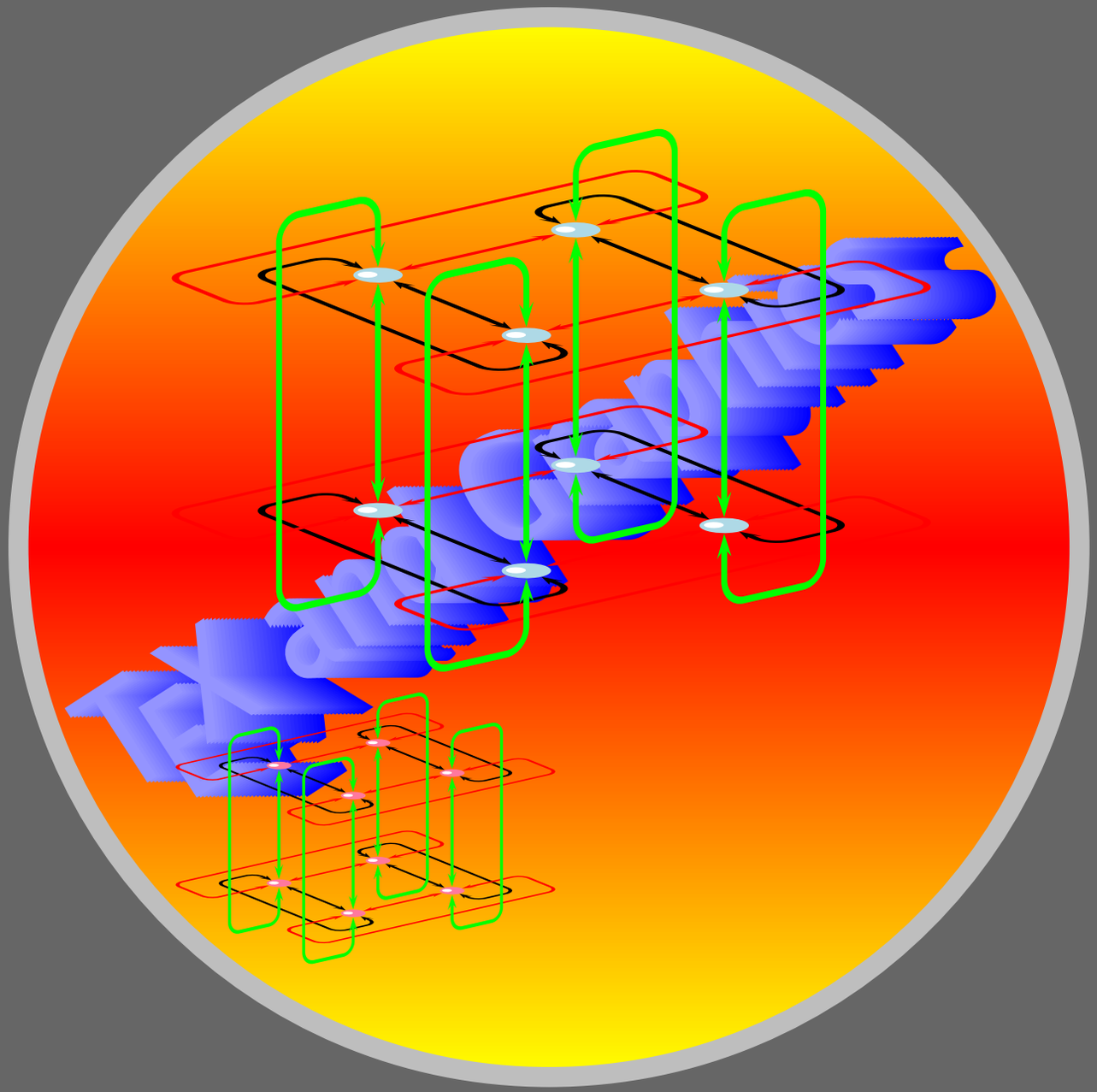


Placing Things

Placing and rotating PSTricks objects
Placing and rotating TEX objects
Putting labels

Online LATEX Tutorial
Part II – Graphics
PSTricks

E Krishnan, CV Radhakrishnan and AJ Alex constitute the graphics tutorial team. Comments and suggestions may be mailed to tutorialteam@tug.org.in

©2002, 2003, The Indian TEX Users Group
This document is generated by PDFTEX with hyperref, pstricks, pdftricks and pdfscreen packages in an intel PC running GNU/LINUX and is released under LPPL

The Indian TEX Users Group
Floor III, SJP Buildings, Cotton Hills
Trivandrum 695014, INDIA
http://www.tug.org.in

6. Placing Things

PSTricks have several commands to place graphics objects and normal LaTeX material where you want and rotate them, if need be. We discuss some of these in this chapter. Other such commands will be described in the chapter on nodes and connections.

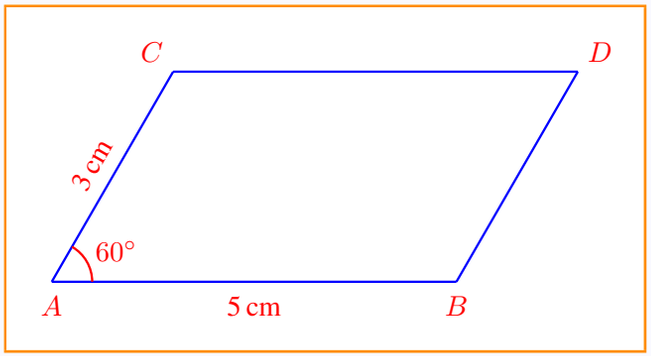
E Krishnan, CV Radhakrishnan and AJ Alex constitute the graphics tutorial team.
Comments and suggestions may be mailed to tutorialteam@tug.org.in

©2002, 2003, The Indian TeX Users Group
This document is generated by PDFTeX with hyperref, pstricks, pdftricks and pdfscreen packages in an intel PC running GNU/LINUX and is released under LPPL

The Indian TeX Users Group
Floor III, SJP Buildings, Cotton Hills
Trivandrum 695014, INDIA
http://www.tug.org.in

6.1. Placing and rotating PSTricks objects

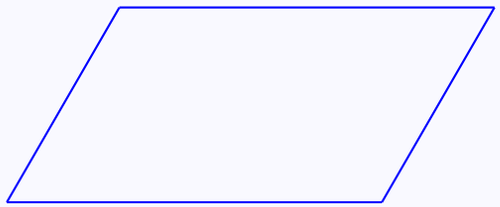
Suppose we want to draw a picture like the one given below using PSTricks:



Taking A as $(0,0)$, we can easily specify B as $(5,0)$ and C as $(3;60)$. (Remember `\SpecialCoor` of the last chapter?). Again, by shifting the origin to B, by setting `origin={-5,0}`, we can specify D as $(3;60)$. And then AB, AC and BD can be drawn with `\psline`. But how do we draw CD? Recall that C and D are specified using different coordinate systems.

