

L^AT_EX for linguists

Sylvia Blaho & Patrik Bye



University of Tromsø

Spring 2005

1. Lists

- (1) Itemized lists:
- First item.
 - Second item, and some more text to show indentation.

```
\begin{itemize}  
\item First item.  
\item Second item...  
\end{itemize}
```

1. Lists

- (1) Itemized lists:
- First item.
 - Second item, and some more text to show indentation.

```
\begin{itemize}  
\item First item.  
\item Second item...  
\end{itemize}
```

- (2) Numbered lists:
1. First item.
 2. Second item, and some more text to show indentation.

```
\begin{enumerate}  
\item First item.  
\item Second item...  
\end{enumerate}
```

1. Lists

- (1) Itemized lists:
- First item.
 - Second item, and some more text to show indentation.

```
\begin{itemize}  
\item First item.  
\item Second item...  
\end{itemize}
```

- (2) Numbered lists:
1. First item.
 2. Second item, and some more text to show indentation.

```
\begin{enumerate}  
\item First item.  
\item Second item...  
\end{enumerate}
```

- (3) Descriptions:
- first** First item.
second Second item, and some more text to show indentation.

```
\begin{description}  
\item[first] First item.  
\item[second] Second item...  
\end{description}
```

(4) Nested lists:

- First level.
 - Second level.
 - * Third level.
 - Fourth level.
 - Fourth again.
 - * Third level.
 - Second level.
- First level.

```
\begin{itemize}
\item First level.
  \begin{itemize}
\item Second level.
  \begin{itemize}
\item Third level.
  \begin{itemize}
\item Fourth level.
\item Fourth again.
\end{itemize}
\end{itemize}
\item Third level.
\end{itemize}
\end{itemize}
\item Second level.
\end{itemize}
\item First level.
\end{itemize}
```

1. First level.
 - (a) Second level.
 - i. Third level.
 - A. Fourth level.
 - B. Fourth again.
 - ii. Third level.
 - (b) Second level.
2. First level.

```
\begin{enumerate}
\item First level.
  \begin{enumerate}
    \item Second level.
      \begin{enumerate}
        \item Third level.
          \begin{enumerate}
            \item Fourth level.
            \item Fourth again.
          \end{enumerate}
        \end{enumerate}
      \end{enumerate}
    \item Third level.
  \end{enumerate}
  \item Second level.
\end{enumerate}
\item First level.
\end{enumerate}
```

(5) Manual labels

- α TIPA.
- ♥ Heart.

```
\begin{itemize}  
  \item[\textipa{A}] \tipa.  
  \item[$\heartsuit$] Heart.  
\end{itemize}
```

(5) Manual labels

♠ TIP A.

♥ Heart.

```
\begin{itemize}
  \item[\texttip{A}] \tipa.
  \item[$\heartsuit$] Heart.
\end{itemize}
```

(6) Redefining labels

```
\renewcommand{\labelitemiii}{+}
```

(labelitemi, labelitemii, labelitemiii, labelitemiv)

(5) Manual labels

α TIP A.

♥ Heart.

```
\begin{itemize}
  \item[\texttip{A}] \tipa.
  \item[$\heartsuit$] Heart.
\end{itemize}
```

(6) Redefining labels

```
\renewcommand{\labelitemiii}{+}
```

(labelitemi, labelitemii, labelitemiii, labelitemiv)

```
\renewcommand{\labelenumi}{\roman{enumi}.}
```

(labelenumi, labelenumii, labelenumiii, labelenumiv)

```
(\arabic, \roman, \Roman, \alph, \Alph)
```

(5) Manual labels

♠ TIP A.

♥ Heart.

```
\begin{itemize}
  \item[\texttip{A}] \tipa.
  \item[$\heartsuit$] Heart.
\end{itemize}
```

(6) Redefining labels

```
\renewcommand{\labelitemiii}{+}
```

(labelitemi, labelitemii, labelitemiii, labelitemiv)

```
\renewcommand{\labelenumi}{\roman{enumi}.}
```

(labelenumi, labelenumii, labelenumiii, labelenumiv)

```
(\arabic, \roman, \Roman, \alph, \Alph)
```

