Fundamenta Informaticae XXI (2001) 1001–1008
IOS Press

How to Prepare Articles for Fundamenta Informaticae

First Author[*]
Institute of Informatics
Warsaw University
Banacha 2, 02-097 Warszawa, Poland
first@mimuw.edu.pl

Second Author[†]
Department of Informatics
City University
London, England

Abstract. This guide is for authors who are preparing papers for *Fundamenta Informaticae* using the LaTeX 2_ε document preparation system and the `fundam.cls` class file.

Keywords: LaTeX 2_ε, document class file, BibTeX

1. Introduction

The *Fundamenta Informaticae* class file is based on the standard LaTeX 2_ε `article` class as described in the LaTeX manual. It means that the paper prepared to be process with the standard `article` class can be processed with the `fundam` class without any modifications. Commands which differ from the standard LaTeX interface, or which are provided in addition to the standard interface, are explained in this guide. This guide is not a substitute for the LaTeX manual itself.

The `fundam` class can be used only with LaTeX 2_ε. It is not supposed to work with the obsolete 2.09 (or earlier) version of LaTeX.

Address for correspondence: Address for correspondence goes here
[*]Thanks for something to somebody
[†]Thanks for something else to somebody else

1.1. The `fundam` document class

The use of document styles allows a simple change of style (or style option) to transform the appearance of the document. The `fundam` class file preserves the standard LaTeX interface such that any document which can be produced using the standard LaTeX `article` class, can also be produced with the `fundam` style. However, the measure (or width of text) is different to that used in `article` therefore line breaks will change and long equations may need re-setting.

The authors are encouraged to minimize the manual line and page breaks in the manuscripts.

It is essential that you do not modify `fundam.cls`. When additional macros are required, place them in a separate file with the filename extension `.sty` (e.g. `mymacros.sty`) and load it as additional package file:

```
\usepackage{mymacros}
```

In this way you will distinguish clearly between the main style file and your own macros, which can then be found easily by the journal editor. Do not forget to submit your optional style file for publication along with your input file.

2. Using the `fundam` class

First, copy the file `fundam.cls` (and `fundam.bst` if you are using BibTeX) into the correct subdirectory on your system. In order to use the **fundam** class, place it as the argument of the `\documentclass` command at the beginning of your document. That is,

```
\documentclass{article}
```

is replaced by

```
\documentclass{fundam}
```

Do not use any document style options.

The class uses macros from standard LaTeX 2_ε packages: `theorem` and `amsmath`. These packages are included in every recent LaTeX 2_ε distribution.

2.1. Title, authors and other meta information

The title of the paper should be specified with the standard `\title` command. The names and affiliation and address information of the authors should be specified using `\author` command. The consecutive authors should be separated by `\and` command. The information for every author can contain more than one line. The lines should be separated by `\\` command. The first line is regarded as the name of the author, next lines should contain address/affiliation information.

It is possible to place the authors in two columns – it can be useful when the paper has three, four of more authors. Two column placement of author information is activated after using the `manyauthors` option in the invocation of class:

```
\documentclass[manyauthors]{fundam}
```

The acknowledgement information can be assigned to authors using standard `\thanks` command placed after the name of the author (inside `\author` command). If two authors have the same acknowledgement information, than the first author should be marked with `\thanks` command, the others should be marked with the `\thanksas{<number>}`. The examples of using these commands are shown below.

The corresponding author can be marked with the `\corresponding` command placed after his name. This command produce the footnote marked with the capital "C" as the superscript and the text: "Corresponding author" – it means that the paper should be sent to the marked author for corrections.

Optionally the correspondence address can be specified separately with `\address` command placed after `\maketitle` command. The example of the beginning of the paper written by the four authors is shown below.

```
\documentclass[manyauthors]{fundam}
\begin{document}
\title{The example paper}
\author{First Author\thanks{Thanks for something}\\
  Institute of Informatics \\
  University of Somewere
\and Second Author\thanks{For something else}\\
  Department of Informatics \\
  University of Some other Place
\and Third Author\thanksas{2}\\
  Department of Teleinformatics \\
  Nowhere University
\and Fourth Author\thanksas{1}\\
  Department of Applied Mathematics \\
  Somewere University of Technology}
\maketitle
\address{F. Author, Institute of Informatics, University of Somewhere}
```

Running head must be defined with the two argument command `\runninghead` – the first argument should contain the names of the authors, second one should contain shortened version of title. The issue number and year must be specified with help of the `\issue` command, and the starting page can be changed by setting the value of the counter page to the proper value as in the following example:

```
\setcounter{page}{1001}
\issue{XXI~(2001)}
\runninghead{F. Author, S. Author}{How to Prepare Articles for FI}
```

The issue and page number must be specified before the `\maketitle` command. The running head must be defined on the first page of the paper.