One way out of this difficulty is, instead of trying to join C and D, to shift the line AB to the position of BD. The `\rput` command in PSTricks fills the bill here. Thus the above figure (without the labels) can be drawn thus:

```
\begin{pspicture}(0,0)(6,2.5)
  \psset{linecolor=Blue}
  \psline(0,0)(5,0)
  \SpecialCoor
  \psline(0,0)(3;60)
  \rput(3;60){\psline(0,0)(5,0)}
  \psline[origin={-5,0}](0,0)(3;60)
\end{pspicture}
```



Look at the way the `\rput` command is used. The `\psline(0,0)(5,0)` is

E Krishnan, CV Radhakrishnan and AJ Alex constitute the graphics tutorial team. Comments and suggestions may be mailed to tutorialteam@tug.org.in

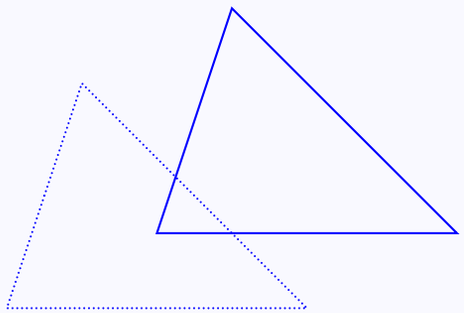
©2002, 2003, The Indian TEX Users Group
This document is generated by PDFTEX with hyperref, pstricks, pdftricks and pdfscreen packages in an intel PC running GNU/LINUX and is released under LPPL

The Indian TEX Users Group
Floor III, SJP Buildings, Cotton Hills
Trivandrum 695014, INDIA
http://www.tug.org.in

the object to place and the coordinates (3;60) is position to place the point (0, 0) of the line (that is the object).

As another example, let's draw a figure we did in the last chapter by shifting the origin this time using the \rput command:

```
\begin{pspicture}(0,-0.5)(6,4.5)
  \begin{pspicture}(0,0)(6,4)
    \pspolygon[linestyle=dotted,%
               dotsep=1pt,%
               linecolor=Blue]%
               (0,0)(4,0)(1,3)
    \rput(2,1){%
      \pspolygon[linecolor=Blue]%
                 (0,0)(4,0)(1,3)}
  \end{pspicture}
\end{pspicture}
```



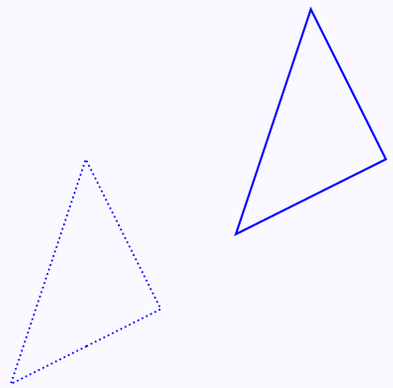
It must be carefully noted that *within* a command \rput(*coordinates*), a new coordinate system is used, with origin at the point with the specified *coordinates*. In other words, what the command \rput(*coordinates*){*object*} does is to *draw* the *object* with reference to a new coordinate system with the axes translated to the point with *coordinates* as the origin. The example below illustrates this:

E Krishnan, CV Radhakrishnan and AJ Alex constitute the graphics tutorial team. Comments and suggestions may be mailed to tutorialteam@tug.org.in

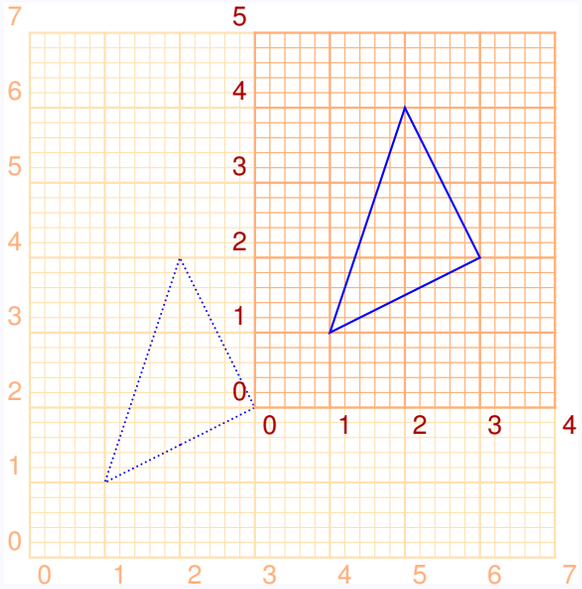
©2002, 2003, The Indian TEX Users Group
This document is generated by PDFTEX with hyperref, pstricks, pdftricks and pdfscreen packages in an intel PC running GNU/LINUX and is released under LPPL

The Indian TEX Users Group
Floor III, SJP Buildings, Cotton Hills
Trivandrum 695014, INDIA
http://www.tug.org.in

```
\begin{pspicture}(0,1)(6,6)
  \pspolygon[linecolor=Blue,%
             linestyle=dotted,%
             dotsep=1pt]%
             (1,1)(2,4)(3,2)
  \rput(3,2){%
    \pspolygon[linecolor=Blue]%
               (1,1)(2,4)(3,2)}
\end{pspicture}
```



