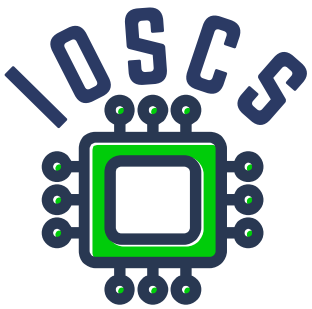
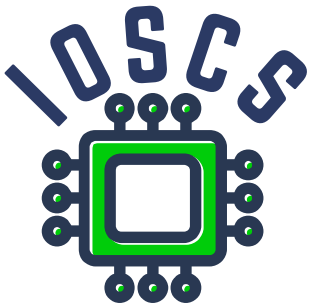


Project: Innovative Open Source Courses for Computer Science

Open source tools for text processing
Material for Laboratories

Jiří Rybička
Mendel University in Brno

29. 5. 2021



This teaching material was written as one of the outputs of the project "Innovative Open Source Courses for Computer Science", funded by the Erasmus+ grant no. 2019-1-PL01-KA203-065564. The project is coordinated by West Pomeranian University of Technology in Szczecin (Poland) and is implemented in partnership with Mendel University in Brno (Czech Republic) and University of Žilina (Slovak Republic). The project implementation timeline is September 2019 to December 2022.

Project information

Project was implemented under the Erasmus+.
Project name: **"Innovative Open Source courses for Computer Science curriculum"**
Project nr: **2019-1-PL01-KA203-065564**
Key Action: **KA2 – Cooperation for innovation and the exchange of good practices**
Action Type: **KA203 – Strategic Partnerships for higher education**

Consortium
ZACHODNIOPOMORSKI UNIWERSYTET TECHNOLOGICZNY W SZCZECINIE
MENDELOVA UNIVERZITA V BRNĚ
ŽILINSKÁ UNIVERZITA V ŽILINE

Co-funded by the
Erasmus+ Programme
of the European Union

Laboratories

Open source tools for text processing

Jiří Rybička
Department of Informatics
FBE MENDELU
rybicka@mendelu.cz

Project: Innovative Open Source Courses for Computer Science

Funded by
the European Union

Contents

- TEX system – basic principles

Contents

- TEX system – basic principles
- Fonts, font sources, macros, definition and parameters

Contents

- TEX system – basic principles
- Fonts, font sources, macros, definition and parameters
- UTF-8 encoding and special characters; lengths and length registers

Contents

- TEX system – basic principles
- Fonts, font sources, macros, definition and parameters
- UTF-8 encoding and special characters; lengths and length registers
- Font shapes, mixed typesetting, colors

Contents

- TEX system – basic principles
- Fonts, font sources, macros, definition and parameters
- UTF-8 encoding and special characters; lengths and length registers
- Font shapes, mixed typesetting, colors
- Sections, initials; counters and cross references

Contents

- TEX system – basic principles
- Fonts, font sources, macros, definition and parameters
- UTF-8 encoding and special characters; lengths and length registers
- Font shapes, mixed typesetting, colors
- Sections, initials; counters and cross references
- Paragraph typesetting – parameters; margin pars; footnotes; title pages

Contents

- TEX system – basic principles
- Fonts, font sources, macros, definition and parameters
- UTF-8 encoding and special characters; lengths and length registers
- Font shapes, mixed typesetting, colors
- Sections, initials; counters and cross references
- Paragraph typesetting – parameters; margin pars; footnotes; title pages
- Math expressions, math environments and elements; sums, integrals, limits

Contents

- TEX system – basic principles
- Fonts, font sources, macros, definition and parameters
- UTF-8 encoding and special characters; lengths and length registers
- Font shapes, mixed typesetting, colors
- Sections, initials; counters and cross references
- Paragraph typesetting – parameters; margin pars; footnotes; title pages
- Math expressions, math environments and elements; sums, integrals, limits
- Tables – `tabbing` and `tabular` environments; numbers aligning

Contents

- Graphics operations, inserting pictures, `picture` environment, floating environments

