LaTeX and Friends
Creating Diagrams with tikz

http://csweb.ucc.ie/~dongen/LAF/LAF.html

M. R. C. van Dongen

UCC

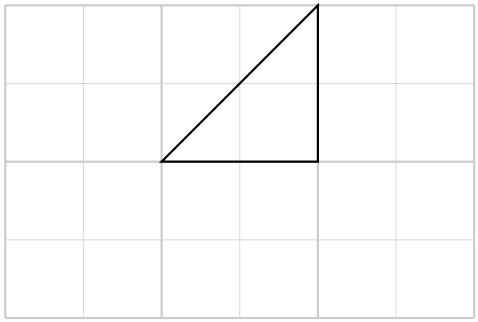
Introducing tikz

- Drawing with tikz is done in tikzpicture environment.
- The tikzpicture is drawn as smallest possible box.
- All *implicit* units inside a tikzpicture are in centimetres.
- The following draws a 0.4 × 0.2 crossed rectangle: ⊠.

LaTeX Usage

```
The following draws
 a $0.4 \times 0.2$ crossed rectangle:
\begin{tikzpicture}
\draw (0.0,0.0) rectangle (0.4,0.2);
\draw (0.0,0.0) -- (0.4,0.2);
\draw (0.0,0.3) -- (0.4,0.0);
\end{tikzpicture}\,.
```

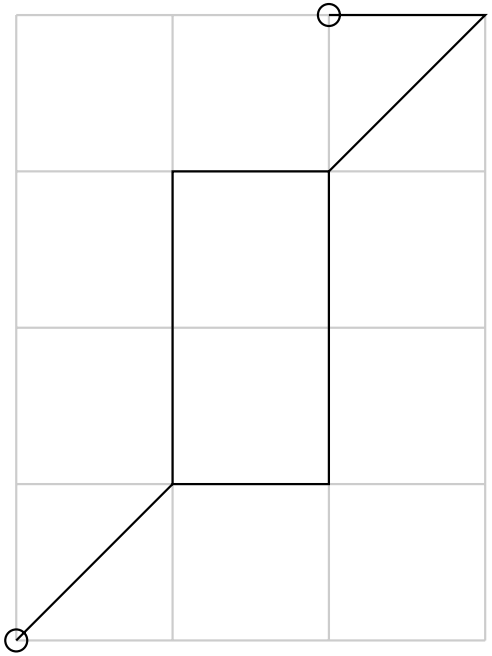
Grids



LaTeX Input

```
\draw[line width=0.1pt,gray!30,step=5mm]
     (0,0) grid (3,2);
\draw[help lines]
     (0,0) grid (3,2);
\draw (1,1) --
      (2,2) -- (2,1) -- cycle;
```

Paths



LaTeX Input

```
\draw[help lines] (0,0) grid (3,4);
\draw (0,0) circle (2pt)
        -- (1,1) rectangle (2,3)
        -- (3,4)
        -- (2,4) circle (2pt);
```

Coordinate Labels

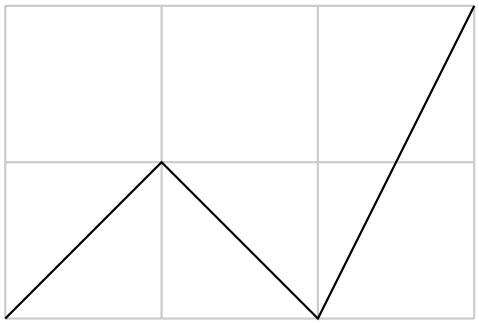
LaTeX Usage

```
The following, which draws a crossed rectangle
(\begin{tikzpicture}
 \draw (0.0,0.0) coordinate(lower left)
         -- (0.4,0.2) coordinate(upper right);
 \draw (0.0,0.2) -- (0.4,0.0);
 \draw (lower left) rectangle (upper right);
 \end{tikzpicture}), demonstrates the mechanism.
```

LaTeX Output

The following, which draws a crossed rectangle (), demonstrates the mechanism.

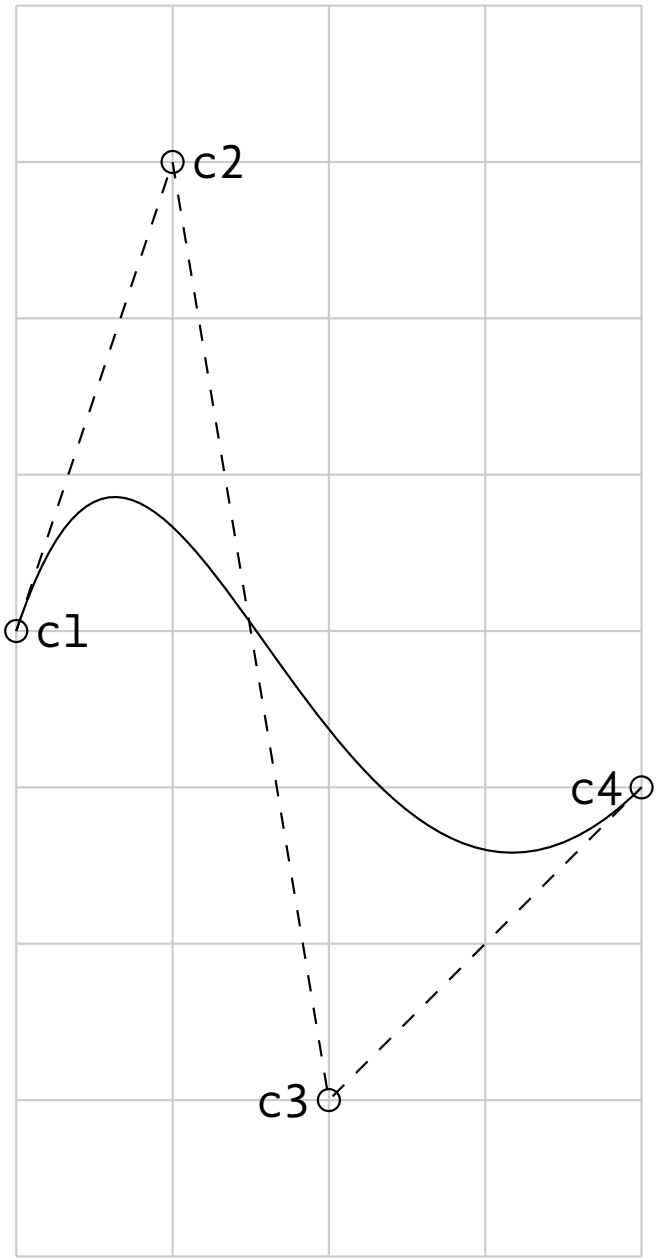
Line-To Operation



LaTeX Input

```
\draw[help lines] (0,0) grid (3,2);
\draw (0,0) -- (1,1) --
      (2,0) -- (3,2);
```