The same picture is shown below with the original coordinate system in a lighter shade and that within the `\rput(3,2)` in a darker shade, to clarify this idea:



Another interesting feature of the `\rput` command is that it can be given an optional argument to *rotate* the object it puts in place. Thus `\rput{`*angle*`}(`*coordinates*`){`*object*`}`. draws the object with reference to

Online LATEX Tutorial
Part II – Graphics
PSTricks

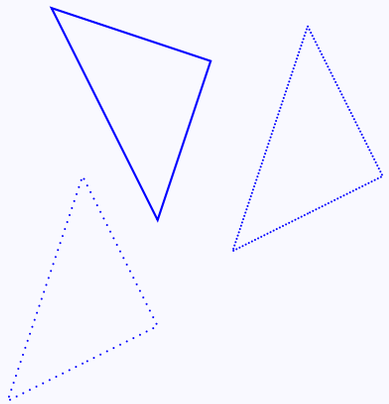
E Krishnan, CV Radhakrishnan and AJ Alex constitute the graphics tutorial team. Comments and suggestions may be mailed to tutorialteam@tug.org.in

©2002, 2003, The Indian TEX Users Group
This document is generated by PDFTEX with hyperref, pstricks, pdftricks and pdfscreen packages in an intel PC running GNU/LINUX and is released under LPPL

The Indian TEX Users Group
Floor III, SJP Buildings, Cotton Hills
Trivandrum 695014, INDIA
http://www.tug.org.in

a new system of axes obtained by first translating the original axes to the point with *coordinates* as origin and then rotating them through *angle*. Look at the next example:

```
\begin{pspicture}(-1,0)(7,9)
  \pspolygon[linecolor=Blue,%
             linestyle=dotted,%
             dotsep=2pt]%
            (1,1)(2,4)(3,2)
  \rput(3,2){%
    \pspolygon[linecolor=Blue,%
               linestyle=dotted,%
               dotsep=0.5pt]%
              (1,1)(2,4)(3,2)}
  \rput{45}(3,2){%
    \pspolygon[linecolor=Blue](1,1)(2,4)(3,2)}
\end{pspicture}
```

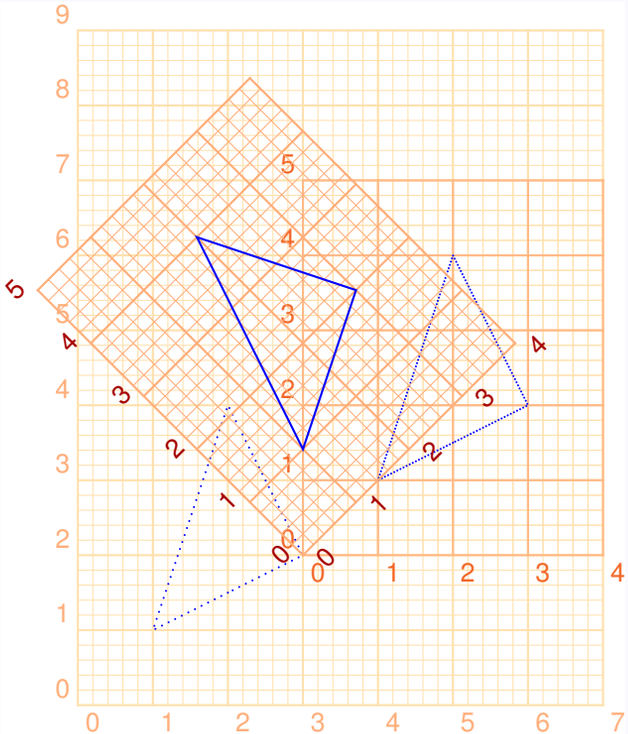


The picture below, which shows the picture above with the different coordinate systems shown in different colors, will make this clear.

E Krishnan, CV Radhakrishnan and AJ Alex constitute the graphics tutorial team. Comments and suggestions may be mailed to tutorialteam@tug.org.in

©2002, 2003, The Indian TEX Users Group
This document is generated by PDFTEX with hyperref, pstricks, pdftricks and pdfscreen packages in an intel PC running GNU/LINUX and is released under LPPL

The Indian TEX Users Group
Floor III, SJP Buildings, Cotton Hills
Trivandrum 695014, INDIA
http://www.tug.org.in



Online LATEX Tutorial
Part II – Graphics
PSTricks

E Krishnan, CV Radhakrishnan and AJ Alex constitute the graphics tutorial team. Comments and suggestions may be mailed to tutorialteam@tug.org.in

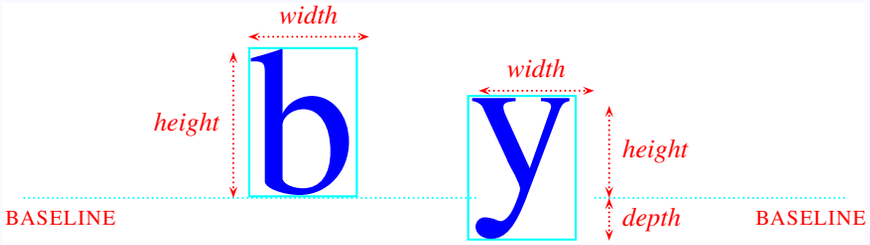
©2002, 2003, The Indian TEX Users Group
This document is generated by PDFTEX with hyperref, pstricks, pdftricks and pdfscreen packages in an intel PC running GNU/LINUX and is released under LPPL

The Indian TEX Users Group
Floor III, SJP Buildings, Cotton Hills
Trivandrum 695014, INDIA
http://www.tug.org.in

6.2. Placing and rotating TEX objects

The `\rput` command can also be used for placing and rotating TEX material, which is usually text of some sort. Before giving examples of this, something must be said about the way TEX does its typesetting job. Internally, TEX does this with *boxes*. For example, when you type `{\Large boy}` in your document, you get boy after TEX has complied it *and* the printer driver has processed it, but what TEX produces internally is something like boy or more precisely, □□□ with instructions on what to put in each box. In other words, individual characters are put in *character boxes* which are stringed together horizontally to form *line boxes* which again are packed vertically into *paragraph boxes* and so on.

