1 Introduction

This document contains the following listings:

Listings

2 Inline Listings

Various delimiters: a_word, a_word, a_word, a_word and even a_word done.
Indirectly: a_word; and with messed up braces foo { bar .
And also as an environment: _word; done.

3 An untyped Listing

No options, language, etc

```
1 stuff1
2 stuff2
3 stuff3
```

4 Some C

```
1 #define EXAMPLE whichwhat
2 x = "foo";
3 break;
```

5 A Pascal Listing

A listing portion:

```
1 begin
2   { do nothing }
3 end;
```

A numbered listing:

```
1 for i:=maxint to 0 do
    begin
3     { do nothing }
    end;
5
  Write('case insensitive');
7 Write('long '' string');
  WritE('Pascal keywords.');
```

A Titled listing:

A bit of Pascal

```
1  for i:=maxint to 0 do
2  begin
3    { do nothing }
4  end;
5  Write('case insensitive');
```

A Captioned listing (known as Listing **??**) :

Listing 1: Another bit of Pascal

```
100  for i:=maxint to 0 do
101  begin
102    { do nothing }
103  end;
```

6 An Environment

```
1  for i:=maxint to 0 do
2  begin
3    { do nothing }
4  end;
```

```
for i:=maxint to 0 do                    1
begin                                    2
  { do nothing }                         3
end;                                     4
```

```
for i:=maxint to 0 do                    1
begin                                    2
  { do nothing }                         3
end;                                     4
```

7 Framing and such

```
1  for i:=maxint to 0 do
2  begin
3    { do nothing }
4  end;
```

```
1  for i:=maxint to 0 do
2  begin
3    { do nothing }
4  end;
```

```
for i:=maxint to 0 do
begin
  { do nothing }
end;
```

```
for i:=maxint to 0 do
begin
  { do nothing }
end;
```

Listing 2: A C language listing

```
#define EXAMPLE whichwhat
x = "foo";
break;
```

8 Listing with Math

```
// calculate a_ij
a[i][j] = a[j][j]/a[i][j];
```

```
// calculate a_ij
a[i][j] = a[j][j]/a[i][j];
```

```
// calculate a_ij
a_ij = a_jj/aij;
// calculate a_ij = sin x

a[i,j]=sin(x)
foo="a word";
foo="a x^2 math";
```

```
// calculate <a_ij>
a_{ij}
 = a_{jj}/a{ij};
```

```
// calculate $a_{ij}$
$a_{ij}
 = a_{jj}/a{ij}$;
// calculate $a_{ij} =
\sin x$
a[i,j]=sin(x)
foo="a word";
foo="a \"string";
foo="a $x^2$ math";
```

9 A Perl Listing

```
1  # -*- CPERL -*-
2  package LaTeXML::Package::Pool;
3  use strict;
4  use LaTeXML::Package;
5
6  DefConstructor('\container{}',"<ltx:special>#1</ltx:special>");
7  DefConstructor('\foo',"<ltx:not-defined/>");
8
9  1;
```