Contents

- Graphics operations, inserting pictures, `picture` environment, floating environments
- Running heads, table of contents, list of figures and tables; pages imposition

Contents

- Graphics operations, inserting pictures, `picture` environment, floating environments
- Running heads, table of contents, list of figures and tables; pages imposition
- Document as a whole – document elements, proposal of parameters, realisation

- The first document:

```
\documentclass{article}
\usepackage{xltxtra}
\begin{document}
The first document -- my own text, default
font, size, aligning.
\end{document}
```

- The first document:

```
\documentclass{article}
\usepackage{xltxtra}
\begin{document}
The first document -- my own text, default
font, size, aligning.
\end{document}
```

- Compile this document

- The first document:

```
\documentclass{article}
\usepackage{xltxtra}
\begin{document}
The first document -- my own text, default
font, size, aligning.
\end{document}
```

- Compile this document
- View the result

- The first document:

```
\documentclass{article}
\usepackage{xltxtra}
\begin{document}
The first document -- my own text, default
font, size, aligning.
\end{document}
```

- Compile this document
- View the result
- View the log file

- The first document:

```
\documentclass{article}
\usepackage{xltxtra}
\begin{document}
The first document -- my own text, default
font, size, aligning.
\end{document}
```

- Compile this document
- View the result
- View the log file
- This document we will use and supplement during all the laboratories

- Errors and warnings – write command `\nothing` anywhere into your document

- Errors and warnings – write command `\nothing` anywhere into your document
- View the final PDF

TEX system

- Errors and warnings – write command `\nothing` anywhere into your document
- View the final PDF
- View the log file and search line starting with "!"

- Errors and warnings – write command `\nothing` anywhere into your document
- View the final PDF
- View the log file and search line starting with "!"
- Recognize the error message

- Errors and warnings – write command `\nothing` anywhere into your document
- View the final PDF
- View the log file and search line starting with "!"
- Recognize the error message
- Experiment with longer text – add min. 3 paragraphs (approx. 30 lines)

TEX system

- Errors and warnings – write command `\nothing` anywhere into your document
- View the final PDF
- View the log file and search line starting with "!"
- Recognize the error message
- Experiment with longer text – add min. 3 paragraphs (approx. 30 lines)
- Watch possible warnings – underfull or overfull boxes

Fonts, main font

- Suppose the text from previous laboratory: Default main font Latin Modern

Fonts, main font

- Suppose the text from previous laboratory: Default main font Latin Modern
- Set the main font by command `\setmainfont`

Fonts, main font

- Suppose the text from previous laboratory: Default main font Latin Modern
- Set the main font by command `\setmainfont`
- Experiment with setting `\defaultfontfeatures`

Fonts, main font

- Suppose the text from previous laboratory: Default main font Latin Modern
- Set the main font by command `\setmainfont`
- Experiment with setting `\defaultfontfeatures`
- Set additional fonts: sans serif, typewriter

Macro definition

- The command `\def` and definition of simple commands

Macro definition

- The command `\def` and definition of simple commands
- Commands for solid nonbreaking spaces 1/6 em, 1/8 em

Macro definition

- The command `\def` and definition of simple commands
- Commands for solid nonbreaking spaces 1/6 em, 1/8 em
- Command for figural space

Macro definition

- The command `\def` and definition of simple commands
- Commands for solid nonbreaking spaces 1/6 em, 1/8 em
- Command for figural space
- Commands for default document information: author, name, date etc.

Macro definition

- The command `\def` and definition of simple commands
- Commands for solid nonbreaking spaces 1/6 em, 1/8 em
- Command for figural space
- Commands for default document information: author, name, date etc.
- Alternative definition commands: `\newcommand`, `\renewcommand`

Macro definition

- The command `\def` and definition of simple commands
- Commands for solid nonbreaking spaces 1/6 em, 1/8 em
- Command for figural space
- Commands for default document information: author, name, date etc.
- Alternative definition commands: `\newcommand`, `\renewcommand`
- Environment definition `\newenvironment`, `\renewenvironment`

Macro definition with parameters

- Non-separated parameters, macro for various spaces

Macro definition with parameters

- Non-separated parameters, macro for various spaces
- Macros for some parts of letters (address, signature etc.)