Note that in typesetting, the bottom of the letters without *descenders* (such as for example, a, b, c, d, e but not f and g) are aligned along a line. This is called the *baseline*. In TEX every box has a *width* (no explanation needed), *height* (which is its height *above the baseline*) and *depth* (which is the distance it extends below the baseline) as shown below for two character boxes:

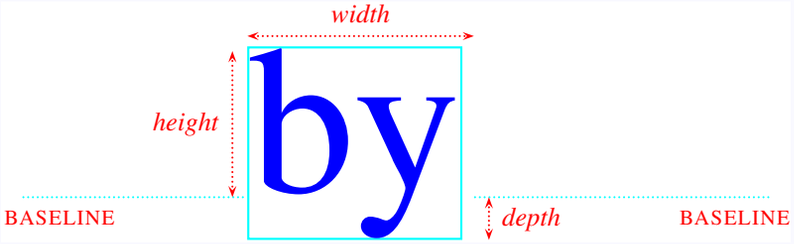


Thus the “b” box has 0 depth, while the “y” box a positive depth. Note that when these are joined to form the word “by”, the resulting box has positive depth, equal to the depth of “y”

E Krishnan, CV Radhakrishnan and AJ Alex constitute the graphics tutorial team. Comments and suggestions may be mailed to `tutorialteam@tug.org.in`

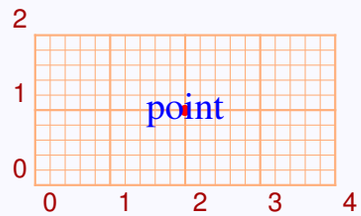
©2002, 2003, The Indian TEX Users Group
This document is generated by PDFTEX with hyperref, pstricks, pdftricks and pdfscreen packages in an intel PC running GNU/LINUX and is released under LPPL

The Indian TEX Users Group
Floor III, SJP Buildings, Cotton Hills
Trivandrum 695014, INDIA
`http://www.tug.org.in`

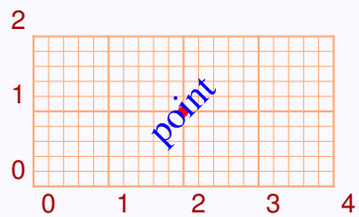


Now when we give a command `\rput(`*coordinate*`){`*text*`}`, TEX makes a box to fit *text* and `\rput` places the center of the box at the point with specified *coordinates*. Also, any rotation specified in the `\rput` command is about this point, as shown below:

```
\psdots[linecolor=Red,dotsize=0 5](2,1)
\rput(2,1){\Large\color{Blue} point}
```




```
\psdots[linecolor=Red,dotsize=0 5](2,1)
\rput{45}(2,1){\Large\color{Blue} point}
```



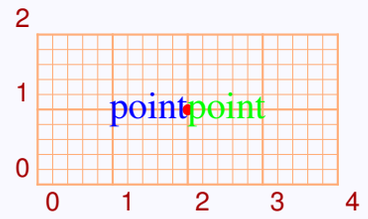
The textbox can be placed with the the midpoint of its left or right edge on the specified point (and then rotated about this point, if desired) using the optional arguments `l` or `r` immediately after `\rput` as shown below:

E Krishnan, CV Radhakrishnan and AJ Alex constitute the graphics tutorial team. Comments and suggestions may be mailed to tutorialteam@tug.org.in

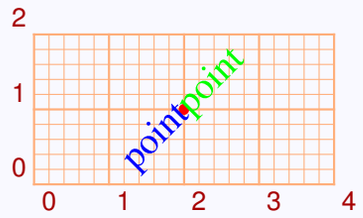
©2002, 2003, The Indian TEX Users Group
This document is generated by PDFTEX with hyperref, pstricks, pdftricks and pdfscreen packages in an intel PC running GNU/LINUX and is released under LPPL

The Indian TEX Users Group
Floor III, SJP Buildings, Cotton Hills
Trivandrum 695014, INDIA
http://www.tug.org.in

```
\psdots[linecolor=Red,dotsize=0 5](2,1)
\rput[l](2,){\Large\color{Green}point}
\rput[r](2,1){\Large\color{Blue}point}
```

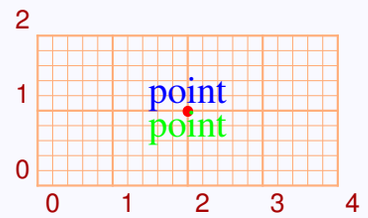



```
\psdots[linecolor=Red,dotsize=0 5](2,1)
\rput[l]{45}(2,){\Large\color{Green}point}
\rput[r]{45}(2,1){\Large\color{Blue}point}
```

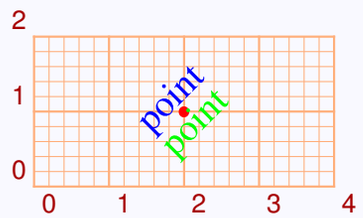


On the other hand, the textbox can be placed with the midpoint of its top or bottom edge at the specified point (and rotated about this point) using the `t` or `b` option.

```
\psdots[linecolor=Red,dotsize=0 5](2,1)
\rput[t](2,){\Large\color{Green}point}
\rput[b](2,1){\Large\color{Blue}point}
```




```
\psdots[linecolor=Red,dotsize=0 5](2,1)
\rput[t]{45}(2,){\Large\color{Green}point}
\rput[b]{45}(2,1){\Large\color{Blue}point}
```



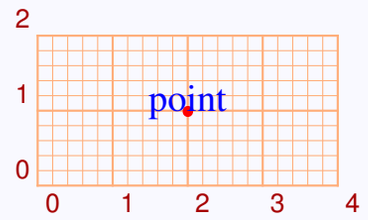
E Krishnan, CV Radhakrishnan and AJ Alex constitute the graphics tutorial team. Comments and suggestions may be mailed to tutorialteam@tug.org.in