2.2. Sections

LaTeX provides five levels of section headings and although they are all defined in the `fundam` style file, only the first four should be used.

- Heading A – `\section`;
- Heading B – `\subsection`;
- Heading C – `\subsubsection`;
- Heading D – `\paragraph`;
- Heading E – `\subparagraph` should not be used.

2.3. lists

The `fundam` class provides the three standard list environments:

- Numbered lists, created using the `enumerate` environment.
- Unnumbered lists, created using the `itemize` environment.
- Labelled lists, created using the `description` environment.

The enumerated list numbers each list item with an arabic numeral; alternative styles can be achieved by inserting a redefinition of the number labeling command after the `\begin{enumerate}`. For example, a list numbered with roman numerals inside parentheses can be produced by the following commands:

```
\begin{enumerate}
 \renewcommand\labelenumi{\theenumi}
 \renewcommand{\theenumi}{(\roman{enumi})}
 \item first item
 \item second item
 \item etc\dots
\end{enumerate}
```

This produces the following list:

(i) first item

(ii) second item

(iii) etc...

2.4. Formulae and equations

Standard LaTeX environments (`equation` and `eqnarray`) for typesetting equations can be used in the articles. The numbering of equation is performed by LaTeX. These environments were used to typeset formulae (1), (2) and (3)

∗
$$a + b = c \tag{1}$$

(!)
$$\begin{aligned} a + b &= c & (2) \\ c - b &= a & (3) \end{aligned}$$

Because the `amsmath` package is included by the `fundam` class, the author can use the $\mathcal{A}\mathcal{M}\mathcal{S}$TEX macros for better handling of multi line formulae as shown in the examples (4)–(7). For details refer to the standard $\mathcal{A}\mathcal{M}\mathcal{S}$LATEX documentation.

$$(a) \qquad a + b = c \tag{4}$$
$$c - b = a \tag{5}$$

$$(*) \qquad a + b = c \tag{6}$$
$$(**) \qquad c - b = a \tag{7}$$

In addition to the standard numbering the equations can be marked with extra tags placed on the left margin. The `\ltag{}` command is provided for such a purpose. The mandatory argument of this command typeset in math mode is used as the tag. The equation (7) was for instance marked with command `\ltag{(**)}`.

2.5. Theorem like environments and proofs

The `fundam` class defines ready to use environments for typesetting typical theorem like environments: `definition`, `theorem`, `fact`, `lemma`, `example`, `assumption`, `proposition`, `remark`, `corollary`, `claim`. There is also defined the `proof` environment. An example of using this environment is given below (theorem 2.1):

```
\begin{theorem}[trivial]
There exist numbers $a$, $b$, and $c$, such that $2a+2b=2c$.
\end{theorem}
\begin{proof} We leave the proof for the reader.
\end{proof}
```

Theorem 2.1. (trivial)
There exists numbers a, b, and c, such that $2a + 2b = 2c$.

Proof:
We leave the proof for the reader. □

You can define more theorem like environments if you need them using the `\newtheorem` command.

2.6. Figures and Tables

The `figure` and `table` environments are inherited from the standard article class as described in the LATEX Manual to provide consecutively numbered floating inserts for illustrations and tables respectively.

Figure captions should be below the figure itself, therefore the `\caption` command should appear after the figure or space left for an illustration. For example, Figure 1 is produced using the following commands:

```
  \begin{figure}
    ...
    \caption{An example figure with space for artwork.}
    \label{sample-figure}
  \end{figure}
```

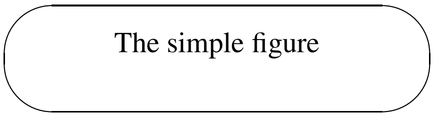

Figure 1. An example figure with space for artwork.

The preferred method of preparing figures is to save or convert them to the Encapsulated PostScript format (EPS) without headers. Such pictures can be included in consistent way and easily scaled. The `graphics` or `graphicx` package can be used to include graphics into the paper. Both of them provide the command `\includegraphics`, but the second one provides the easier way of resizing the picture using the optional, key-value parameters.

```
\includegraphics[width=0.5\textwidth]{file.eps}
```

The above example presets the use of `\includegraphics` command as provided by the `graphicx` package. The included picture will be resized to the half width of the typesetting area. If you not specify the height of the picture it will be scaled proportionally.

