Some informations about LaTeX packages

TeX Users Group

1 What is Metafont?

Metafont, Knuth's font creation program, is independent of TeX. It generates only bitmap fonts (although internally it creates outlines on which the bitmaps are based). There is still research to be done on combining overlapping outlines into a single outline that can be used like the outline of a Type 1 font; Knuth has "frozen" Metafont, so further research and development are being done by someone else, and the result will not be called "Metafont". In fact, it's also possible to use Type 1 fonts with TeX, and nearly all TeX installations routinely use free or commercial Type 1 fonts, especially if they're not producing heavily technical material; only Computer Modern (the font Knuth developed), Lucida Bright, and Times have anywhere near a comprehensive symbol complement, and none of the symbol sets are from "major" font suppliers (too much work, not enough money; the demise of Monotype's symbol catalogmetal onlyis a particularly great loss).

2 Packages

2.1 The minipage environment

Metafont, Knuth's font creation program, is independent of TeX. It generates only bitmap fonts (although internally it creates outlines on which the bitmaps are based). There is still research to be done on combining overlapping outlines into a single outline that can be used like the outline of a Type 1 font; Knuth has "frozen" Metafont, so furthe research and development are being done by someone else, and the result will not be called "Metafont".

In fact, it's also possible to use Type 1 fonts with TeX, and nearly all TeX installations routinely use free or commercial Type 1 fonts, espe- cially if they're not producing heav- ily technical material; only Com- puter Modern (the font Knuth de- veloped), Lucida Bright, and Times have anywhere near a comprehen- sive symbol complement, and none of the symbol sets are from "major" font suppliers (too much work, not enough money; the demise of Mono- type's symbol catalogmetal onlyis a particularly great loss).

2.2 Package multicol

Metafont, Knuth's font creation program, is independent of TEX.
It generates only bitmap fonts (al- though internally it creates outlines on which the bitmaps are based). There is still research to be done on combining overlapping outlines into a single outline that can be used like the outline of a Type 1 font; Knuth has "frozen" Metafont, so further research and development are being done by someone else, and the result will not be called "Metafont". In fact, it's also possible to use Type 1 fonts with TEX, and nearly all TEX installations routinely use free or commercial Type 1 fonts, especially if they're not producing heavily technical material; only Computer Modern (the font Knuth developed), Lucida Bright, and Times have anywhere near a comprehen- sive symbol complement, and none of the symbol sets are from "major" font suppliers (too much work, not enough money; the demise of Mono- type's symbol catalogmetal onlyis a particularly great loss).

2.3 Package listings

```
\documentclass{article}
\usepackage{listings}
\begin{document}
\lstset{language=Pascal}
% Insert Pascal examples here. You can use another language,
then put its name instead of Pascal. For list of languages consult
the documentation.
\end{document}
```

2.4 Code examples

Pascal Example

```
for i:=maxint to 0 do
begin
{ do nothing }
end;
Write('Case insensitive ');
WritE('Pascal keywords.');
```

```
type
color = (red, yellow, blue);
hue = set of color;
vector = array [1..100] of integer
var
```

Python Code

Listing 1: Python example

```
import numpy as np

def incmatrix(genl1,genl2):
    m = len(genl1)
    n = len(genl2)
    M = None #to become the incidence matrix
    VT = np.zeros((n*m,1), int)  #dummy variable

    #compute the bitwise xor matrix
    M1 = bitxormatrix(genl1)
    M2 = np.triu(bitxormatrix(genl2),1)

    for i in range(m-1):
        for j in range(i+1, m):
            [r,c] = np.where(M2 == M1[i,j])
            for k in range(len(r)):
                VT[(i)*n + r[k]] = 1;
                VT[(i)*n + c[k]] = 1;
                VT[(j)*n + r[k]] = 1;
                VT[(j)*n + c[k]] = 1;

                if M is None:
                    M = np.copy(VT)
                else:
                    M = np.concatenate((M, VT), 1)

                VT = np.zeros((n*m,1), int)

    return M
```