©2002, 2003, The Indian TEX Users Group
This document is generated by PDFTEX with hyperref, pstricks, pdftricks and pdfscreen packages in an intel PC running GNU/LINUX and is released under LPPL

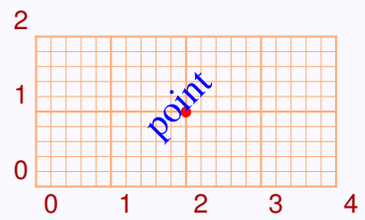
The Indian TEX Users Group
Floor III, SJP Buildings, Cotton Hills
Trivandrum 695014, INDIA
http://www.tug.org.in

For vertically shifting a textbox from its default placement, there is another option `B`, which places the box with the midpoint of the baseline at the specified point as in the picture below:

```
\psdots[linecolor=Red,dotsize=0 5](2,1)
\rput[B](2,){\Large\color{Blue}point}
```




```
\psdots[linecolor=Red,dotsize=0 5](2,1)
\rput[B]{45}(2,){\Large\color{Blue}point}
```



Using the combinations `bl`, `br`, `tl`, `tr`, the textbox can also be placed with any of its corners at the specified point (and rotated too).

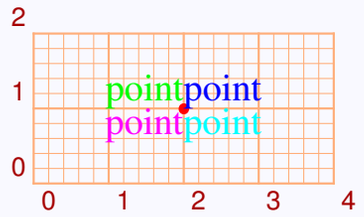
Online LATEX Tutorial
Part II – Graphics
PSTricks

E Krishnan, CV Radhakrishnan and AJ Alex constitute the graphics tutorial team. Comments and suggestions may be mailed to tutorialteam@tug.org.in

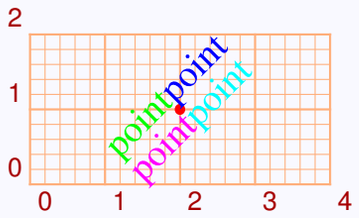
©2002, 2003, The Indian TEX Users Group
This document is generated by PDFTEX with hyperref, pstricks, pdftricks and pdfscreen packages in an intel PC running GNU/LINUX and is released under LPPL

The Indian TEX Users Group
Floor III, SJP Buildings, Cotton Hills
Trivandrum 695014, INDIA
http://www.tug.org.in

```
\psdots[linecolor=red,dotsize=0 5](2,1)
\rput[bl](2,1){\Large\color{Blue} point}
\rput[br](2,1){\Large\color{Green} point}
\rput[tl](2,1){\Large\color{Cyan} point}
\rput[tr](2,1){\Large\color{Magenta} point}
```

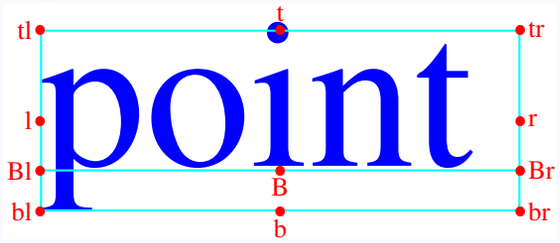



```
\psdots[linecolor=red,dotsize=0 5](2,1)
\rput[bl]{45}(2,1){\Large\color{Blue} point}
\rput[br]{45}(2,1){\Large\color{Green} point}
\rput[tl]{45}(2,1){\Large\color{Cyan} point}
\rput[tr]{45}(2,1){\Large\color{Magenta} point}
```



The combinations `Br` and `Bl` can also be used. (Try them!)

The picture below shows those points on the textbox placed at the specified point for various positional arguments to the `\rput` command:



Thus the full syntax of the command is

`\rput[`*boxpoint*`]{`*angle*`}{`*coordinates*`}{`*text*`}`

Since the pure graphic objects produced by `PSTricks` are zero-dimensional TEX boxes, the *boxpoint* argument is redundant for placing such objects.

As an example, let's see how we can put labels on the pie-chart drawn in the last chapter:

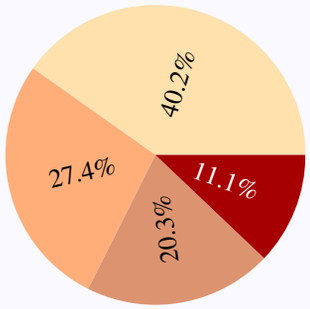
Online LATEX Tutorial
Part II – Graphics
PSTricks

E Krishnan, CV Radhakrishnan and AJ Alex constitute the graphics tutorial team. Comments and suggestions may be mailed to tutorialteam@tug.org.in

©2002, 2003, The Indian TEX Users Group
This document is generated by PDFTEX with hyperref, pstricks, pdftricks and pdfscreen packages in an intel PC running GNU/LINUX and is released under LPPL

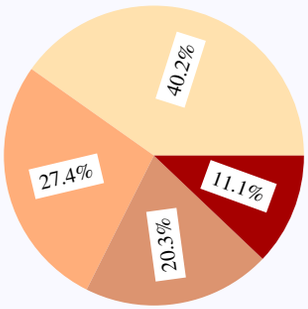
The Indian TEX Users Group
Floor III, SJP Buildings, Cotton Hills
Trivandrum 695014, INDIA
http://www.tug.org.in

```
\definecolor{PaleApricot}{cmyk}{0,0.12,0.32,0}
\begin{pspicture}(-2,-2)(2,2)
   \degrees[100]
   \SpecialCoor
   \pswedge*[linecolor=PaleApricot](0,0){2}{0}{40.2}
   \pswedge*[linecolor=Apricot](0,0){2}{40.2}{67.6}
   \pswedge*[linecolor=Tan](0,0){2}{67.6}{87.9}
   \pswedge*[linecolor=Mahogany](0,0){2}{87.9}{100}
   \rput{20.1}(1;20.2){40.2\%}
   \rput{3.7}(1;53.7){27.4\%}
   \rput{27.15}(1;77.15){20.3\%}
   \rput{-5.55}(1;-5.55){\color{White} 11.1\%}
\end{pspicture}
```