Macro definition with parameters

- Non-separated parameters, macro for various spaces
- Macros for some parts of letters (address, signature etc.)
- Separated parameters and its advantages

Macro definition with parameters

- Non-separated parameters, macro for various spaces
- Macros for some parts of letters (address, signature etc.)
- Separated parameters and its advantages
- Macros for complete structural marking of letter

Macro definition with parameters

- Non-separated parameters, macro for various spaces
- Macros for some parts of letters (address, signature etc.)
- Separated parameters and its advantages
- Macros for complete structural marking of letter
- The style file, link to style file (example of letter marking)

UTF-8 encoding

- Experiment with text from previous laboratory – insert dashes into source text (code 0150 or 0151) and compare it with two (three) hyphens

UTF-8 encoding

- Experiment with text from previous laboratory – insert dashes into source text (code 0150 or 0151) and compare it with two (three) hyphens
- Insert quotes “, ”

UTF-8 encoding

- Experiment with text from previous laboratory – insert dashes into source text (code 0150 or 0151) and compare it with two (three) hyphens
- Insert quotes “, ”
- Create macro `\quote` with 1 parameter for typesetting text in quotes

UTF-8 encoding

- Experiment with text from previous laboratory – insert dashes into source text (code 0150 or 0151) and compare it with two (three) hyphens
- Insert quotes “, ”
- Create macro `\quote` with 1 parameter for typesetting text in quotes
- Insert various national characters – package `polyglossia` and setting `\setdefaultlanguage`

UTF-8 encoding

- Experiment with text from previous laboratory – insert dashes into source text (code 0150 or 0151) and compare it with two (three) hyphens
- Insert quotes “, ”
- Create macro `\quote` with 1 parameter for typesetting text in quotes
- Insert various national characters – package `polyglossia` and setting `\setdefaultlanguage`
- Insert various math symbols and test if it is in main font

UTF-8 encoding

- Experiment with text from previous laboratory – insert dashes into source text (code 0150 or 0151) and compare it with two (three) hyphens
- Insert quotes “, ”
- Create macro `\quote` with 1 parameter for typesetting text in quotes
- Insert various national characters – package `polyglossia` and setting `\setdefaultlanguage`
- Insert various math symbols and test if it is in main font
- Change main font and test presence of special characters

UTF-8 encoding

- Experiment with text from previous laboratory – insert dashes into source text (code 0150 or 0151) and compare it with two (three) hyphens
- Insert quotes “, ”
- Create macro `\quote` with 1 parameter for typesetting text in quotes
- Insert various national characters – package `polyglossia` and setting `\setdefaultlanguage`
- Insert various math symbols and test if it is in main font
- Change main font and test presence of special characters
- Insert and test various national characters from different languages

Lengths

- Use previous text. Predefined lengths: experiment with setting of `\parindent` and `\parskip`

Lengths

- Use previous text. Predefined lengths: experiment with setting of `\parindent` and `\parskip`
- Define new length register for form fill places. Define new macro `\fplace` which creates a hole in a paragraph of the size given by the length register

Lengths

- Use previous text. Predefined lengths: experiment with setting of `\parindent` and `\parskip`
- Define new length register for form fill places. Define new macro `\fplace` which creates a hole in a paragraph of the size given by the length register
- Define a similar macro which creates a hole in a paragraph filled by dots

Lengths

- Use previous text. Predefined lengths: experiment with setting of `\parindent` and `\parskip`
- Define new length register for form fill places. Define new macro `\fplace` which creates a hole in a paragraph of the size given by the length register
- Define a similar macro which creates a hole in a paragraph filled by dots
- Experiment with commands `\hspace` and `\hspace*`

Lengths

- Use previous text. Predefined lengths: experiment with setting of `\parindent` and `\parskip`
- Define new length register for form fill places. Define new macro `\fplace` which creates a hole in a paragraph of the size given by the length register
- Define a similar macro which creates a hole in a paragraph filled by dots
- Experiment with commands `\hspace` and `\hspace*`
- Create macro `\flushbottom` to move any following text to the bottom margin of this page