Curve-to Operation: Output



Curve-to Operation: Input

LaTeX Input

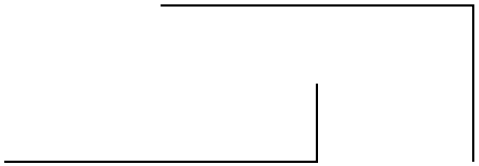
```
\draw[help lines] (-2,-4) grid (+2,+4);
  \path (-2,+0) coordinate(c1)
        (-1,+3) coordinate(c2)
        (+0,-3) coordinate(c3)
        (+2,-1) coordinate(c4);
  \draw[dashed] (c1) -- (c2) -- (c3) -- (c4);
  \draw (c1) circle (2pt)
        (c2) circle (2pt)
        (c3) circle (2pt)
        (c4) circle (2pt)
        (c1) .. controls (c2)
                     and (c3) .. (c4)
        (c1) node[anchor=west] {\texttt{c1}}
        (c2) node[anchor=west] {\texttt{c2}}
        (c3) node[anchor=east] {\texttt{c3}}
        (c4) node[anchor=east] {\texttt{c4}};
```

Cycle Operation

LaTeX Input

```
\draw (0,0) -- (1,1)
      (2,0) -- (3,0) --
      (3,1) -- cycle;
```

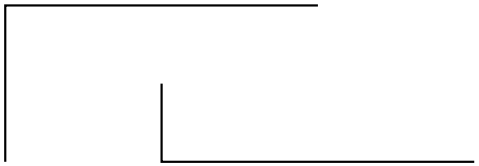
Horizontal and Vertical Connections



LaTeX Input

```
\tikz \draw (0.0,0.0) -| (2.0,0.5)
            (1.0,1.0) -| (3.0,0.0);
```

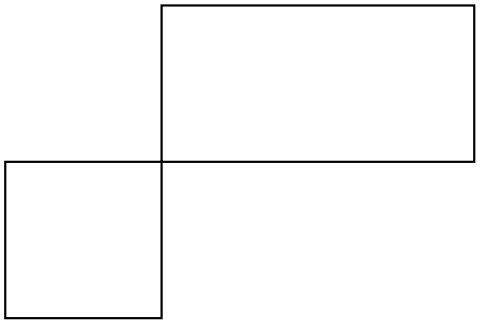
Horizontal and Vertical Connections (Continued)



LaTeX Input

```
\tikz \draw (0.0,0.0) |- (2.0,1.0)
            (1.0,0.5) |- (3.0,0.0);
```

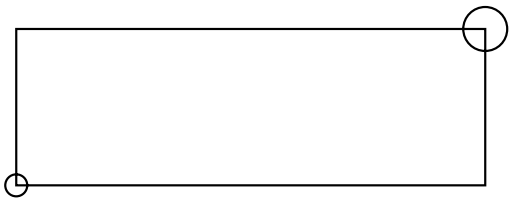
Rectangle Operation



LaTeX Input

```
\begin{tikzpicture}
\draw (0,0) rectangle (1,1)
            rectangle (3,2);
\end{tikzpicture}
```

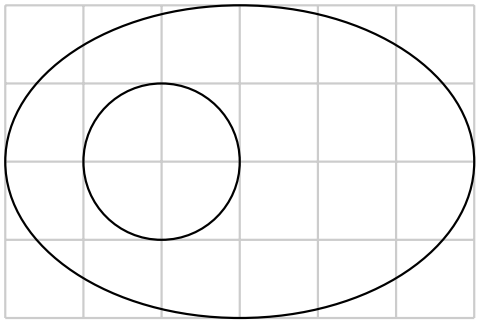
Circle Operation



LaTeX Input

```
\tikz \draw (0,0) circle (2pt)
                  rectangle (3,1)
                  circle (4pt);
```

Ellipse Operation

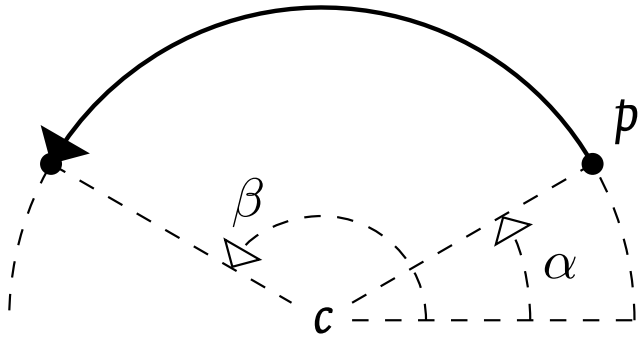


LaTeX Input

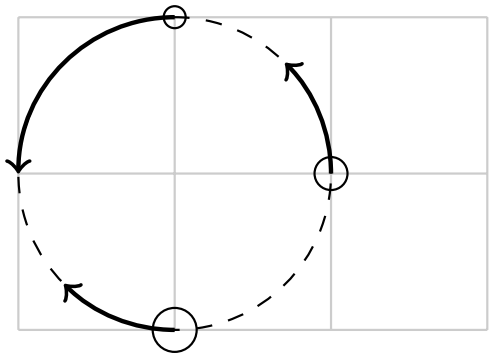
```
\begin{tikzpicture}[scale=0.5]
\draw[help lines] (0,0) grid (6,4);
\draw (2,2) ellipse (1cm and 1cm)
      (3,2) ellipse (3cm and 2cm);
\end{tikzpicture}
```

