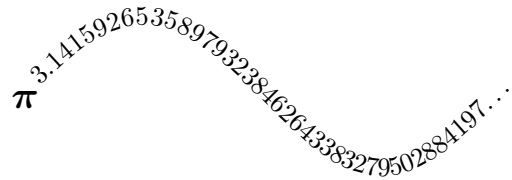
Syntax: in diesem Bsp.: `\myObjekt[Optionen]{ x_{min} }{ x_{max} }{Funktion}`

- ① funktioniert auch bei anderen Linien bzw. Kurvenzügen
- ② Art und Anzahl der Pfeile wird über Optionen festgelegt
- ③ die Richtung der Pfeile erfolgt über `{ x_{min} }{ x_{max} }` bzw. `{ x_{max} }{ x_{min} }`

4 Diverses

4.1 Text auf Kurve

Es sind die Pakete `pstricks-add` und `pst-text` erforderlich.



Komplette Eingabe für Skizze

```

1 \begin{pspicture}(-.2,-1)(7,1.3)
  \psset{linestyle=none,algebraic,plotpoints=50}
3 \rput(0,0){\Large\boldmath$\pi$}
  \pstextpath[l](0,0){%
5 \psplot{0.2}{7}{sin(x)}
  }{3.141592653589793238462643383279502884197\dots}
7 \end{pspicture}

```

Erläuterungen

→ Grafikeinstellungen

Bsp Z.2: `\psset{linestyle=none,algebraic,plotpoints=50}`

- ① `linestyle=none` – Kurve an der der Text entlang läuft, nicht ausgeben
- ② `algebraic` – auf algebraische Funktionseingabe schalten (Paket `pstricks-add`)
- ③ `plotpoints=50` – Kurvengenauigkeit (Punkte für angegebenes Intervall)

→ Pi als Variable auf 0,0 setzen

Bsp Z.3: `\rput(0,0){\Large\boldmathπ}`

Syntax: `\rput(x,y){Was}`

→ Textkurve mit Text erstellen (Paket `pst-text`)

Bsp Z.4-6: `\pstextpath[l](0,0){\psplot{0.2}{7}{sin(x)}}{3.141\dots}`

Syntax: `\pstextpath[Optionen](x,y){Funktion}{Text}`

- ① als Optionen sind `l,c,r` zulässig
 - `l` – Text startet am Funktionsanfang
 - `c` – Text startet mittig
 - `r` – Text startet am Funktionsende
- ② `(x,y)` – Offset (Funktion – Text)