Lengths

- Use previous text. Predefined lengths: experiment with setting of `\parindent` and `\parskip`
- Define new length register for form fill places. Define new macro `\fplace` which creates a hole in a paragraph of the size given by the length register
- Define a similar macro which creates a hole in a paragraph filled by dots
- Experiment with commands `\hspace` and `\hspace*`
- Create macro `\flushbottom` to move any following text to the bottom margin of this page
- Change the macro `\fplace` so that the size will be from 2 to 4 cm depending on the alignment of the paragraph

Font shapes

- Experiment with text from previous laboratory – insert command for emphasize `\em`

Font shapes

- Experiment with text from previous laboratory – insert command for emphasize `\em`
- Check the nested appearence of emphasize command (it may switch to italic and back to upshape)

Font shapes

- Experiment with text from previous laboratory – insert command for emphasize `\em`
- Check the nested appearence of emphasize command (it may switch to italic and back to upshape)
- Insert command for bold face, both variant (with parameter or as a switch)

Font shapes

- Experiment with text from previous laboratory – insert command for emphasize `\em`
- Check the nested appearence of emphasize command (it may switch to italic and back to upshape)
- Insert command for bold face, both variant (with parameter or as a switch)
- Test if main font has small caps

Font shapes

- Experiment with text from previous laboratory – insert command for emphasize `\em`
- Check the nested appearence of emphasize command (it may switch to italic and back to upshape)
- Insert command for bold face, both variant (with parameter or as a switch)
- Test if main font has small caps
- Create simple macros for typesetting of bibliographic citation: `\bibauthor` (for small caps) and `\bibname` (for italic)

Font shapes

- Experiment with text from previous laboratory – insert command for emphasize `\em`
- Check the nested appearence of emphasize command (it may switch to italic and back to upshape)
- Insert command for bold face, both variant (with parameter or as a switch)
- Test if main font has small caps
- Create simple macros for typesetting of bibliographic citation: `\bibauthor` (for small caps) and `\bibname` (for italic)
- Create macro for code typeset (for typewriter shape)

Colors

- Experiment with previous text – define command `\strong` with one parameter which typeset text in parameter in bold face and blue color. Apply this macro in text for strong emphasizing

Colors

- Experiment with previous text – define command `\strong` with one parameter which typeset text in parameter in bold face and blue color. Apply this macro in text for strong emphasizing
- Select some words in text and typeset it with light gray background

Colors

- Experiment with previous text – define command `\strong` with one parameter which typeset text in parameter in bold face and blue color. Apply this macro in text for strong emphasizing
- Select some words in text and typeset it with light gray background
- Select some words in text and typeset it with red frame

Colors

- Experiment with previous text – define command `\strong` with one parameter which typeset text in parameter in bold face and blue color. Apply this macro in text for strong emphasizing
- Select some words in text and typeset it with light gray background
- Select some words in text and typeset it with red frame
- Experiment with command `\fcolorbox`

Colors

- Experiment with previous text – define command `\strong` with one parameter which typeset text in parameter in bold face and blue color. Apply this macro in text for strong emphasizing
- Select some words in text and typeset it with light gray background
- Select some words in text and typeset it with red frame
- Experiment with command `\fcolorbox`
- Define your own color by command `\definecolor`. Use an RGB color space

Colors

- Experiment with previous text – define command `\strong` with one parameter which typeset text in parameter in bold face and blue color. Apply this macro in text for strong emphasizing
- Select some words in text and typeset it with light gray background
- Select some words in text and typeset it with red frame
- Experiment with command `\fcolorbox`
- Define your own color by command `\definecolor`. Use an RGB color space
- Use your own color to redefine the `\strong` macro

Colors

- Experiment with previous text – define command `\strong` with one parameter which typeset text in parameter in bold face and blue color. Apply this macro in text for strong emphasizing
- Select some words in text and typeset it with light gray background
- Select some words in text and typeset it with red frame
- Experiment with command `\fcolorbox`
- Define your own color by command `\definecolor`. Use an RGB color space
- Use your own color to redefine the `\strong` macro
- Experiment with command `\pagecolor`

Sections

- Work with text form previous laboratory and make sure the text is long enough for multiple pages

Sections

- Work with text form previous laboratory and make sure the text is long enough for multiple pages
- Divide text into two or three sections and apply commands for section titles (`\section`, `\subsection` etc.)