Arc Operation

`\path ... arc (α:β:r) ...;`



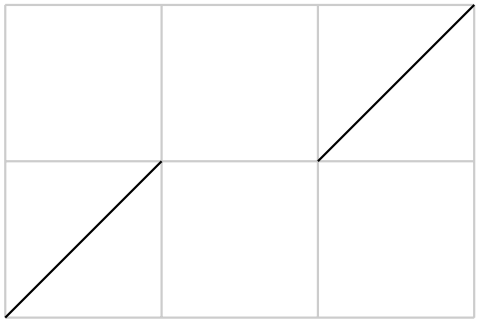
Arc Operation (Continued)



LaTeX Input

```
\begin{tikzpicture}
\draw[help lines] (0,0) grid (3,2);
\draw[dashed] (1,1) circle (1cm);
\draw (1,2) coordinate(a) circle (2pt)
      (2,1) coordinate(b) circle (3pt)
      (1,0) coordinate(c) circle (4pt);
\draw[->,thick] (a) arc (90:180:1cm);
\draw[->,thick] (b) arc (0:45:1cm);
\draw[->,thick] (c) arc (270:225:1cm);
\end{tikzpicture}
```

Move-To Operation



LaTeX Input

```
\begin{tikzpicture}
\draw[help lines] (0,0) grid (3,2);
\draw (0,0) -- (1,1)
      (2,1) -- (3,2);
\end{tikzpicture}
```

Filling a Path

LaTeX Output

LaTeX Input

```
\fill[gray] (0,0) rectangle (3,0.5);
```

Filling and Drawing a Path

LaTeX Output

LaTeX Input

```
\filldraw[fill=gray,draw=black]
        (0,0) rectangle (3,0.5);
```

Shading a Path

LaTeX Output

LaTeX Input

```
\shade[left color=black,right color=gray]
      (0,0) rectangle (3,0.5);
```

Shading and Drawing a Path

LaTeX Output

LaTeX Input

```
\shadedraw[left color=black,
           right color=white,
           draw=gray]
          (0,0) rectangle (3,0.5);
```

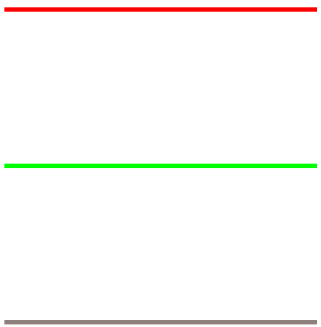
Some Existing Named Colours

black	darkgray	lime	pink	violet
blue	gray	magenta	purple	white
brown	green	olive	red	yellow
cyan	lightgray	orange	teal	

Defining New Colours

```
\definecolor{⟨name⟩}{rgb}{⟨red⟩,⟨green⟩,⟨blue⟩}
\definecolor{⟨name⟩}{gray}{⟨ratio⟩}
\colorlet{⟨name⟩}{⟨colour⟩!⟨percentage⟩}
\colorlet{⟨name⟩}{⟨colour_1⟩!⟨percentage⟩!⟨colour_2⟩}
```

Colour



LaTeX Input

```
\begin{tikzpicture}[color=red]
\draw                       (0,3) -- (2,3);
\draw[color=green]          (0,2) -- (2,2);
\draw[color=cyan!50!red] (0,1) -- (2,1);
\end{tikzpicture}
```

Variations

LaTeX Input

```
\begin{tikzpicture}[gray]
\draw[orange!80!teal] (0,0) -- (2,0);
\end{tikzpicture}
```

Drawing the Path

LaTeX Input

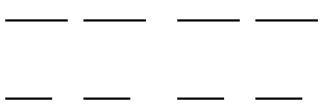
```
\draw[draw=gray] (0,1) -- (2,1);
```

Line Style

LaTeX Input

```
\draw[line width=8pt]
     (0,0) -- (2,4pt);
```

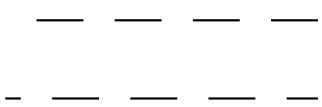
Dash Patterns



LaTeX Input

```
\draw[dash pattern=on 4mm off 1mm on 4mm off 2mm]
      (0,0.5) -- (2,0.5);
\draw[dash pattern=on 3mm off 2mm on 3mm off 3mm]
      (0,0.0) -- (2,0.0);
```

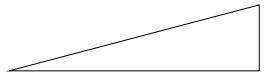
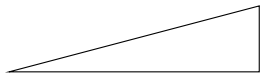
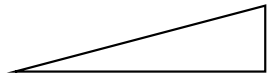
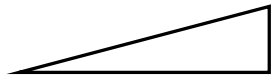
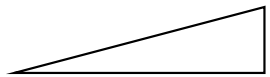
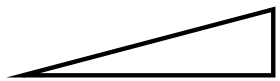
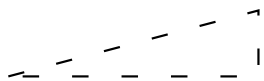
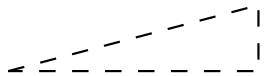
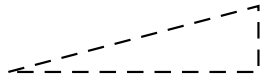
Dash Pahases



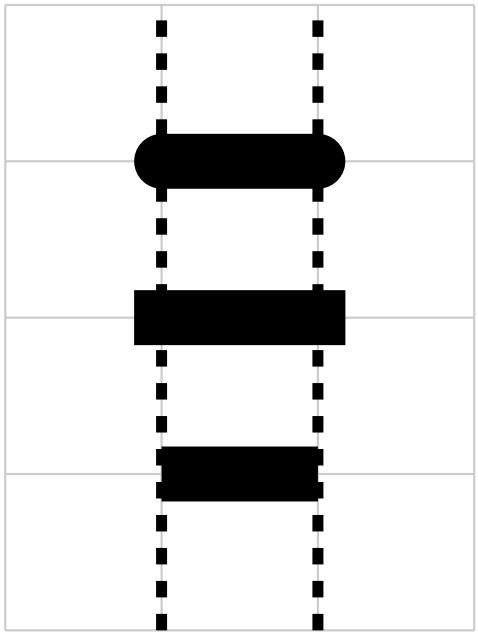
LaTeX Input

```
\begin{tikzpicture}[dash pattern=on 3mm off 2mm]
\draw[dash phase=3mm] (0,0.5) -- (2,0.5);
\draw[dash phase=2mm] (0,0.0) -- (2,0.0);
\end{tikzpicture}
```