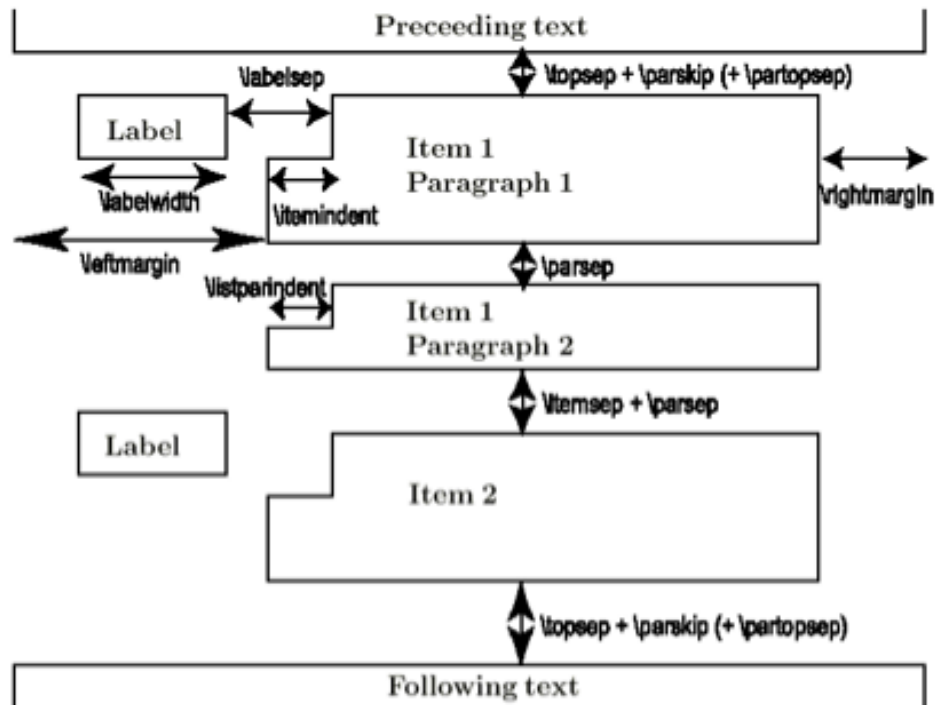
(7) Generalised lists

```
\begin{list}
```

```
{label declarations}
```

```
{list declarations}
```

```
\end{list}
```



1.1. Lengths

- Fixed lengths

cm centimetre

mm millimetre

in inch (1 in = 2.54 cm)

pt point (1 in = 72.27 pt)

bp big point (1 in = 72 bp)

pc pica (1 pc = 12 pt)

1.1. Lengths

- Fixed lengths

`cm` centimetre

`mm` millimetre

`in` inch (1 in = 2.54 cm)

`pt` point (1 in = 72.27 pt)

`bp` big point (1 in = 72 bp)

`pc` pica (1 pc = 12 pt)

- Font-specific lengths

`em` the width of the capital M

`ex` the height of the letter x

- Commands for changing lengths

`\setlength{\mylength}{length_spec}`

Assigns the value of `length_spec` to `\mylength`. Note that `length_spec` can be a length specification or another length command.

- Commands for changing lengths

`\setlength{\mylength}{length_spec}`

Assigns the value of `length_spec` to `\mylength`. Note that `length_spec` can be a length specification or another length command.

`\addtolength{\mylength}{length_spec}`

Adds `length_spec` to the current value of `\mylength`. If `length_spec` has a negative value, `\mylength` decreases.

- Commands for changing lengths

`\setlength{\mylength}{length_spec}`

Assigns the value of `length_spec` to `\mylength`. Note that `length_spec` can be a length specification or another length command.

`\addtolength{\mylength}{length_spec}`

Adds `length_spec` to the current value of `\mylength`. If `length_spec` has a negative value, `\mylength` decreases.

Note that `length_spec` can be a length specification or another length command.

- Commands for changing lengths

`\setlength{\mylength}{length_spec}`

Assigns the value of `length_spec` to `\mylength`. Note that `length_spec` can be a length specification or another length command.

`\addtolength{\mylength}{length_spec}`

Adds `length_spec` to the current value of `\mylength`. If `length_spec` has a negative value, `\mylength` decreases.

Note that `length_spec` can be a length specification or another length command.

`\settowidth{\mylength}{text}`

Sets `\mylength` to the natural length of the specified text.

- Commands for changing lengths

`\setlength{\mylength}{length_spec}`

Assigns the value of `length_spec` to `\mylength`. Note that `length_spec` can be a length specification or another length command.

`\addtolength{\mylength}{length_spec}`

Adds `length_spec` to the current value of `\mylength`. If `length_spec` has a negative value, `\mylength` decreases.

