Linguistic examples and trees

Christine M. Nilsen, University of Oslo

c.m.nilsen@ilos.uio.no

February 7, 2020

1. Linguistic input and special characters
2. The Babel package
3. The Linguex package
   - Examples and glossing
4. The Xyling package
   - Drawing trees

Special characters

- Most text editors have the 26 English letters as a default setting.
- However, we want to write examples in Greek, Icelandic, Slavic, Russian, French, Norwegian, IPA ...
- Two important factors to facilitate non-English characters: text encoding and packages

Text encoding

- Characters are stored as codes in the system.
- By using font encoding, you will need to type codes to generate diachronic signs and special characters. \"{a} $\longrightarrow$ ä
- By using UTF-8 encoding, the programme decodes the input in the compilation so that the output equals the input: ä is typed this way in the input.
- In order to use non-English characters, add the following to the preamble:

```
\usepackage[T1]{fontenc}
\usepackage[utf8]{inputenc}
```

The Babel package

- The babel package provides the necessary data to write text in other languages than English, both with respect to special characters and hyphenation.
- Add the following to the preamble:

```
\usepackage{babel}
```

- Specify the language between brackets within the command and the package:

```
\usepackage[francais]{babel}
```

- You can specify an illlimited number of languages separated by commas. The last language will be the main language of the document.

```
\usepackage[francais, icelandic, english]{babel}
```

Exercice 1 – change the language

Open a new file and type the following:

```
\documentclass{article}
\usepackage[russian, greek, english, german, latin, francais]{babel}
\usepackage{blindtext}

\begin{document}

\Blindtext
\end{document}
```

1. Compile the document.
2. Change the order of the languages and compile again. Repeat.

The Babel package – Cyrillic and Greek

- Non-latin alphabets can easily be loaded with the babel package
- Add the language between the brackets

```
\usepackage[russian, greek, english]{babel}
```

- For sections of text in other alphabets in the main text, use the following:

```
\selectlanguage{language}{text}
```

The Babel package – Cyrillic and Greek

- Non-latin alphabets can easily be loaded with the babel package
- Add the language between the brackets

```
\usepackage[russian, greek, english]{babel}
```

- For sections of text in other alphabets in the main text, use the following:

```
\selectlanguage{language}{text}
```

`input:` `\selectlanguage{russian}{text}`

`output:` Я помню чудное мгновенье:
Передо мной явилась ты

`input:` `\selectlanguage{greek}{text}`

`output:` Ειμαι Χριστινα

Exercice 2 – insert Greek and Cyrillic text

Open a new file and type the following:

```
\documentclass{article}
\usepackage[russian, greek, francais, german, latin, english]{babel}

\begin{document}

This is an example of how to insert Cyrillic and Greek letters into a document in English. First some Russian text \selectlanguage{russian}{whatever Cyrillic letters you find}, then some Greek \selectlanguage{greek}{something in Greek}.
\end{document}
```

Compile the document.

The Linguex package

- The Linguex package was 'designed for the lazy linguist'.
- Facilitates numbering of examples, glossing, and bracketing.
- The package is specified in the preamble.

```
\usepackage{linguex}
```

Examples and glossing

- The code \ex. provides numbering of examples.
- Numbering will always be updated, even though preceding examples are deleted, or if new are added.

Examples and glossing

- The code \ex. provides numbering of examples.
- Numbering will always be updated, even though preceding examples are deleted, or if new are added.

`input`:

\ex. This is an example.

\ex. This is another example

`output`:

(1) This is an example.

(2) This is another example

Examples

- After the command \ex., it is possible to add a), b), c) etc.
- While the numbers of examples is always updated, a), b), c) must be handled manually (NB!)

Examples

- After the command \ex., it is possible to add a), b), c) etc.
- While the numbers of examples is always updated, a), b), c) must be handled manually (NB!)

`input:`

```
\ex. This is an example.
\a. This is type a.
\b. This is type b.
```

`output:`

(1) This is an example.

a. This is type a.

b. This is type b.

Examples

- Level a) may occur at the same line as the example numbering provided no text is entered here

Examples

- Level a) may occur at the same line as the example numbering provided no text is entered here

input:

```
\ex.
\a. This is type a.
\b. This is type b.
```

output:

(1) a. This is type a.
b. This is type b.

Glossing

- Linguistic examples should be glossed where the gloss appears immediately below the word.
- The idiomatic translation should be on a third line.
- This is easily done with the command \exg.
- There is no need to align words – LaTeXdoes that for you.
- At the end of every line, add \ \ (code for line break)

Glossing

- Linguistic examples should be glossed where the gloss appears immediately below the word.
- The idiomatic translation should be on a third line.
- This is easily done with the command \exg.
- There is no need to align words – LaTeXdoes that for you.
- At the end of every line, add \ \ (code for line break)

`input:`

```
\exg. Ronja er en snill hest.\\
Ronja is a kind horse \\
'Ronja is a kind horse.'
```

`output:`

(1) *Ronja er en snill hest.*
Ronja is a kind horse
'Ronja is a kind horse.'