Predefined Line Styles

Line Styles			Dash Patterns	
Name	**Width**	**Example**	**Name**	**Example**
ultra thin	0.1 pt		loosely dotted	
very thin	0.2 pt		dotted	
thin	0.4 pt		densely dotted	
semithick	0.6 pt		solid	
thick	0.8 pt		loosely dashed	
very thick	1.2 pt		dashed	
ultra thick	1.6 pt		densely dashed	

Line Cap



LaTeX Input

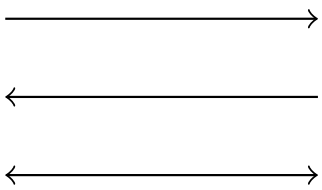
```
\begin{tikzpicture}[line width=10pt]
\draw[help lines] (0,0) grid (3,4);
\draw[line width=2pt,dashed]
     (1,0) -- (1,4)   (2,0) -- (2,4);
\draw[line cap=round] (1,3) -- (2,3);
\draw[line cap=rect]  (1,2) -- (2,2);
\draw[line cap=butt]  (1,1) -- (2,1);
\end{tikzpicture}
```

Line Join

LaTeX Input

```
\begin{tikzpicture}[line width=8pt]
\draw[line join=round]
     (0.0,.8)--(0.3,.0)--(0.6,.8);
\draw[line join=miter]
     (0.9,.0)--(1.2,.8)--(1.5,.0);
\draw[line join=bevel]
     (1.8,.8)--(2.1,.0)--(2.4,.8);
\end{tikzpicture}
```

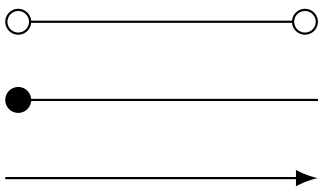
Arrows



LaTeX Input

```
\draw[->]  (0,1.0) -- (2,1.0);
\draw[<-]  (0,0.5) -- (2,0.5);
\draw[<->] (0,0.0) -- (2,0.0);
```

Using Different Arrow Heads

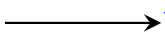
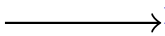
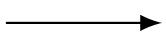


LaTeX Input

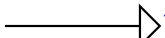
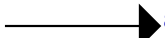
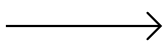
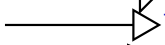
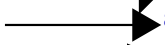
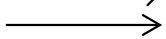
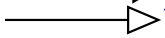
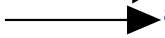
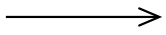
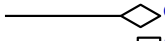
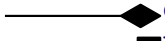
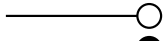
```
\draw[>=o,<->]    (0,1.0) -- (2,1.0);
\draw[>=*,<-]     (0,0.5) -- (2,0.5);
\draw[>=latex,->] (0,0.0) -- (2,0.0);
```

Predefined Arrow Heads

Predefined

Style	Arrow	Style	Arrow	Style	Arrow
stealth		to		latex	
space					

Provided by `arrows`

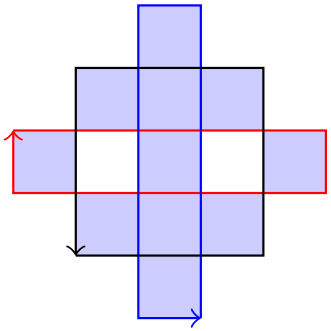
open triangle 90		triangle 90		angle 90	
open triangle 60		triangle 60		angle 60	
open triangle 45		triangle 45		angle 45	
open diamond		diamond		o	
open square		square		*	

Filling a Path

LaTeX Input

```
\begin{tikzpicture}[scale=0.4,fill=gray]
\path[fill]
     (0,0) rectangle (1,1);
\path[fill=black!30]
     (2,0) -- (3,0) -- (3,1) -- cycle;
\path[fill,color=gray]
     (4,0) -- (5,0) -- (5,1);
\end{tikzpicture}
```

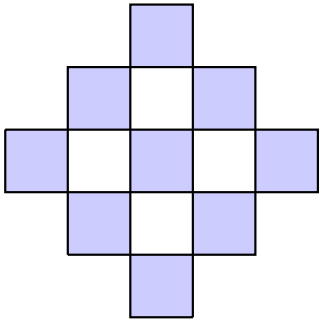
Filling Options: The Nonzero Rule



LaTeX Input

```
\begin{tikzpicture}[fill=blue!20,scale=0.4]
\fill (0,2) -- (0,3) -- (5,3) -- (5,2)
      (2,0) -- (3,0) -- (3,5) -- (2,5)
      (1,1) -- (4,1) -- (4,4) -- (1,4);
\draw[red,->]
     (0,3) -- (5,3) -- (5,2) -- (0,2) -- (0,3);
\draw[blue,->]
     (3,0) -- (3,5) -- (2,5) -- (2,0) -- (3,0);
\draw[->]
     (1,1) -- (4,1) -- (4,4) -- (1,4) -- (1,1);
\end{tikzpicture}
```

Filling Options: The Even Odd Rule



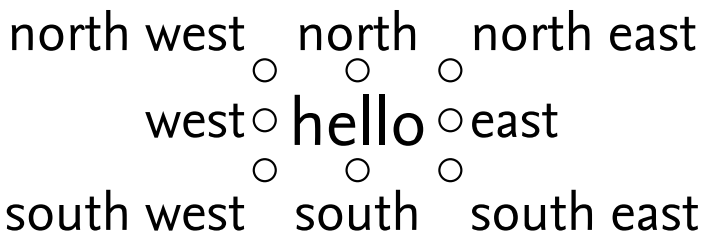
LaTeX Input

```
\begin{tikzpicture}[fill=blue!20,scale=0.4]
\fill[even odd rule]
      (0,2) -- (0,3) -- (5,3) -- (5,2)
      (2,0) -- (3,0) -- (3,5) -- (2,5)
      (1,1) -- (4,1) -- (4,4) -- (1,4);
\draw (0,3) -- (5,3) -- (5,2) -- (0,2) -- (0,3);
\draw (3,0) -- (3,5) -- (2,5) -- (2,0) -- (3,0);
\draw (1,1) -- (4,1) -- (4,4) -- (1,4) -- (1,1);
\end{tikzpicture}
```