Note that `length_spec` can be a length specification or another length command.

`\settoheight{\mylength}{text}`

Sets `\mylength` to the natural length of the specified text.

`\settotoheight{\mylength}{text}`

Sets `\mylength` to the natural height of the specified text.

- Commands for changing lengths

`\setlength{\mylength}{length_spec}`

Assigns the value of `length_spec` to `\mylength`. Note that `length_spec` can be a length specification or another length command.

`\addtolength{\mylength}{length_spec}`

Adds `length_spec` to the current value of `\mylength`. If `length_spec` has a negative value, `\mylength` decreases.

Note that `length_spec` can be a length specification or another length command.

`\settowidth{\mylength}{text}`

Sets `\mylength` to the natural length of the specified text.

`\settoheight{\mylength}{text}`

Sets `\mylength` to the natural height of the specified text.

`\settodepth{\mylength}{text}`

Sets `\mylength` to the natural depth of the specified text.

- Commands for changing lengths

`\setlength{\mylength}{length_spec}`

Assigns the value of `length_spec` to `\mylength`. Note that `length_spec` can be a length specification or another length command.

`\addtolength{\mylength}{length_spec}`

Adds `length_spec` to the current value of `\mylength`. If `length_spec` has a negative value, `\mylength` decreases.

Note that `length_spec` can be a length specification or another length command.

`\settowidth{\mylength}{text}`

Sets `\mylength` to the natural length of the specified text.

`\settoheight{\mylength}{text}`

Sets `\mylength` to the natural height of the specified text.

`\settodepth{\mylength}{text}`

Sets `\mylength` to the natural depth of the specified text.

- You can define a new length by `\newlength{\lengthname}`. The default value of `\newlength` is 0 pt.

This is text preceding the list.

- ♥ Some text
- ♥ More text, and yet
some more text to see
indentation.

And this is text following the list.

This is text preceding the list.

```
\begin{list}{\heartsuit$}
{\setlength{\rightmargin}{\leftmargin}
\setlength{\itemsep}{-0.3cm}}
\item Some text.
\item More text...
\end{list}
```

And this is text following the list.

FIGURE I: *First figure.*
FIGURE II: *Second figure.*
FIGURE III: *Third figure. And now some
more text to show sizes and in-
dentation.*
*And here is another ¶ in Figure
III.*

```
\newcounter{myfigure}  
\begin{list}{\textsc{Figure}}  
~\upshape{\Roman{myfigure}}:\hfill}  
{\usecounter{myfigure}}  
\itshape  
\setlength{\labelwidth}{2cm}  
\setlength{\labelsep}{0.5cm}}  
\item First figure.  
\item Second figure.  
\item Third figure. And now...  
\end{list}
```

The 1950's: Sex.

The 1960's: Drugs.

The 1970's: Rock'n'roll.

```
\newcounter{decade}
\newcounter{year}
\begin{list}
{\setcounter{year}{\value{decade}*10}%
The \arabic{year}'s:\hfill\stepcounter{decade}}%
{\usecounter{year}}
\setcounter{decade}{195}
\item Sex.
\item Drugs.
\item Rock'n'roll.
\end{list}
```

You need the package `calc` for this (part of the standard distribution).

2. Linguistic examples

- (8) Probably the most frequently used packages are `linguex` and `gb4e`. We strongly recommend `linguex`, because it takes considerably less typing and is much easier to customise than `gb4e`. Warning: if you try to load both `gb4e` and `linguex` in the same document, you're most likely to get into trouble!

Note, however, that `linguex` uses parts of `gb4e`, so it will only work if you obtain `cgloss4e` and `xspace.sty` along with `linguex.sty`.

- (1) Here is an example.
 - a. This is the second level of examples.
 - b. Still the second level.
 - (i) And the third level.
 - (ii) But don't go deeper than that.

Because you run out of labels.

```
\ex. Here is an example.  
\a. This is the second level of examples.  
\b. Still the second level.  
\a. And the third level.  
\b. But don't go deeper than that.  
    \a. Because you run out of labels.
```

The `\ex.` environment must be closed by a single blank line – this closes all embedded `\a.` environments as well.

(9) To close only one level of embedded lists:

(2) Here is an example.

- a. This is the second level of examples.
- b. Still the second level.
 - (i) And the third level.
 - (ii) But don't go deeper than that.
- c. And back to level 2.