The `\rput` has also a starred form, in which the text is first put into a white box (actually a `\psframebox*`, but that can wait) and then placed in the required point, blotting out whatever is behind. For example, the pie-chart above can be labeled differently as below:

```
\definecolor{PaleApricot}{cmyk}{0,0.12,0.32,0}
\begin{pspicture}(-2,-2)(2,2)
   \degrees[100]
   \SpecialCoor
   \pswedge*[linecolor=PaleApricot](0,0){2}{0}{40.2}
   \pswedge*[linecolor=Apricot](0,0){2}{40.2}{67.6}
   \pswedge*[linecolor=Tan](0,0){2}{67.6}{87.9}
   \pswedge*[linecolor=Mahogany](0,0){2}{87.9}{100}
   \rput*{20.1}(1;20.2){40.2\%}
   \rput*{3.7}(1;53.7){27.4\%}
   \rput*{27.15}(1;77.15){20.3\%}
   \rput*{-5.55}(1;-5.55){11.1\%}
\end{pspicture}
```



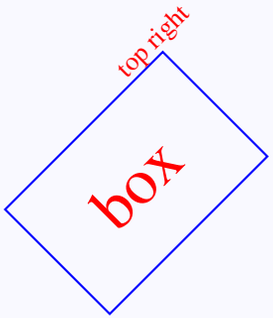
The `\rput` commands can be nested. Look at this example:

E Krishnan, CV Radhakrishnan and AJ Alex constitute the graphics tutorial team. Comments and suggestions may be mailed to tutorialteam@tug.org.in

©2002, 2003, The Indian TEX Users Group
This document is generated by PDFTEX with hyperref, pstricks, pdftricks and pdfscreen packages in an intel PC running GNU/LINUX and is released under LPPL

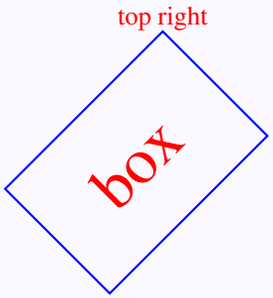
The Indian TEX Users Group
Floor III, SJP Buildings, Cotton Hills
Trivandrum 695014, INDIA
http://www.tug.org.in

```
\definecolor{PaleApricot}{cmyk}{0,0.12,0.32,0}
\begin{pspicture}(0,0)(4,4)
  \rput[bl]{45}(0,0){%
    \psframe[linecolor=Blue](0,0)(3,2)
    \rput(1.5,1){\color{Red}\Huge box}
    \rput[b](3,2){\color{Red}top right}}
\end{pspicture}
```



The above picture will look better with the text “top right” set horizontally (that is, without the rotation through 45°. For this, the rotation due to the outer `\rput` has to be reset. This can be done by giving the angle argument `*0` to the inner `\rput`, where the `*` option resets the previous rotation.

```
\definecolor{PaleApricot}{cmyk}{0,0.12,0.32,0}
\begin{pspicture}(0,0)(4,4)
  \rput[bl]{45}(0,0){%
    \psframe[linecolor=Blue](0,0)(3,2)
    \rput(1.5,1){\color{Red}\Huge box}
    \rput[b]{*0}(3,2){\color{Red}top right}}
\end{pspicture}
```



Some of the frequently required angles for rotation have shorter representation as letters, listed in the table below:

E Krishnan, CV Radhakrishnan and AJ Alex constitute the graphics tutorial team. Comments and suggestions may be mailed to tutorialteam@tug.org.in

©2002, 2003, The Indian TEX Users Group
This document is generated by PDFTEX with hyperref, pstricks, pdftricks and pdfscreen packages in an intel PC running GNU/LINUX and is released under LPPL

The Indian TEX Users Group
Floor III, SJP Buildings, Cotton Hills
Trivandrum 695014, INDIA
http://www.tug.org.in

ANGLE	LETTER	MEANING	ANGLE	LETTER	MEANING
0	U	Up	*0	N	North
90	L	Left	*90	W	West
180	D	Down	*180	S	South
270	R	Right	*270	E	East

Thus for example, instead of `\rput[90]`, we can also say `\rput[L}`.

Online LATEX Tutorial
Part II – Graphics
PSTricks

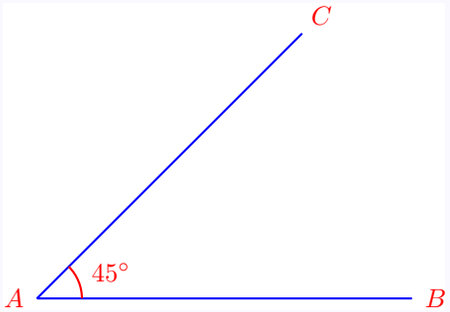
E Krishnan, CV Radhakrishnan and AJ Alex constitute the graphics tutorial team. Comments and suggestions may be mailed to tutorialteam@tug.org.in

©2002, 2003, The Indian TEX Users Group
This document is generated by PDFTEX with hyperref, pstricks, pdftricks and pdfscreen packages in an intel PC running GNU/LINUX and is released under LPPL

The Indian TEX Users Group
Floor III, SJP Buildings, Cotton Hills
Trivandrum 695014, INDIA
http://www.tug.org.in

6.3. Putting labels

One of the frequent needs for placing objects in specific locations is in affixing labels to parts of technical (especially mathematical) drawings. Consider the simple example shown below:



If we try to get this with something like

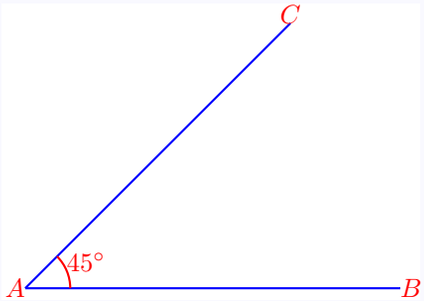
```
\begin{pspicture}(0,-1)(6,5)
  \psset{linecolor=Blue}
  \psline(0,0)(5,0)
  \SpecialCoor
  \psline(0,0)(5;45)
  \rput[r](0,0){\color{Red} $A$}
  \rput[l](5,0){\color{Red} $B$}
  \rput[b](5;45){\color{Red} $C$}
  \psarc[linecolor=Red](0,0){0.6}{0}{45}
  \rput[bl](0.6;22.5){\color{Red} $45^\circ$}
\end{pspicture}
```

we get

E Krishnan, CV Radhakrishnan and AJ Alex constitute the graphics tutorial team. Comments and suggestions may be mailed to tutorialteam@tug.org.in