Table captions should be at the top therefore the `\caption` command should appear before the body of the table. For example, Table 1 is produced using the following commands:

```
\begin{table}[htbp]\small
  \caption{An example table}
  \label{sample-table}
  \begin{center}
  \begin{tabular}{|c|c|c|}
\hline
$m$ & $z_\kappa$ & $s_\kappa$\\\hline
10& 1.13641 & 1.27700  \\\hline
20& 1.06601 & 1.27801  \\\hline
  \end{tabular}
\end{center}
\end{table}
```

The `tabular` environment should be used to produce ruled tables. Commands to redefine quantities such as `\arraystretch` should be avoided.

Table 1. An example table

m	z_κ	s_κ
10	1.13641	1.27700
20	1.06601	1.27801

2.7. Bibliography

The best way to prepare bibliography for your paper is to use the BibTEX program. We assume that you know how to use the BibTEX program. If not, consult the LATEX manual. The special BibTEX style `fundam.bst` is provided with the LATEX class. Assuming that your BibTEX database file is named `citations.bib` you should add the following two lines to your manuscript file. The citations should be defined with `\cite` command. Here we cite the first and last bibliographical entries [1, 7].

```
\bibliographystyle{fundam}
\bibliography{citations}
```

The manual formating of your reference list is highly discouraged. We strongly recommend that you ask your system personnel to install BibTEX.

If you cannot use BibTEXfor some reason, you must prepare your bibliography manually, as the `thebibiliography` environment. For example, the bibliography of this guide can be produced manually as follows:

```
\begin{thebibliography}{1}
\bibitem{AlessiBarb} Alessi, F., Barbanera, F.: Strong conjunction and
  intersection types, \emph{Proc. Mathematical Foundations of Computer
    Science} (A.~Tarlecki, Ed.), LNCS 520, Springer-Verlag, Berlin,
  1991.

\bibitem{AB75} Anderson, A.~R., Belnap, N.~A.: \emph{Entailment. The
    Logic of Relevance and Necessity}, vol.~I, Princeton University
  Press, 1975.

\bibitem{arts} Arts, T.: \emph{Embedding first order predicate logic
    in second order propositional logic}, Master Thesis, Katholieke
  Universiteit Nijmegen, December 1992.

\bibitem{arts-dekkers} Arts, T., Dekkers, W.: \emph{Embedding first
    order predicate logic in second order propositional logic},
  Technical Report 93-02, Katholieke Universiteit Nijmegen, February
  1993.

\bibitem{barendregt} Barendregt, H.~P.: Lambda calculi with types, in:
  \emph{Handbook of Logic in Computer Science} (S.~Abramsky, D.~M.
  Gabbay, T.~Maibaum, Eds.), Oxford University Press, 1990, 117--309.
```

```
\bibitem{TCS} Dezani-Ciancaglini, M., de'Liguoro, U., Piperno, A.:
  Filter models for conjunctive-disjunctive $\lambda$-calculi,
  \emph{Theoretical Computer Science}, \textbf{179}(1--2), 1996,
  83--128.

\bibitem{wells} Wells, J.~B.: Typability and type checking in the
  second-order $\lambda$-calculus calculus are equivalent and
  undecidable, \emph{Proc. 9th IEEE Symposium on Logic in Computer
    Science}, Paris, France, 1994, (To appear in \emph{Annals of Pure
    and Applied Logic.}).
\end{thebibliography}
```

References

[1] Alessi, F., Barbanera, F.: Strong conjunction and intersection types, *Proc. Mathematical Foundations of Computer Science* (A. Tarlecki, Ed.), LNCS 520, Springer-Verlag, Berlin, 1991.

[2] Anderson, A. R., Belnap, N. A.: *Entailment. The Logic of Relevance and Necessity*, vol. I, Princeton University Press, 1975.

[3] Arts, T.: *Embedding first order predicate logic in second order propositional logic*, Master Thesis, Katholieke Universiteit Nijmegen, December 1992.

[4] Arts, T., Dekkers, W.: *Embedding first order predicate logic in second order propositional logic*, Technical Report 93-02, Katholieke Universiteit Nijmegen, February 1993.

[5] Barendregt, H. P.: Lambda calculi with types, in: *Handbook of Logic in Computer Science* (S. Abramsky, D. M. Gabbay, T. Maibaum, Eds.), Oxford University Press, 1990, 117–309.

[6] Dezani-Ciancaglini, M., de'Liguoro, U., Piperno, A.: Filter models for conjunctive-disjunctive λ-calculi, *Theoretical Computer Science*, **179**(1–2), 1996, 83–128.

[7] Wells, J. B.: Typability and type checking in the second-order λ-calculus calculus are equivalent and undecidable, *Proc. 9th IEEE Symposium on Logic in Computer Science*, Paris, France, 1994, (To appear in *Annals of Pure and Applied Logic.*).