`\ex. Here is an example.`

`\a. This is the second level of examples.`

`\b. Still the second level.`

`\a. And the third level.`

`\b. But don't go deeper than that.`

`\z.`

`\b. And back to level 2.`

(10) Glosses are automatically aligned with the example sentence

(3) Here is an example.

a. This is the second level of examples.

b. Az első mondat.
the first sentence
'The first sentence.'

`\ex. Here is an example.`

`\a. This is the second level of examples.`

`\bg. Az els\H{o} mondat.\\`

`the first sentence\\`

`'The first sentence.'`

(10) Glosses are automatically aligned with the example sentence

(3) Here is an example.

a. This is the second level of examples.

b. Az első mondat.
the first sentence
'The first sentence.'

`\ex. Here is an example.`

`\a. This is the second level of examples.`

`\bg. Az első mondat.\`

`the first sentence\`

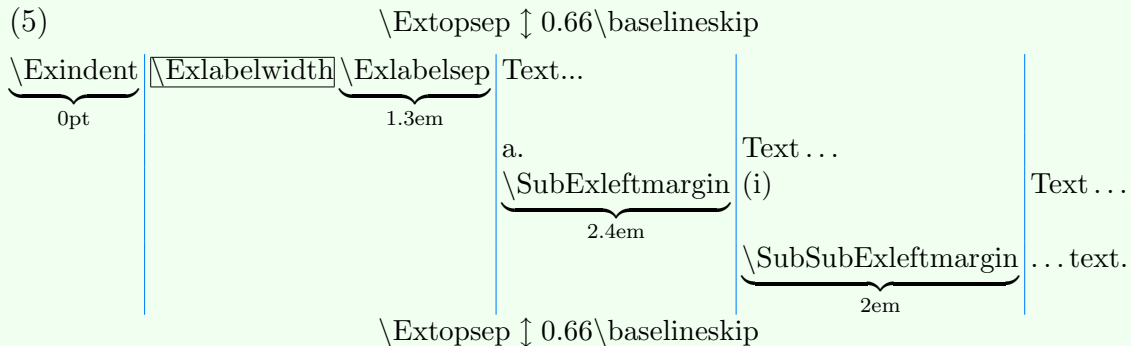
`'The first sentence.'`

(11) Symbols for grammaticality judgements

(4) *?Az mondat első.
the sentence first

`\exg. *? Az mondat első.\`
`the sentence first\`

(12) Customising lengths and margins



Note that the commands changing the defaults for `linguex` must *follow* `\begin{document}`.

(13) Customising labels

The counters used are **ExNo**, **SubExNo** and **SubSubExNo**, respectively, and treated like any other counter.

(13) Customising labels

The counters used are `ExNo`, `SubExNo` and `SubSubExNo`, respectively, and treated like any other counter.

`\a.` Can be used without being embedded in `\ex.`, with the expected result:

- | | | |
|----|--------------|-------------------------------|
| a. | Example one. | <code>\a. Example one.</code> |
| b. | Example two. | <code>\b. Example two.</code> |

The list commands also take optional arguments:

(13) Customising labels

The counters used are `ExNo`, `SubExNo` and `SubSubExNo`, respectively, and treated like any other counter.

`\a.` Can be used without being embedded in `\ex.`, with the expected result:

a.	Example one.	<code>\a. Example one.</code>
b.	Example two.	<code>\b. Example two.</code>

The list commands also take optional arguments:

one	Example one.	<code>\a.[one] Example one.</code>
two	Example two.	<code>\b.[two] Example two.</code>

There should be no space between the command-final period and the optional argument!

For other things these packages can do, see their documentation.

Customising page layout

- Headers and footers with `fancyhdr`
- Margins with `geometry`
- Section titles with `titlesec`

3. Customising headers and footers

(14) Package: `fancyhdr` (part of most T_EX distributions).

```
\usepackage{fancyhdr}
```

```
\pagestyle{fancy}
```

```
\fancyhead[L]{Your Name}
```

```
\fancyfoot[L]{\roman{page}}
```

3. Customising headers and footers

- (14) Package: `fancyhdr` (part of most $\text{T}_\text{E}\text{X}$ distributions).

```
\usepackage{fancyhdr}
\pagestyle{fancy}

\fancyhead[L]{Your Name}
\fancyfoot[L]{\roman{page}}
```

- (15) To cancel previous header/footer settings:

```
\fancyhead{}
\fancyhead[L]{Your Name}
\fancyhead[R]{The title of the paper}
\fancyfoot{}
\fancyfoot[L]{\roman{page}}
```