©2002, 2003, The Indian TEX Users Group
This document is generated by PDFTEX with hyperref, pstricks, pdftricks and pdfscreen packages in an intel PC running GNU/LINUX and is released under LPPL

The Indian TEX Users Group
Floor III, SJP Buildings, Cotton Hills
Trivandrum 695014, INDIA
http://www.tug.org.in



where the labels sort of stick to the points referred to. So, to `\rput` labels, we have to compute the coordinates of points a little away from the points referred to. The command `\uput` saves us this trouble. Thus the command

`\uput[`*dirangle*`](`*coordinates*`){`*label*`}`

puts *label* at a distance of 5 pt away from the point with the specified *coordinates* in the direction specified by *dirangle*. thus the first figure above is produced by

```
\begin{center}
  \begin{pspicture}(0,-1)(6,3.5)
    \psset{linecolor=Blue}
    \psline(0,0)(5,0)\SpecialCoor
    \psline(0,0)(5;45)
    \uput[180](0,0){\color{Red} $A$}
    \uput[0](5,0){\color{Red} $B$}
    \uput[45](5;45){\color{Red} $C$}
    \psarc[linecolor=Red](0,0){0.6}{0}{45}
    \uput[30](0.6;22.5){\color{Red} $45^\circ$}
  \end{pspicture}
\end{center}
```

The distance between the point and the label is controlled by the parameter `labelsep`. We can change its default value (5 pt) by using `psset`. We can

E Krishnan, CV Radhakrishnan and AJ Alex constitute the graphics tutorial team. Comments and suggestions may be mailed to tutorialteam@tug.org.in

©2002, 2003, The Indian TEX Users Group
This document is generated by PDFTEX with hyperref, pstricks, pdftricks and pdfscreen packages in an intel PC running GNU/LINUX and is released under LPPL

The Indian TEX Users Group
Floor III, SJP Buildings, Cotton Hills
Trivandrum 695014, INDIA
http://www.tug.org.in

also include this value as the first argument to `\uput` itself. Thus in the above picture, if we want the label 45° to be a little closer to the arc, then we change the last line in the above code as

```
\uput{3pt}[30](0.6;30){\color{Red} $45^\circ$}
```

(Try it!) Not that the value of `labelsep` is to be given within curly braces `{ }`.

As in the case of rotation angles in `\rput`, the frequently used direction angles in `\uput` also has letter abbreviations.

ANGLE	LETTER	MEANING	ANGLE	LETTER	MEANING
0	`r`	right	45	`ur`	up-right
90	`u`	up	135	`ul`	up-left
180	`l`	left	225	`dl`	down-left
270	`d`	down	315	`dr`	down-right

Thus the code for the first picture above can also be written

```
\begin{center}
  \begin{pspicture}(0,-1)(6,3.5)
    \psset{linecolor=Blue}
    \psline(0,0)(5,0)\SpecialCoor
    \psline(0,0)(5;45)
    \uput[l](0,0){\color{Red} $A$}
    \uput[r](5,0){\color{Red} $B$}
    \uput[ur](5;45){\color{Red} $C$}
    \psarc[linecolor=Red](0,0){0.6}{0}{45}
    \uput[22.5](0.6;22.5){\color{Red} $45^\circ$}
  \end{pspicture}
\end{center}
```

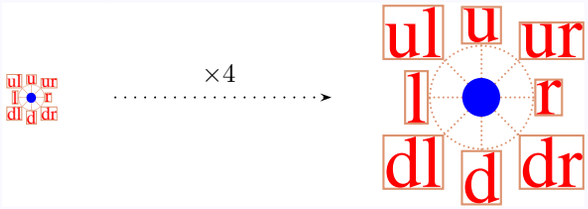

Online LATEX Tutorial
Part II – Graphics
PSTricks

E Krishnan, CV Radhakrishnan and AJ Alex constitute the graphics tutorial team. Comments and suggestions may be mailed to tutorialteam@tug.org.in

©2002, 2003, The Indian TEX Users Group
This document is generated by PDFTEX with hyperref, pstricks, pdftricks and pdfscreen packages in an intel PC running GNU/LINUX and is released under LPPL

The Indian TEX Users Group
Floor III, SJP Buildings, Cotton Hills
Trivandrum 695014, INDIA
http://www.tug.org.in

We show below The positions of the *label boxes* with direction specified by all these letters relative to a specified point, both in actual size and magnified four-fold (using the `\tiny` font for labels):



The `\uput` command also has an argument to rotate the object placed and this argument is to be included in curly braces `{ }`, between the *dirangle* and *coordinate* specifications. Thus the full form of the command `\uput` is

`\uput{`*labelsep*`}[`*dirangle*`]{`*rotangle*`}(`*coordinates*`){`*label*`}`

The example below shows the use of rotated labels with `\uput`

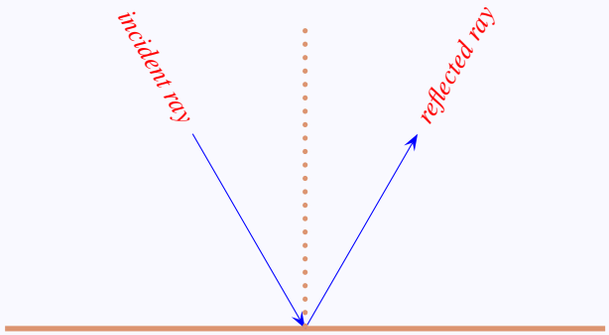
Online LATEX Tutorial
Part II – Graphics
PSTricks

E Krishnan, CV Radhakrishnan and AJ Alex constitute the graphics tutorial team. Comments and suggestions may be mailed to tutorialteam@tug.org.in