Exercice 3 – Add examples to your text

\documentclass{article}

\usepackage[english]{babel} (add any other language you want)

\usepackage{linguex}

\begin{document}

It is always important to add examples to a text.

\ex. This is my first example.

\a. This is my level a. example.

It is equally important to gloss examples.

\exg. Du er en trollmann, Harry, sa Gygrid. \ \

you are a wizard Harry said Hagrid \ \

'You're a wizard Harry, said Hagrid.'

\end{document}

Exercice 4 – Add italics

The file you just compiled looks good – but you may want to put all the quotes in italics. This can easily be done, adding a command to your body immediately after the command \begin{document}.

```
\let\eachwordone\itshape
```

Compile the document

Bracketing

- It is useful to use brackets in linguistic examples.
- The command \exi. sets the initial word after a bracket as subscript
- NB! There must be space before the bracket.
- NB! There must be space before and after the line with the command.

Bracketing

- It is useful to use brackets in linguistic examples.
- The command \exi. sets the initial word after a bracket as subscript
- NB! There must be space before the bracket.
- NB! There must be space before and after the line with the command.

input: \exi. [DP Ronja] [V is] [DP a kind horse].

output:

(1) $[_{\mathrm{DP}}$ Ronja] $[_{\mathrm{V}}$ is] $[_{\mathrm{DP}}$ a kind horse].

Cross-references

- Cross-references may be used at all levels: to examples, sections, and pages.
- \label{label1} creates a bookmark.
- The command \ref{label1} produces a cross-reference to the point where label1 has been placed.
- The command \pageref{label1} produces a cross-reference to the page where label1 has been placed.
- The file must be compiled twice to generate cross-references.

Exercice 5 – Add brackets and cross-references

Use the file you just compiled.

1. After your first example, add \label{ex}
2. Add the following lines after the text you have already entered.

```
\exi. [DP Harry] [V is] [DP a wizard]
```

3. Compile the document.
4. Below, write the following:

```
As we saw in example \ref{ex}, this is an example.
```

5. Compile the document twice.

The Xyling package

- The Xyling package makes it possible to draw trees, indicate movement with arrows, and a lot more.
- The following code must be added to the preamble:

```
\usepackage{xyling}
```

Drawing trees

- Drawing trees is based on points (or columns) in a grid.
- The tree is declared in the following way:

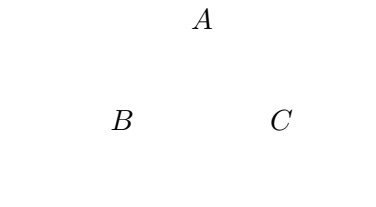
```
\Tree{}
```

- Columns are separated by &.
- Rows are separated by \\

Drawing trees

```
input: \Tree{
&&A \ \
&B && C \ \
D&&E&&F }
```

output:



Exercice 6 – Draw a grid

Open a new file. Write the following code. Compile.

```
\documentclass{article}
\usepackage{xyling}
\begin{document}

\Tree{
&&A \ \
&B && C \ \
D&&E&&F }

\end{document}
```

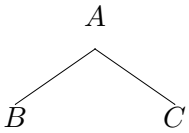
Drawing trees

- In order to draw a tree, we need branches.
- The branches are declared in the following way:

```
\B{dl}
\B{d}
\B{dr}
```

- B = branch; d = down; r = right; l = left
- These may be accumulated: dd = down two steps; rrr = three steps to the right.
- NB! For branches to be shown, typesetting must be set to produce DVI files.

Drawing trees



Exercice 7 – Add branches

Use the file you just created. Add branches. Compile.

```
\B{dl}
\B{d}
\B{dr}
```

Problems? Check the following:

- Do you compile using Tex/DVI?
- No branch can reach an empty point in the grid (or an empty line)

Generative trees

- Xyling featurs built-in macros for drawing generative trees
- The following codes will generate a node or a node (+ branches) (works for CP, TP, VP, and NP)

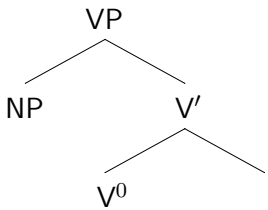
```
\VP
\Vbar
\Vzero
```

Drawing trees

input:

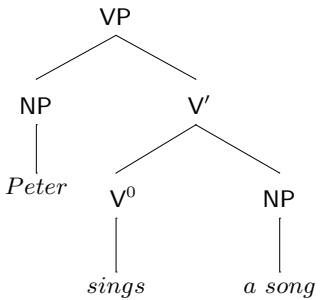
```
\Tree{ && \VP\\
& \NP && \Vbar\\
&& \Vzero && \\
}
```

output:



Exercice 8 – Draw this tree

Use the preamble from the previous tree. Use the macros described above. Draw the tree below.



Exercice 8 – code

```
\Tree{
&& \VP
& \NP \B{d} &&  \Vbar \ \
& Peter& \Vzero \B{d}&& \NP \B{d}
& \ \
&&sings && a \ song}
```

Notes

- The Xyling package offers an array of possibilities regarding treedrawing, arrows, highlighting etc. We refer to the package documentation for more info on the features of this packages.
- Also the Linguex package has more features than shown in this presentation. Here too, the package documentation is a useful document.