(16) with the option `twoside`

```
\documentclass[twoside]{article}
\usepackage{fancyhdr}
\pagestyle{fancy}

\fancyhead{}
\fancyhead[OL]{Your Name}
\fancyhead[OR]{The title of the paper}
\fancyhead[EL]{\today}
\fancyhead[ER]{Troms\o{}}
\fancyfoot{}
\fancyfoot[OL]{\roman{page}}
\fancyfoot[ER]{\roman{page}}
```

(16) with the option `twoside`

```
\documentclass[twoside]{article}
\usepackage{fancyhdr}
\pagestyle{fancy}

\fancyhead{}
\fancyhead[OL]{Your Name}
\fancyhead[OR]{The title of the paper}
\fancyhead[EL]{\today}
\fancyhead[ER]{Troms\o{}}
\fancyfoot{}
\fancyfoot[OL]{\roman{page}}
\fancyfoot[ER]{\roman{page}}
```

(17) Changing the rule in headers/footers:

```
\renewcommand{\headrulewidth}{0pt}
\renewcommand{\footrulewidth}{0.1ex}
```

(18) Different headers and footers on even and odd-numbered pages

```
\documentclass[twoside]{article}
\usepackage{fancyhdr}
\pagestyle{fancy}

\fancyhead{}
\fancyhead[OL]{Your Name}
\fancyhead[OR]{The title of the paper}
\fancyhead[EL]{\today}
\fancyhead[ER]{Troms\o{}}
\fancyfoot{}
\fancyfoot[OL]{\roman{page}}
\fancyfoot[ER]{\roman{page}}
```

(18) Different headers and footers on even and odd-numbered pages

```
\documentclass[twoside]{article}
\usepackage{fancyhdr}
\pagestyle{fancy}

\fancyhead{}
\fancyhead[OL]{Your Name}
\fancyhead[OR]{The title of the paper}
\fancyhead[EL]{\today}
\fancyhead[ER]{Troms\o{}}
\fancyfoot{}
\fancyfoot[OL]{\roman{page}}
\fancyfoot[ER]{\roman{page}}
```

(19) Adding vertical rules to headers and footers

```
\fancyfoot{}
\fancyfoot[OL,ER]{\hrulefill\\ This is page \roman{page}%
\\ \vspace{-1ex}\hrulefill}
\fancyfoot[OR,EL]{\hrulefill\\ \LaTeX{} rocks!\\ \vspace{-1ex}\hrulefill}
\renewcommand{\headrulewidth}{0pt}
\renewcommand{\footrulewidth}{0pt}
```

(20) Alternative ways to define text in headers and footers with fancyhdr:

```
\lhead{Left head}   \chead{Centre head}   \rhead{Right head}  
\lfoot{Left foot}   \cfoot{Centre foot}   \rfoot{Right foot}
```

(20) Alternative ways to define text in headers and footers with `fancyhdr`:

```
\lhead{Left head}   \chead{Centre head}   \rhead{Right head}  
\lfoot{Left foot}   \cfoot{Centre foot}   \rfoot{Right foot}
```

(21) If you define the headers and footers in the preamble, the definitions will have scope over the whole document. If you change them in the body of the document, the changes will take effect from the next page onwards.

4. Changing margins

(22) Package: geometry

Make sure you change the margins *before* defining the header and footer!

So `\geometry{options}` must precede `\pagestyle{fancy}`, and the rest of the fancy commands. If it doesn't, the header and footer will have different margins than the rest of the text.

4. Changing margins

(22) Package: geometry

Make sure you change the margins *before* defining the header and footer!

So `\geometry{options}` must precede `\pagestyle{fancy}`, and the rest of the fancy commands. If it doesn't, the header and footer will have different margins than the rest of the text.

(23) Changing the margins

```
\usepackage{geometry}  
\geometry{a4paper,top=3cm,bottom=2cm,left=2cm,right=2cm}
```

4. Changing margins

(22) Package: geometry

Make sure you change the margins *before* defining the header and footer!

So `\geometry{options}` must precede `\pagestyle{fancy}`, and the rest of the fancy commands. If it doesn't, the header and footer will have different margins than the rest of the text.