Sections

- Work with text form previous laboratory and make sure the text is long enough for multiple pages
- Divide text into two or three sections and apply commands for section titles (`\section`, `\subsection` etc.)
- Apply star-form commands too (`\section*`)

Sections

- Work with text form previous laboratory and make sure the text is long enough for multiple pages
- Divide text into two or three sections and apply commands for section titles (`\section`, `\subsection` etc.)
- Apply star-form commands too (`\section*`)
- Watch the result form of titles

Sections

- Work with text form previous laboratory and make sure the text is long enough for multiple pages
- Divide text into two or three sections and apply commands for section titles (`\section`, `\subsection` etc.)
- Apply star-form commands too (`\section*`)
- Watch the result form of titles
- Experiment with `\pagestyle` command – parameter `headings`

Sections

- Work with text form previous laboratory and make sure the text is long enough for multiple pages
- Divide text into two or three sections and apply commands for section titles (`\section`, `\subsection` etc.)
- Apply star-form commands too (`\section*`)
- Watch the result form of titles
- Experiment with `\pagestyle` command – parameter `headings`
- Apply command `\markright` or `\markboth`

Sections

- Work with text form previous laboratory and make sure the text is long enough for multiple pages
- Divide text into two or three sections and apply commands for section titles (`\section`, `\subsection` etc.)
- Apply star-form commands too (`\section*`)
- Watch the result form of titles
- Experiment with `\pagestyle` command – parameter `headings`
- Apply command `\markright` or `\markboth`
- Experiment with initials – apply very large first letter in selected paragraph, can be used for example `yinit` font

Counters

- Change the value of page counter, set initial page number to 10

Counters

- Change the value of `page` counter, set initial page number to 10
- Change the shape of `section` counter to upper Roman numerals

Counters

- Change the value of `page` counter, set initial page number to 10
- Change the shape of `section` counter to upper Roman numerals
- Create your own counter for numbering of special paragraphs. Create macro which steps this counter and display its value on start of paragraph in green color

Counters

- Change the value of `page` counter, set initial page number to 10
- Change the shape of `section` counter to upper Roman numerals
- Create your own counter for numbering of special paragraphs. Create macro which steps this counter and display its value on start of paragraph in green color
- Apply previous macro to selected paragraphs

Counters

- Change the value of `page` counter, set initial page number to 10
- Change the shape of `section` counter to upper Roman numerals
- Create your own counter for numbering of special paragraphs. Create macro which steps this counter and display its value on start of paragraph in green color
- Apply previous macro to selected paragraphs
- Create reference to paragraphs labelled with your own counter

Counters

- Change the value of `page` counter, set initial page number to 10
- Change the shape of `section` counter to upper Roman numerals
- Create your own counter for numbering of special paragraphs. Create macro which steps this counter and display its value on start of paragraph in green color
- Apply previous macro to selected paragraphs
- Create reference to paragraphs labelled with your own counter
- Create page reference to selected sections in text

Paragraphs

- Use text from previous laboratory. On selected part apply left aligning

Paragraphs

- Use text from previous laboratory. On selected part apply left aligning
- Create own macro for numbered titles which will typeset centered one

Paragraphs

- Use text from previous laboratory. On selected part apply left aligning
- Create own macro for numbered titles which will typeset centered one
- Experiment with paragraph typesetting into three columns – the environment `multicols` from package `multicol`. Change the parameters of word division algorithm: `\lefthyphenmin` and `\righthyphenmin`