Implicit Node Labels

```
\path ... node(⟨label⟩)[⟨options⟩]{⟨content⟩} ... ;
\draw ... node(⟨label⟩)[⟨options⟩]{⟨content⟩} ... ;
```

Example



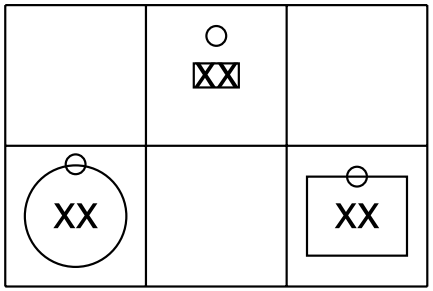
LaTeX Input

```
\begin{tikzpicture}
\draw (0,0) node(hello)[scale=1.25] {hello};
\draw (hello.north)      circle (2pt)
      node[anchor=south]            {north};
\draw (hello.north east) circle (2pt)
      node[anchor=south west]       {north east};
... % remaining commands omitted.
```

Node Shapes

`coordinate`	For coordinates.
`rectangle`	For rectangles (default).
`circle`	For circles.
`ellipse`	For ellipses.

Example



LaTeX Input

```
\draw (0,0) grid (3,2);
\draw (1.5,2.5) node(a)[draw,inner sep=0pt,
                        outer sep=5pt] {xx};
\draw (3.5,1.5) node(b)[draw,inner sep=5pt,
                        outer sep=0pt] {xx};
\draw (1.5,1.5) node(c)[draw,shape=circle]  {xx};
\draw (a.north) circle (2pt);
\draw (b.north) circle (2pt);
\draw (c.north) circle (2pt);
```

Node Options

draw

scale=⟨factor⟩

anchor=⟨anchor⟩

shift=⟨shift⟩

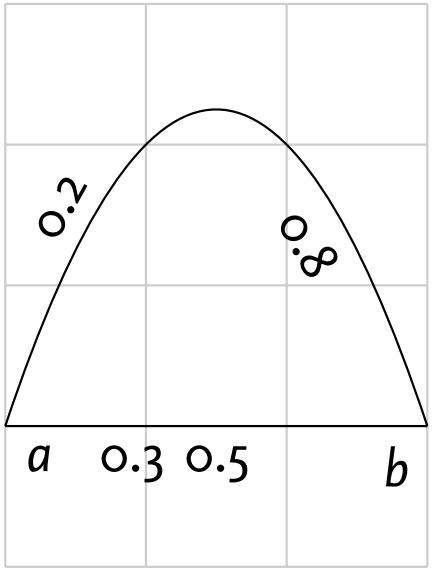
rotate=⟨angle⟩

pos=⟨real⟩

pos=sloped

midway

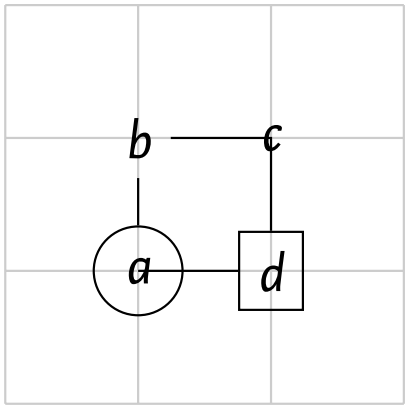
Example



LaTeX Input

```
\draw[help lines] (0,0) grid (3,4);
\draw (0,1) coordinate(a)
          node[anchor=north west] {$a$}
   -- (3,1) coordinate(b)
          node[anchor=north east] {$b$}
          node[pos=0.3,anchor=north] {$0.3$}
          node[pos=0.5,anchor=north] {$0.5$}
      (a) .. controls (1,4) and (2,4) .. (b)
          node[pos=0.2,sloped,anchor=south] $0.2$
          node[pos=0.8,sloped,anchor=north] $0.8$;
```

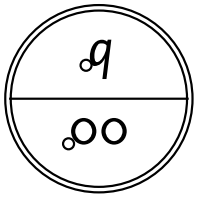
Drawing to and from Nodes



LaTeX Input

```
\draw[help lines] (0,0) grid (3,3);
\path (1,1) node(a)[draw,shape=circle]    {$a$};
\path (1,2) node(b)[shape=rectangle]      {$b$};
\path (2,2) node(c)[shape=circle]         {$c$};
\path (2,1) node(d)[draw,shape=rectangle] {$d$};
\draw (a) -- (b) -- (c.center) -- (d) -- (a.center);
```

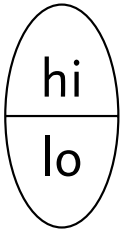
Special Node Shapes: `circle split`



LaTeX Input

```
\draw (0,0)
      node(double)[circle split,draw,double]
                  {$q$ \nodepart{lower} $OO$}
      (double.lower) circle (1pt)
      (double.text)  circle (1pt);
```

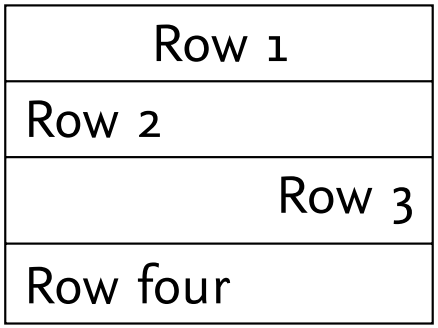
Special Node Shapes: `ellipse split`



LaTeX Input

```
\draw (0,0) node[ellipse split,draw]
            {hi \nodepart{lower} lo};
```

