‘infix-RPN’ – ‘pst-infixplot’

ver. 0.11

Jean-Côme CHARPENTIER <jean-come.charpentier@wanadoo.fr>
Christophe JORSSEN <christophe.jorssen@libre.fr.invalid>
‘libre’ is the french word for ‘free’

2005/05/16

Abstract
Plotting functions with \psplot is very powerful but sometimes difficult to learn since the syntax of \psplot
and \parametricplot requires some PostScript knowledge. What ‘infix-RPN’ and ‘pst-infixplot’ intend to do is
to simplify the usage of \psplot for the beginner, providing macro commands that convert natural mathematical
expressions to PS syntax.

1 Basic examples: usage of ‘infix-RPN’
\usepackage{infix-RPN} for \LaTeX\ users or \input infix-RPN.tex for \TeX\ users gives access to three macros:
\infixtoRPN, \RPN\ and \DeclareNewPSOperator.

The macro \infixtoRPN takes an infix expression as argument and converts it to Reverse Polish Notation. The
result of the conversion is put in the macro \RPN.

\begin{verbatim}
2 3 add 4 \* mul sub
x neg log
2 \* y \div sin \* mul
\end{verbatim}

Multiple signs are OK:
\begin{verbatim}
3 0 5 neg neg neg add
\end{verbatim}

For operators that require more than one argument, arguments must be separated with commas:
\begin{verbatim}
x y atan
\end{verbatim}

There is a difference between variables and operators. There are 11 pre-defined operators\footnote{Actually, there are five more operators defined : add, sub, mul, div and exp. Those ones should not be used directly. Use \*, -, *, / and ^ instead, which is, by the way, the main interest of using infix notation.} which are basically
those of PostScript: abs, sin, cos, atan, neg, ceiling, floor, truncate, sqrt, ln, log. You can define more
operators with the \DeclareNewPSOperator macro. Div is a PS operator defined by \psmath\ \begin{verbatim}
x y Div
\end{verbatim}

\begin{verbatim}
\DeclareNewPSOperator{Div}
\end{verbatim}

\begin{verbatim}
\infixtoRPN(Div(x,y))\RPN
\end{verbatim}

\footnote{If you use \psmath with \infix-RPN, PS operators added by \psmath are declared by \DeclareNewPSOperator and are therefore directly
accessible in any infix expression.}
2 Plot examples with ‘infix-RPN’

\begin{verbatim}
\psset{plotpoints=1000}
\psaxes{-}{(0,0)(0,-2)(10,3)}
\infixtoRPN{sqrt(x)}
\psplot[linewidth=1pt,linecolor=green]{0}{10}[\RPN]
\psplot[linewidth=1pt,linecolor=red]{0}{10}[\RPN]
\infixtoRPN{\sin(-x*180/3.1415)}
\psplot[linewidth=1pt,linecolor=blue]{0}{10}[\RPN]
\end{verbatim}

3 Plot examples with ‘pst-infixplot’

If you don’t want the limitation of having to invoke two macro calls (namely \infixtoRPN and \RPN) for plotting, then use the ‘\texttt{pst-infixplot}’ package! \LaTeX\ users should type \texttt{\textbackslash usepackage{pst-infixplot}} in the preamble when \LaTeX\ users should type \texttt{\textbackslash input \texttt{pst-infixplot.tex}}.

‘\texttt{pst-infixplot}’ automatically loads \texttt{pstricks}, \texttt{pstricks-plot} and \texttt{infix-RPN}. ‘\texttt{pst-infixplot}’ defines two macro commands: \texttt{psPlot} and \texttt{par\texttt{ametricPlot}}.

The syntax of \texttt{psPlot} is: \texttt{\textbackslash \texttt{psPlot}}\texttt{[parameters]}\{x begin\}{x end\}\{infix expression\} where the optional parameters are any \texttt{pstricks} valid parameter.

\begin{verbatim}
\psset{plotpoints=1000}
\psaxes{-}{(0,0)(0,-2)(7,3)}
\psPlot[linewidth=1pt,linecolor=green]{0}{7}[\RPN]
\psPlot[linewidth=1pt,linecolor=red]{0}{7}[\RPN]
\psPlot[linewidth=1pt,linecolor=blue]{0}{7}[\RPN]
\psplot{0}{7}{x neg 180 mul 3.1415 div cos}
\end{verbatim}

The syntax of \texttt{par\texttt{ametricPlot}} is:
\texttt{par\texttt{ametricPlot}}\texttt{[parameters]}\{x begin\}{x end\}\{infix x expression\}\{infix y expression\} where the optional parameters are any \texttt{pstricks} valid parameter.

\begin{verbatim}
\psset{plotpoints=1000}
\psaxes{-}{(0,0)(0,-2)(3.8,3)}
\par\texttt{ametricPlot}[linecolor=red]{-30}{70}{3.5*cos(t)}{2.3*sqrt(abs(sin(t)))}
\par\texttt{ametricPlot}[linecolor=green]{-30}{60}{4*sqrt(abs(floor(t)))/10}{t/20}
\par\texttt{ametricplot}{-30}{60}{2 t cos mul 3 t sin mul}
\end{verbatim}