Paragraphs

- Use text from previous laboratory. On selected part apply left aligning
- Create own macro for numbered titles which will typeset centered one
- Experiment with paragraph typesetting into three columns – the environment `multicols` from package `multicol`. Change the parameters of word division algorithm: `\lefthyphenmin` and `\righthyphenmin`
- Use manual hyphens to appropriate word divide

Paragraphs

- Use text from previous laboratory. On selected part apply left aligning
- Create own macro for numbered titles which will typeset centered one
- Experiment with paragraph typesetting into three columns – the environment `multicols` from package `multicol`. Change the parameters of word division algorithm: `\lefthyphenmin` and `\righthyphenmin`
- Use manual hyphens to appropriate word divide
- Use `\leftskip` or `\rightskip` registers to change left and right paragraph margin

Paragraphs

- Use text from previous laboratory. On selected part apply left aligning
- Create own macro for numbered titles which will typeset centered one
- Experiment with paragraph typesetting into three columns – the environment `multicols` from package `multicol`. Change the parameters of word division algorithm: `\lefthyphenmin` and `\righthyphenmin`
- Use manual hyphens to appropriate word divide
- Use `\leftskip` or `\rightskip` registers to change left and right paragraph margin
- Redefine the `\baselinestretch` to change line spacing of selected part of text

Math expressions

- Insert into text from previous laboratory the following math expressions:

Math expressions

- Insert into text from previous laboratory the following math expressions:
-

$$P = \sum_{i=1}^{n} g(A + ik) \tag{1}$$

Math expressions

- Insert into text from previous laboratory the following math expressions:
-
$$P = \sum_{i=1}^{n} g(A + ik) \tag{1}$$
-
$$y = \left[\frac{x(x^2 - 1)}{x + 1} - \frac{x + 1}{x(x - 1)^2} \right]^3$$

Math expressions

- Insert into text from previous laboratory the following math expressions:
-
$$P = \sum_{i=1}^{n} g(A + ik) \tag{1}$$
-
$$y = \left[\frac{x(x^2 - 1)}{x + 1} - \frac{x + 1}{x(x - 1)^2} \right]^3$$
-
$$\begin{aligned} C &\leq 2^N && (2) \\ \log C &\leq N \log 2 \\ N &= \left\lceil \frac{\log C}{\log 2} \right\rceil && (3) \end{aligned}$$

Math expressions

- $$\text{success rate} = \frac{\text{number of correct answers}}{\text{number of examples entered}}$$

Math expressions

- $$\text{success rate} = \frac{\text{number of correct answers}}{\text{number of examples entered}}$$
- $$\lim_{\Delta x \to 0} \frac{f(x) - f(x + \Delta x)}{\Delta x}$$

Math expressions

- $$\text{success rate} = \frac{\text{number of correct answers}}{\text{number of examples entered}}$$
- $$\lim_{\Delta x \to 0} \frac{f(x) - f(x + \Delta x)}{\Delta x}$$
- $$\text{discontinuous function } g(\xi) = \begin{cases} 0 & \text{for } \xi < 0 \\ 2\xi & \text{for } \xi \in (0, 1) \\ 2 & \text{for } \xi \leq 1 \end{cases}$$

Math expressions

- $$\text{success rate} = \frac{\text{number of correct answers}}{\text{number of examples entered}}$$
- $$\lim_{\Delta x \to 0} \frac{f(x) - f(x + \Delta x)}{\Delta x}$$
- $$\text{discontinuous function } g(\xi) = \begin{cases} 0 & \text{for } \xi < 0 \\ 2\xi & \text{for } \xi \in (0, 1) \\ 2 & \text{for } \xi \leq 1 \end{cases}$$
- $$\text{transfer matrix: } \begin{bmatrix} \alpha_{11} & \beta_{12} \\ \gamma_{21} & \delta_{22} \end{bmatrix} \qquad (4)$$

The `tabbing` environment

- Typeset the following table:

The `tabbing` environment

- Typeset the following table:
- **City** **Temperature at 7 AM**

Philadelphia	17.5° C
New York	16.8° C
Washington	15.9° C*
*uncalibrated scale	
Boston	19.5° C
Los Angeles	12.8° C
San Francisco	16.6° C
Anchorage	−0.5° C
Fairbanks	−2.8° C