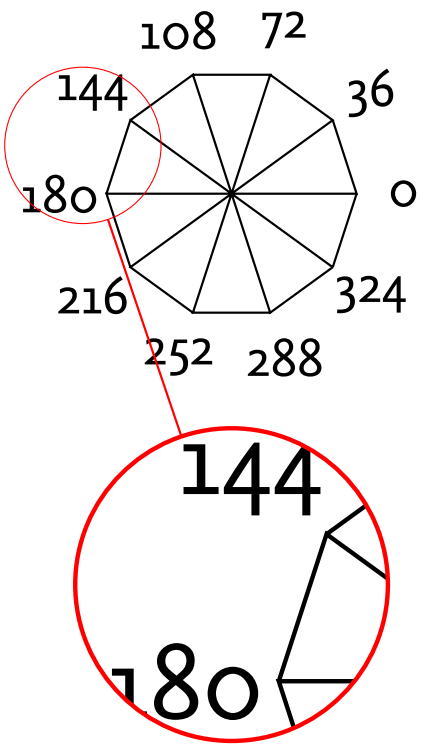
Special Node Shapes: `rectangle split`



LaTeX Input

```
\node[rectangle split, rectangle split parts=4,
      every text  node part/.style={align=center},
      every two   node part/.style={align=left},
      every three node part/.style={align=right},
      draw, text width=2.5cm]
     { Row 1
       \nodepart{two} Row 2
       \nodepart{three} Row 3
       \nodepart{four} Row four };
```

The spy Library (Output)

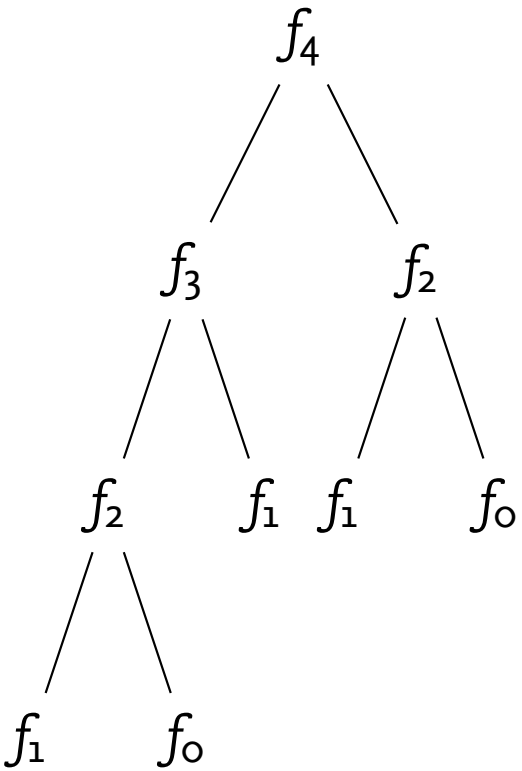


The spy Library (Input)

LaTeX Input

```
\begin{tikzpicture}
      [spy using outlines={circle,
                           magnification=2,
                           size=2cm,
                           connect spies}]
\draw (-36:0.8)
      \foreach \angle in {0,36,...,359} {
          -- (\angle:0.8)
             (\angle:1.1) node {$\angle$}
             (0,0) -- (\angle:0.8)
      };
\spy[red] on (162:1.0) in node[right] at (0,-2.5);
\end{tikzpicture}
```

Drawing Trees (Output)

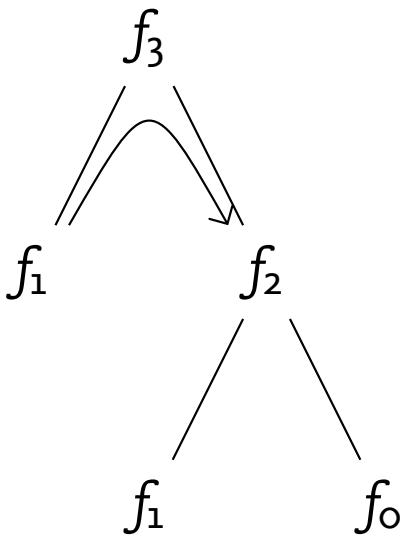


Drawing Trees (Input)

LaTeX Input

```
\begin{tikzpicture}
      [level 2/.style={sibling distance=10mm}]
  \node {$f_4$}
        child {node {$f_3$}
               child {node {$f_2$}
                      child {node {$f_1$}}
                      child {node {$f_0$}}}
               child {node {$f_1$}}}
        child {node {$f_2$}
               child {node {$f_1$}}
               child {node {$f_0$}}};
\end{tikzpicture}
```

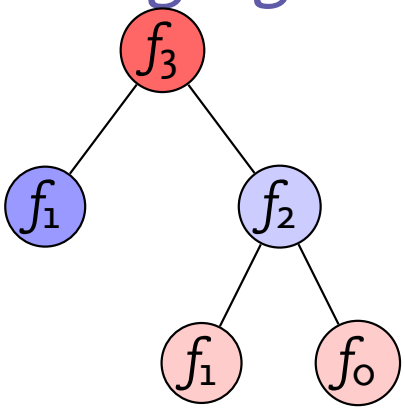
Node Labels in Trees



LaTeX Input

```
\node (top) {$f_3$}
      child {node {$f_1$}}
      child {node {$f_2$}
             child {node {$f_1$}}
             child {node {$f_0$}}};
\draw[-angle 90]
     (top-1.north east) .. controls (top.south)
                        .. (top-2.north west);
```

Changing the Node Style

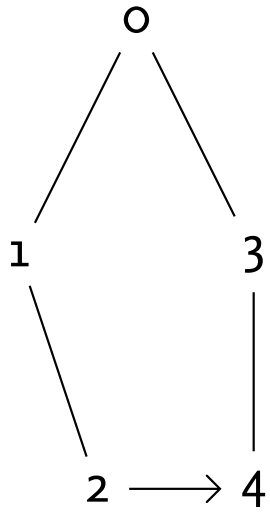


LaTeX Input

```
\begin{tikzpicture}
      [level distance=10mm%
      ,every node/.style={fill=red!60,%
                          circle,%
                          draw=black,%
                          inner sep=1pt}%
      ,level 1/.style={sibling distance=15mm},%
      ,level 2/.style={sibling distance=10mm,%
                       nodes={fill=red!20}}]
  \node (top) {$f_3$}
        child {node[fill=blue!40] {$f_1$}}
        child {node[fill=blue!20] {$f_2$}
               child {node {$f_1$}}
               child {node {$f_0$}}};
\end{tikzpicture}
```