10 A Recursive TEX listing

```
1  \documentclass{article}
2  \usepackage{makeidx}
3  \makeindex
4  \usepackage{listings}
5  \usepackage[dvipsnames]{color}
6  \begin{document}
7  \lstset{numbers=left}
8  \section{Introduction}
9  This document contains the following listings:
10 \lstlistoflistings
11
12 \section{Inline Listings}
13 Various delimiters: \lstinline{a_word},
14 \lstinline!a_word!, \lstinline Aa_wordA,
15 \lstinline&a_word& and even \lstinline^a_word^ done.
16
17 \def\justcopy#1{#1}
18 Indirectly: \justcopy{\lstinline|a_word|};
19 and with messed up braces \lstinline{foo { bar }.% }
20
21 And also as an environment:
22 \begin{lstinline}
23 a_word
24 \end{lstinline}; done.
25 \section{An untyped Listing}
26 No options, language, etc
27 \begin{lstlisting}
28 stuff1
29 stuff2
30 stuff3
31 \end{lstlisting}
```

```
32
33 \section{Some C}
34
35 \begin{lstlisting}[language=C, identifierstyle=\slshape, directivestyle=\ttfamily]
36 #define EXAMPLE whichwhat
37 x = "foo";
38 break;
39 \end{lstlisting}
40
41 \section{A Pascal Listing}
42 A listing portion:
43 \begin{lstlisting}[language=Pascal, firstline=2, lastline=5, caption={}]
44 for i:=maxint to 0 do
45 begin
46   { do nothing }
47 end;
48
49 Write('case insensitive');
50 Write('long '' string');
51 WritE('Pascal keywords.');
52 \end{lstlisting}
53
54 A numbered listing:
55 \begin{lstlisting}[language=Pascal, numbers=left, numberstyle=\tiny, stepnumber=2
56 for i:=maxint to 0 do
57         begin
58                 { do nothing }
59         end;
60
61 Write('case insensitive');
62 Write('long '' string');
63 WritE('Pascal keywords.');
64 \end{lstlisting}
65
66 A Titled listing:
67 \begin{lstlisting}[language=Pascal, title={A bit of Pascal}]
68 for i:=maxint to 0 do
69 begin
70   { do nothing }
71 end;
72 Write('case insensitive');
73 \end{lstlisting}
74
75
76 A Captioned listing (known as Listing \ref{pascallisting}) :
77 \begin{lstlisting}[language=Pascal, caption=Another bit of Pascal, label=pascallis
```

```
78  for i:=maxint to 0 do
79  begin
80    { do nothing }
81  end;
82  \end{lstlisting}
83
84  \section{An Environment}
85  \begin{lstlisting}[language=Pascal]
86  for i:=maxint to 0 do
87  begin
88    { do nothing }
89  end;
90  \end{lstlisting}
91
92  \lstnewenvironment{colored}[1]{\lstset{language=Pascal,numbers=right,numberstyle
93  \begin{colored}{red}
94  for i:=maxint to 0 do
95  begin
96    { do nothing }
97  end;
98  \end{colored}
99
100 \begin{colored}{blue}
101 for i:=maxint to 0 do
102 begin
103   { do nothing }
104 end;
105 \end{colored}
106
107 \section{Framing and such}
108 \lstset{backgroundcolor=\color[named]{CarnationPink}}
109 \begin{lstlisting}[language=Pascal,frame=single,rulecolor=\color{red}]
110 for i:=maxint to 0 do
111 begin
112   { do nothing }
113 end;
114 \end{lstlisting}
115
116 \begin{lstlisting}[language=Pascal,frameround=tttt,backgroundcolor=\color{yellow
117 for i:=maxint to 0 do
118 begin
119   { do nothing }
120 end;
121 \end{lstlisting}
122 \lstset{backgroundcolor=}
123 \begin{lstlisting}[language=Pascal,frame=single]
```

```
124 for i:=maxint to 0 do
125 begin
126   { do nothing }
127 end;
128 \end{lstlisting}
129
130 \begin{lstlisting}[language=Pascal,frame=lines]
131 for i:=maxint to 0 do
132 begin
133   { do nothing }
134 end;
135 \end{lstlisting}
136
137 \begin{lstlisting}[language=C,identifierstyle=\slshape,directivestyle=\ttfamily,
138 caption=A C language listing, frame=lines,backgroundcolor={\color[cmyk]{0,0,0,0.
139 #define EXAMPLE whichwhat
140 x = "foo";
141 break;
142 \end{lstlisting}
143
144 \section{Listing with Math}
145 \begin{lstlisting}[language=c,texcl,commentstyle=\color{green}]
146 // \upshape calculate $a_{ij}$
147 a[i][j] = a[j][j]/a[i][j];
148 \end{lstlisting}
149
150 \begin{lstlisting}[texcl,language=c]
151 // \upshape calculate $a_{ij}$
152 a[i][j] = a[j][j]/a[i][j];
153 \end{lstlisting}
154
155 \begin{lstlisting}[language=c,mathescape,numbers=left,commentstyle=\color{green}
156 // calculate $a_{ij}$
157 $a_{ij}
158  = a_{jj}/a{ij}$;
159 // calculate $a_{ij} =
160 \sin x$
161 a[i,j]=sin(x)
162 foo="a word";
163 foo="a $x^2$ math";
164 \end{lstlisting}
165
166 \begin{lstlisting}[language=c,escapechar=\%,escapebegin=\textless,escapeend=\tex
167 // calculate %$a_{ij}$%
168 a_{ij}
169  = a_{jj}/a{ij};
```

```
\end{lstlisting}

\begin{lstlisting}[language=c,numbers=left,stringstyle=\ttfamily]
// calculate $a_{ij}$
$a_{ij}
 = a_{jj}/a{ij}$;
// calculate $a_{ij} =
\sin x$
a[i,j]=sin(x)
foo="a word";
foo="a \"string";
foo="a $x^2$ math";
\end{lstlisting}

\section{A Perl Listing}
\lstinputlisting[language=perl]{any.sty.ltxml}

\section{A Recursive \TeX\ listing}
\lstinputlisting[language={[LaTeX]TeX}]{listing.tex}

\section{Testing Tag}
% AHA, tagstyle only is in effect with XML (?)
\begin{lstlisting}[language=XML,tagstyle=\bf]
<element attr='value'>content</element>
\end{lstlisting}
\begin{lstlisting}[language=XML,tagstyle=\bf,usekeywordsintag=false]
<element attr='value'>content</element>
\end{lstlisting}
\begin{lstlisting}[language=XML,tagstyle=\bf,markfirstintag]
<element attr='value'>content</element>
\end{lstlisting}

\section{Screwiness}
\lstdefinelanguage{bingo}{morekeywords={foo,bar},morekeywords=[2]{bing,bar}}
%,
% AHA, words can only be in one class (1st one declared?)
% BUT, index is separate, and classname is without the "style" !!
\begin{lstlisting}[language=bingo,keywordstyle=\bfseries,keywordstyle={[2]\itsha
foo bar baz bing booboo
\end{lstlisting}
{\bfseries\itshape bfit}
{\itshape\bfseries itbf}
\printindex
\end{document}
```

11 Testing Tag

```
<element attr='value'>content</element>
```

```
<element attr='value'>content</element>
```

```
<element attr='value'>content</element>
```

12 Screwiness

```
foo bar baz bing booboo
bfit itbf
```