The `tabbing` environment

- Typeset the following source code:

The `tabbing` environment

- Typeset the following source code:
- **procedure** Number(**var** C: longint);
  var R: **string**;
  Position: byte;
  begin Position:=5;
  while not eof **do begin**
  readln(R);
  //now will be write the main information:
  writeln(C: Position, '–', R)
  end
  end;

The `tabular` environment

- 1. Typeset the following table:

The `tabular` environment

- 1. Typeset the following table:

O.	Country	Score	Points
1.	Italy	23: 4	14
2.	France	18: 7	10
3.	Sweden	14: 10	8
4.	Switzerland	3: 18	3
5.	Austria	1: 19	0

The `tabular` environment

- 1. Typeset the following table:

O.	Country	Score	Points
1.	Italy	23: 4	14
2.	France	18: 7	10
3.	Sweden	14: 10	8
4.	Switzerland	3: 18	3
5.	Austria	1: 19	0

- 2. Typeset your personal timetable

Insert graphics

- Work with file from previous laboratory

Insert graphics

- Work with file from previous laboratory
- Prepare a graphics file of JPG format (from internet or own sources)

Insert graphics

- Work with file from previous laboratory
- Prepare a graphics file of JPG format (from internet or own sources)
- Insert graphics file into document and set it width to 70% of text width

Insert graphics

- Work with file from previous laboratory
- Prepare a graphics file of JPG format (from internet or own sources)
- Insert graphics file into document and set it width to 70% of text width
- Experiment with other parameters of `\includegraphics` command

Insert graphics

- Work with file from previous laboratory
- Prepare a graphics file of JPG format (from internet or own sources)
- Insert graphics file into document and set it width to 70% of text width
- Experiment with other parameters of `\includegraphics` command
- Rotate the previous picture to 35° counterclockwise

Insert graphics

- Work with file from previous laboratory
- Prepare a graphics file of JPG format (from internet or own sources)
- Insert graphics file into document and set it width to 70% of text width
- Experiment with other parameters of `\includegraphics` command
- Rotate the previous picture to 35° counterclockwise
- Reflect selected part of text

The `picture` environment

- Prepare any scheme (blocks, oriented links, texts)

The `picture` environment

- Prepare any scheme (blocks, oriented links, texts)
- Use the `picture` environment to draw prepared scheme

The `picture` environment

- Prepare any scheme (blocks, oriented links, texts)
- Use the `picture` environment to draw prepared scheme
- Use the `\multiput` command to repeat selected part of picture

The `picture` environment

- Prepare any scheme (blocks, oriented links, texts)
- Use the `picture` environment to draw prepared scheme
- Use the `\multiput` command to repeat selected part of picture
- Combine commands for line and text with various colors

The `picture` environment

- Prepare any scheme (blocks, oriented links, texts)
- Use the `picture` environment to draw prepared scheme
- Use the `\multiput` command to repeat selected part of picture
- Combine commands for line and text with various colors
- Use the `picture` environment with zero dimensions for draw the cropmarks

The `picture` environment

- Prepare any scheme (blocks, oriented links, texts)
- Use the `picture` environment to draw prepared scheme
- Use the `\multiput` command to repeat selected part of picture
- Combine commands for line and text with various colors
- Use the `picture` environment with zero dimensions for draw the cropmarks
- Use the `picture` environment with zero dimensions for draw the watermark under text of selected page

Running heads

- Use the document from previous laboratory

Running heads

- Use the document from previous laboratory
- Set the twoside document and default page headings

Running heads

- Use the document from previous laboratory
- Set the twoside document and default page headings
- Redefine the command `\ps@headings` to set your own shape of headings

Running heads

- Use the document from previous laboratory
- Set the twoside document and default page headings
- Redefine the command `\ps@headings` to set your own shape of headings
- Use the command `\markboth` to set the content of running heads