Missing in Action

Automatic Node Placement is not Always Ideal



LaTeX Input

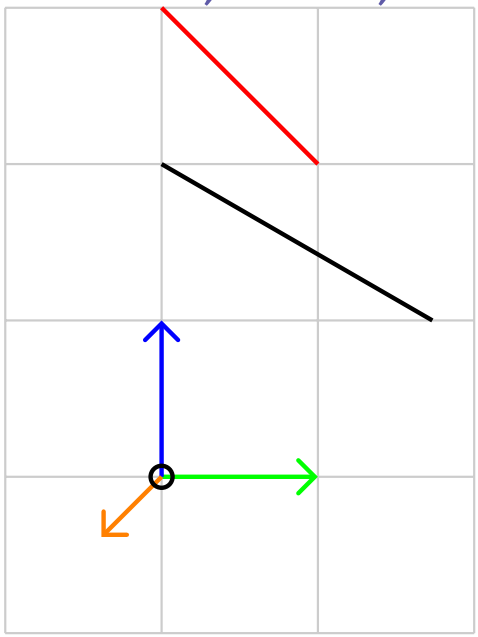
```
\begin{tikzpicture}
      [level 2/.style={sibling distance=10mm}]
  \node (top) {$0$}
        child {node {$1$}
               child[missing]
               child {node {$2$}}}
        child {node {$3$}
               child {node {$4$}}};
  \draw[-angle 90]
       (top-1-2.east) -- (top-2-1.west);
\end{tikzpicture}
```

Coordinate Systems

explicit ⟨system name⟩ cs:⟨coordinate specification⟩.

implicit (0,1), (label), (0,1 |- label), ...

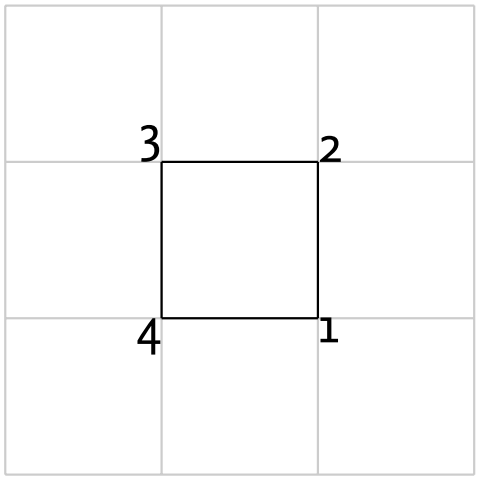
Canvas, XYZ, and Polar



LaTeX Input

```
\begin{tikzpicture}[>=angle 90,thick]
\draw[help lines] (-1,-1) grid (2,3);
\draw[red] (canvas cs:x=1cm,y=2cm) -- (0,3);
\draw[green,->]  (0,0) -- (xyz cs:x=1,y=0,z=0);
\draw[blue,->]   (0,0) -- (0,1,0);
\draw[orange,->] (0,0) -- (0,0,1);
\draw (canvas polar cs:radius=2cm,angle=30)
        -- (90:2);
\path (0,0) coordinate (origin);
\draw (origin) node circle (2pt);
\end{tikzpicture}
```

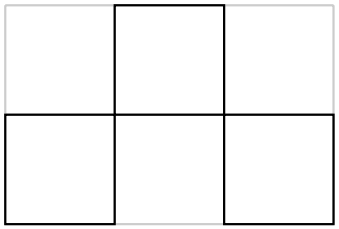
Intersection Coordinate Systems



LaTeX Input

```
\draw[help lines] (0,0) grid +(3,3);
\path (1,1) coordinate (ll);
\path (2,2) coordinate (ur);
\draw (ll) -- (ll -| ur) node[anchor=north west] {1};
\draw (ll -| ur) -- (ur) node[anchor=south west] {2};
\draw (ur) -- (ur -| ll) node[anchor=south east] {3};
\draw (ur -| ll) -- (ll) node[anchor=north east] {4};
```

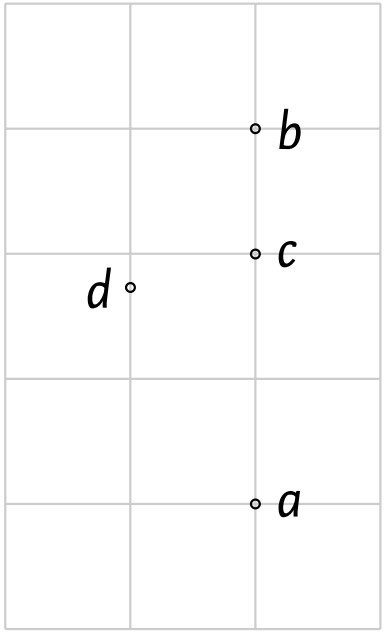
Relative and Incremental Coordinates



LaTeX Input

```
\draw[help lines] (0,0) grid +(3,2);
\draw (0,0) --   (+1,0) --
      (1,1) --   (+0,1) -- cycle;
\draw (1,1) --  +(+1,0) --
     +(1,1) --  +(+0,1) -- cycle;
\draw (2,0) -- ++(+1,0) --
    ++(0,1) -- ++(-1,0) -- cycle;
```

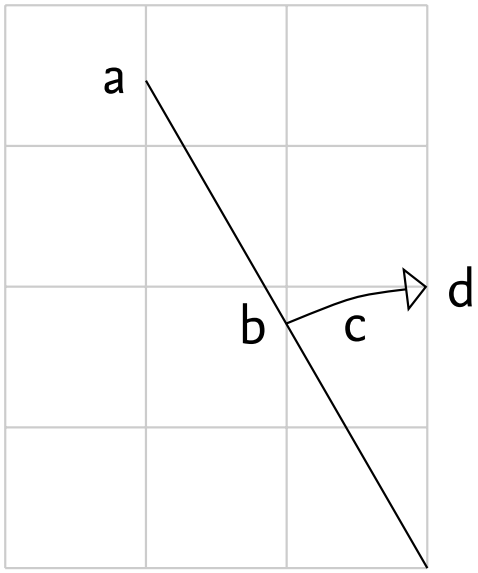
Coordinate Computations: Partway Modifiers