(23) Changing the margins

```
\usepackage{geometry}  
\geometry{a4paper,top=3cm,bottom=2cm,left=2cm,right=2cm}
```

(24) alternatively:

```
\usepackage[a4paper,top=3cm,bottom=2cm,left=2cm,right=2cm]{geometry}
```

(25) If all margins are the same:

```
\usepackage[a4paper,margin=1in]{geometry}
```

- (26) With the options `includehead` and `includefoot`, the header/footer are included in the total body of the text. So, with `\geometry{a4paper,1in}`, there's 1 inch between the top of the page and the top of the *text body*, but with `\geometry{a4paper,includehead,margin,1in}`, the specified distance is between the top of the page and the top of the *header*.
- (27) For many more things `geometry` can do, see the documentation.

5. Redefining section titles

(28) Package: titlesec

```
\titleformat{command}[shape]{format}{label}{sep}{before}
```

command is the sectioning command to be redefined, e.g. `\section`.

shape sets the paragraph shape. See the package documentation for possible values.

format is the set of commands that are applied to the whole title (the label and the text as well).

label defines the label of the title.

sep is the horizontal space between the label and the text in the title. It must be defined as a length.

before is the code preceding the title body.

5. Redefining section titles

(28) Package: titlesec

```
\titleformat{command}[shape]{format}{label}{sep}{before}
```

command is the sectioning command to be redefined, e.g. `\section`.

shape sets the paragraph shape. See the package documentation for possible values.

format is the set of commands that are applied to the whole title (the label and the text as well).

label defines the label of the title.

sep is the horizontal space between the label and the text in the title. It must be defined as a length.

before is the code preceding the title body.

(29) An example

```
\usepackage{titlesec}  
\titleformat{\section}{\itshape}{\thesection.}{1em}{\bfseries}
```

(30) Changing the title spacing

```
\titlespacing{command}{left}{beforesep}{aftersep}
```

command is the sectioning command to be redefined, e.g. `\section`.

left is horizontal space before the title. If it has a negative value, the section title is shifted to the left.

beforesep is the vertical space before the title.

aftersep is space following the title. The default is vertical space, but it can also be horizontal space depending on the **shape** selected.

(30) Changing the title spacing

```
\titlespacing{command}{left}{beforesep}{aftersep}
```

command is the sectioning command to be redefined, e.g. `\section`.

left is horizontal space before the title. If it has a negative value, the section title is shifted to the left.

beforesep is the vertical space before the title.

aftersep is space following the title. The default is vertical space, but it can also be horizontal space depending on the **shape** selected.

```
\usepackage{titlesec}  
\titlespacing{\section}{3em}{1em}{1em}
```

(30) Changing the title spacing

```
\titlespacing{command}{left}{beforesep}{aftersep}
```

command is the sectioning command to be redefined, e.g. `\section`.

left is horizontal space before the title. If it has a negative value, the section title is shifted to the left.

beforesep is the vertical space before the title.

aftersep is space following the title. The default is vertical space, but it can also be horizontal space depending on the **shape** selected.

```
\usepackage{titlesec}  
\titlespacing{\section}{3em}{1em}{1em}
```

Using the starred version of `titlespacing` makes sure the paragraph following the title is not indented.

```
\usepackage{titlesec}  
\titlespacing*{\section}{3em}{1em}{1em}
```

(31) Fixed width labels

This might be necessary if a journal's style sheet requires paragraph indentation to be the same as the indentation of the title text.

```
\titleformat{\section}{\itshape }{\makebox[3em]{\thesection.}}{0em}{\bfseries}  
\titlespacing{\section}{0em}{1em}{1em}  
\setlength{\parindent}{3em}
```

(32) This package can do much, much more – check out the documentation if you're interested.

Homework

Take your sample article (HW 1) and modify it in the following ways.

1. Headers and footers:
 - (a) Odd page header: your name on the right in italics, the page number in capital Roman numbers on the left, slanted
 - (b) Even page header: the title of your article on the left in bold italics, the page number in capital Roman numbers on the right, slanted
 - (c) Odd page footer: the name of the journal on the left
 - (d) Even page footer: the starting and ending page numbers of the article on the right (automatically, not entered manually). The article should start on page 24.
 - (e) The header and the footer are separated from the text body by a 1pt wide rule.
2. All margins should be 2.5 cm. The header and footer should be included in the total body of the text (there should be 2.5 cm between the top of the header and the top of the page, as well as the bottom of the footer and the bottom of the page).
3. Section titles

- (a) Section titles should be slanted and bold, section numbers should be slanted but not bold. Section numbers should be followed by period. The same applies to subsection titles.
- (b) There should be one empty line between (sub)section titles and preceding text, and no lines between (sub)section titles and the following text.