Running heads

- Use the document from previous laboratory
- Set the twoside document and default page headings
- Redefine the command `\ps@headings` to set your own shape of headings
- Use the command `\markboth` to set the content of running heads
- Experiment with `fancyhdr` package for set running heads

Running heads

- Use the document from previous laboratory
- Set the twoside document and default page headings
- Redefine the command `\ps@headings` to set your own shape of headings
- Use the command `\markboth` to set the content of running heads
- Experiment with `fancyhdr` package for set running heads
- Set the name of section to the left heading and the name of subsection to the right heading; set page numbering in bold face to the outer margin of page foot

Table of contents

- Use the document from previous laboratory

Table of contents

- Use the document from previous laboratory
- Create title page of document

Table of contents

- Use the document from previous laboratory
- Create title page of document
- Create special page with table of contents after title page

Table of contents

- Use the document from previous laboratory
- Create title page of document
- Create special page with table of contents after title page
- Add all star-form titles to the table of contents

Table of contents

- Use the document from previous laboratory
- Create title page of document
- Create special page with table of contents after title page
- Add all star-form titles to the table of contents
- Wrap all tables with `table` environment and all pictures with `figure` environment; add all captions

Table of contents

- Use the document from previous laboratory
- Create title page of document
- Create special page with table of contents after title page
- Add all star-form titles to the table of contents
- Wrap all tables with `table` environment and all pictures with `figure` environment; add all captions
- Add list of tables and list of figures after table of contents

Page imposition

- Use the document from previous laboratory and add the `geometry` package

Page imposition

- Use the document from previous laboratory and add the `geometry` package
- Set the page dimensions to format A5 and correct all possible overfulls

Page imposition

- Use the document from previous laboratory and add the `geometry` package
- Set the page dimensions to format A5 and correct all possible overfulls
- Create a new document with `pdfpages` package

Page imposition

- Use the document from previous laboratory and add the `geometry` package
- Set the page dimensions to format A5 and correct all possible overfulls
- Create a new document with `pdfpages` package
- Insert previous document and impose 2 pages to 1 A4 paper in brochure format

Document proposal

- Collect material to the new document (texts, pictures, tables etc.)

Document proposal

- Collect material to the new document (texts, pictures, tables etc.)
- Establish general form of whole document (printed/electronic form, paper format etc.)

Document proposal

- Collect material to the new document (texts, pictures, tables etc.)
- Establish general form of whole document (printed/electronic form, paper format etc.)
- Find all document elements and create an overview on paper

Document proposal

- Collect material to the new document (texts, pictures, tables etc.)
- Establish general form of whole document (printed/electronic form, paper format etc.)
- Find all document elements and create an overview on paper
- Propose the visual parameters for each document element, use the typographics rules

Document proposal

- Collect material to the new document (texts, pictures, tables etc.)
- Establish general form of whole document (printed/electronic form, paper format etc.)
- Find all document elements and create an overview on paper
- Propose the visual parameters for each document element, use the typographics rules
- Establish typographics and logical links between individual document elements

Document realisation

- Prepare text material – solve the use of special characters, divide text into paragraphs

Document realisation

- Prepare text material – solve the use of special characters, divide text into paragraphs
- Use the document proposal and prepare system of structural markup

Document realisation

- Prepare text material – solve the use of special characters, divide text into paragraphs
- Use the document proposal and prepare system of structural markup
- Use proposed structural marks in document material

Document realisation

- Prepare text material – solve the use of special characters, divide text into paragraphs
- Use the document proposal and prepare system of structural markup
- Use proposed structural marks in document material
- Create style file and create all macros for structural markup

Document realisation

- Prepare text material – solve the use of special characters, divide text into paragraphs
- Use the document proposal and prepare system of structural markup
- Use proposed structural marks in document material
- Create style file and create all macros for structural markup
- Prepare and realise tables and math expressions

Document realisation

- Prepare text material – solve the use of special characters, divide text into paragraphs
- Use the document proposal and prepare system of structural markup
- Use proposed structural marks in document material
- Create style file and create all macros for structural markup
- Prepare and realise tables and math expressions
- Insert all pictures and complete the document