LaTeX Input

```
\draw[help lines] (0,0) grid +(3,5);
\draw (2.0,1.0)            circle (1pt)
                           coordinate(a)
                           node[anchor=west] {$a$}
      (2.0,4.0)            circle (1pt)
                           coordinate(b)
                           node[anchor=west] {$b$}
      ($(a)!0.666!(b)$)    circle (1pt)
                           node[anchor=west] {$c$}
      ($(a)!0.666!30:(b)$) circle (1pt)
                           node[anchor=east] {$d$};
```

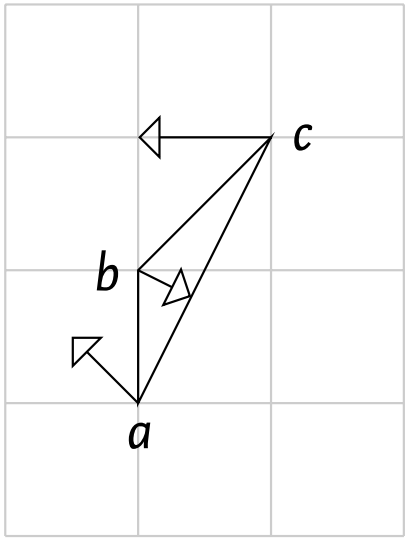
Coordinate Computations: Distance Modifiers



LaTeX Input

```
\draw[help lines] (-3,0) grid +(3,4);
\draw (0,0) --
      ($(0,0)!  1! 30:(0,4)$) coordinate(a) node[anchor=east]  {a}
      ($(0,0)!2cm!    (a)$)   coordinate(b) node[anchor=east]  {b}
      ($(0,0)!2cm!-15:(a)$)   coordinate(c) node[anchor=north] {c}
      ($(0,0)!2cm!-30:(a)$)   coordinate(d) node[anchor=west]  {d};
\draw[-open triangle 90]
      (b) .. controls (c) .. (d);
```

Coordinate Computations: Projection Modifiers



LaTeX Input

```
\begin{tikzpicture}[>=open triangle 90]
\draw[help lines] (0,0) grid +(3,4);
\draw    (1,1) coordinate(a) node[anchor=north] {$a$}
      -- (1,2) coordinate(b) node[anchor=east]  {$b$}
      -- (2,3) coordinate(c) node[anchor=west]  {$c$}
      -- cycle;
\draw[->]  (b) -- ($(a)!(b)!(c)$);
\draw[->]  (c) -- ($(b)!(c)!(a)$);
\draw[->]  (a) -- ($(c)!(a)!(b)$);
\end{tikzpicture}
```

Structuring Pictures with Styles

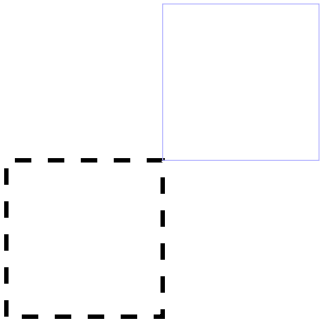
control	Let style make things stand out more/less.
consistency	Guarantees consistent appearance.
reusability	Define style once, use several times.
simplicity	Easier to use. Avoids errors.
refinement	Allows stepwise refinement.
maintenance	Make easy changes which global effect.

Using Styles: \tikzset

LaTeX Input

```
\tikzset{Cork/.style={red,dashed,thick}}
\draw[Cork] (0,0) rectangle (1,1);
```

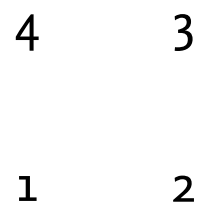
Local Style Changes



LaTeX Input

```
\tikzset{thick dashed/.style={thick,dashed}}
\begin{tikzpicture}[{help lines/.style={ultra thin,blue!30}]
\draw[thick dashed] (0,0) rectangle (1,1);
\draw[help lines]   (1,1) rectangle (2,2);
\end{tikzpicture}
```

The foreach Command



LaTeX Input

```
\foreach \pos/\text in {{0,0}/1,
                        {1,0}/2,
                        {1,1}/3,
                        {0,1}/4} {
   \draw (\pos) node {\text};
}
```

More Examples

Command	Yields
`\foreach \x in {1,2,...,6} {\x,}`	1, 2, 3, 4, 5, 6,
`\foreach \x in {1,3,...,10} {\x,}`	1, 3, 5, 7, 9,
`\foreach \x in {1,3,...,11} {\x,}`	1, 3, 5, 7, 9, 11,
`\foreach \x in {0,0.1,...,0.3} {\x,}`	0, 0.1, 0.20001, 0.30002,
`\foreach \x in {a,b,...,d,9,8,...,6} {\x,}`	a, b, c, d, 9, 8, 7, 6,
`\foreach \x in {7,5,...,0} {\x,}`	7, 5, 3, 1,
`\foreach \x in {Z,X,...,M} {\x,}`	Z, X, V, T, R, P, N,
`\foreach \x in {1,...,5} {\x,}`	1, 2, 3, 4, 5,
`\foreach \x in {5,...,1} {\x,}`	5, 4, 3, 2, 1,
`\foreach \x in {a,...,e} {\x,}`	a, b, c, d, e,
`\foreach \x in {2^1,2^...,2^6} {$\x$,}`	2^1, 2^2, 2^3, 2^4, 2^5, 2^6
`\foreach \x in {0\pi,0.5\pi,...\pi,2\pi} {$\x$,}`	0π, 0.5π, 1.5π, 2.0π,
`\foreach \x in {A_1,..._1,D_1} {$\x$,}`	A_1, B_1, C_1, D_1,

Bibliography

Acronyms and Abbreviations

AMS	American Mathematical Society
API	Application Programming Interface
APL	A Programming Language
CTAN	Comprehensive TEX Archive Network
CD	Compact Disk
FAQ	Frequently Asked Question
GUI	Graphical User Interface
IDE	Integrated Development Environment
ISBN	International Standard Book Number
OS	Operating System
SI	Système International d'Unités/International System of Units
TUG	TEX Users Group
URL	Uniform Resource Locator
WYSIWYG	What You See Is What You Get

About this Document

- This document was created with `pdflatex`.
- The LaTeX document class is `beamer`.