©2002, 2003, The Indian TEX Users Group
This document is generated by PDFTEX with hyperref, pstricks, pdftricks and pdfscreen packages in an intel PC running GNU/LINUX and is released under LPPL

The Indian TEX Users Group
Floor III, SJP Buildings, Cotton Hills
Trivandrum 695014, INDIA
http://www.tug.org.in

```
\begin{pspicture}(-4,0)(4,5)
  \SpecialCoor
  \psline[linecolor=Tan,%
          linewidth=2pt]%
         (-4,0)(4,0)
  \psline[linecolor=Blue,%
          linewidth=0.4pt,%
          arrowsize=3pt 4]{->}%
         (3;120)(0,0)
  \psline[linecolor=Blue,%
          linewidth=0.4pt,%
          arrowsize=3pt 4]{->}%
         (0,0)(3;60)
  \psline[linecolor=Tan,%
          linewidth=2pt,%
          linestyle=dotted]%
          (0,0)(0,4)
  \uput[120]{300}(3;120){%
    \color{Red}\textit{incident ray}}
  \uput[60]{60}(3;60){%
    \color{Red}\textit{reflected ray}}
\end{pspicture}
```



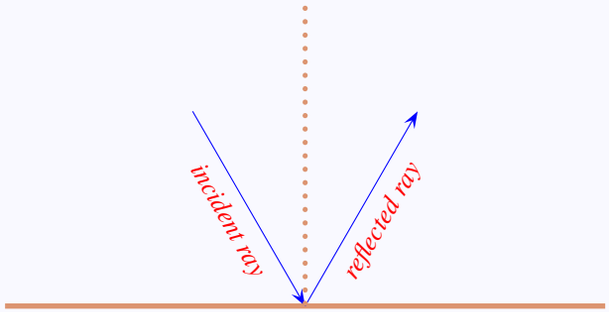
Sometimes for the correct placement of labels, a combination of `\rput` and `\uput` maybe the easiest solution, as in the next example:

E Krishnan, CV Radhakrishnan and AJ Alex constitute the graphics tutorial team. Comments and suggestions may be mailed to tutorialteam@tug.org.in

©2002, 2003, The Indian TEX Users Group
This document is generated by PDFTEX with hyperref, pstricks, pdftricks and pdfscreen packages in an intel PC running GNU/LINUX and is released under LPPL

The Indian TEX Users Group
Floor III, SJP Buildings, Cotton Hills
Trivandrum 695014, INDIA
http://www.tug.org.in

```
\begin{pspicture}(-4,0)(4,3)
  \SpecialCoor
  \psline[linecolor=Tan,%
          linewidth=2pt]%
         (-4,0)(4,0)
  \psline[linecolor=Blue,%
          linewidth=0.4pt,%
          arrowsize=3pt 4]{->}%
         (3;120)(0,0)
  \psline[linecolor=Blue,%
          linewidth=0.4pt,%
          arrowsize=3pt 4]{->}%
         (0,0)(3;60)
  \psline[linecolor=Tan,%
          linewidth=2pt,%
          linestyle=dotted]%
          (0,0)(0,4)
  \rput[b]{300}(1.5;120){%
    \uput[d](0,0){\color{Red}%
      \textit{incident ray}}}
  \rput[b]{60}(1.5;60){%
    \uput[d](0,0){\color{Red}%
      \textit{reflected ray}}}
\end{pspicture}
```



Now we can give the complete code for the first picture of this chapter, labels and all:

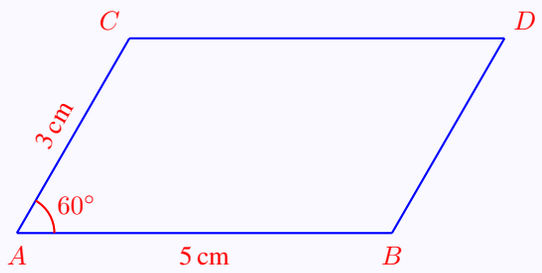
Online LATEX Tutorial
Part II – Graphics
PSTricks

E Krishnan, CV Radhakrishnan and AJ Alex constitute the graphics tutorial team. Comments and suggestions may be mailed to tutorialteam@tug.org.in

©2002, 2003, The Indian TEX Users Group
This document is generated by PDFTEX with hyperref, pstricks, pdftricks and pdfscreen packages in an intel PC running GNU/LINUX and is released under LPPL

The Indian TEX Users Group
Floor III, SJP Buildings, Cotton Hills
Trivandrum 695014, INDIA
http://www.tug.org.in

```
\begin{pspicture}(0,-1)(6,3.5)
  \psset{linecolor=Blue}
  \psline(0,0)(5,0)
  \SpecialCoor
  \psline(0,0)(3;60)
  \rput(3;60){%
    \psline(0,0)(5,0)
    \uput[ur](5,0){%
      \color{Red} $D$}}
  \psline[origin={-5,0}]%
         (0,0)(3;60)
  \psarc[linecolor=Red]%
        (0,0){0.5}{0}{60}
  \uput{0.6}[30](0,0){%
    \color{Red} $60^\circ$}
  \uput[d](2.5,0){%
    \color{Red} 5\,cm}
  \rput[b]{60}(1.5;60){%
    \uput[u](0,0){%
      \color{Red} 3\,cm}}
  \uput[d](0,0){\color{Red} $A$}
  \uput[d](5,0){\color{Red} $B$}
  \uput[ul](3;60){\color{Red} $C$}
\end{pspicture}
```



Online LATEX Tutorial
Part II – Graphics
PSTricks

E Krishnan, CV Radhakrishnan and AJ Alex constitute the graphics tutorial team. Comments and suggestions may be mailed to tutorialteam@tug.org.in

©2002, 2003, The Indian TEX Users Group
This document is generated by PDFTEX with hyperref, pstricks, pdftricks and pdfscreen packages in an intel PC running GNU/LINUX and is released under LPPL

The Indian TEX Users Group
Floor III, SJP Buildings, Cotton Hills
Trivandrum 695014, INDIA
http://www.tug.org.in