## Turinys

## 1. My first document

```
\documentclass[a4paper]{article}
```

For all the cases, we must include these **packages**:

```
\usepackage[utf8]{inputenc}
\usepackage[L7x]{fontenc}
\usepackage[lithuanian]{babel}
\usepackage{lmodern}
```

Finally we can start our first **document**:

```
\begin{document}
Hello, world!
\end{document}
```

## 2. Math mode and mathematical expressions

To write a mathematical expressions we need to switch between normal text and math mode. Therefore, special environments have been declared for this purpose. They can be distinguished into two categories depending on how they are presented:

- **text formulas** are displayed inline, that is, within the body of text where it is declared, for example, I can say that a + a = 2a within this sentence. Use

  ```
  $ ... $
  ```

  or

  ```
  \( ... \)
  ```

- **displayed formulas** are separate from the main text. It's possible to use one of three environments here

  ```
  $$ your formula $$
  ```

  ```
  \[ your formula \]
  ```

```
\begin{equation} your formula \end{equation}
```

**Suggestion:** Using the $$...$$ should be avoided in longer formulas as it may cause problems...

There are even more differences between **text formulas** and **displayed formulas**. For example, if we write a command `\lim_{x\to 0} f(x)` we would get different outputs:

| | |
|---|---|
| `$ \lim_{x\to 0} $` | $\lim_{x\to 0} f(x)$ |
| `$$ \lim_{x\to 0} $$` | $\displaystyle\lim_{x\to 0} f(x)$ |

In order to solve this issue we need to use `\displaystyle` class inside the environment:

```
$ \displaystyle \lim_{x\to 0} $
```

Click **here** to find more basics in typing an expressions.

## 3. Font types and symbols

### 3.1. Font types that I use most.

| | |
|---|---|
| `\textbf{bold}` | I use **bold** to highlight a text |
| `\textit{Italian}` | I use *Italian* when I talk about something abstract |
| `\texttt{scriptfont}` | I use `scriptfont` to write some commands |
| `\textsc{scriptfont}` | I use SCRIPTFONT in paragraph names |
| `\emph{emphasizefont}` | I use *emphasizefont* for additional comments |

Click **here** for more information about fonts.

### 3.2. Mathematical symbols that I use most (in math mode).

| | | | | | |
|---|---|---|---|---|---|
| `\geq` | $\geqslant$ | `\leq` | $\leqslant$ | `\neq` | $\neq$ |
| `\alpha` | $\alpha$ | `\beta` | $\beta$ | `\gamma` | $\gamma$ |
| `\varepsilon` | $\varepsilon$ | `\delta` | $\delta$ | `\omega` | $\omega$ |
| `\varrho` | $\varrho$ | `\lambda` | $\lambda$ | `\mathit O` | $\mathit O$ |
| `\Rightarrow` | $\Rightarrow$ | `\Leftarrow` | $\Leftarrow$ | `\Leftrightarrow` | $\Leftrightarrow$ |
| `\rightarrow` | $\rightarrow$ | `\leftarrow` | $\leftarrow$ | `\leftrightarrow` | $\leftrightarrow$ |
| `\to` | $\to$ | `\equiv` | $\equiv$ | `\lor` | $\lor$ |
| `\&` | $\&$ | `\oplus` | $\oplus$ | `\downarrow` | $\downarrow$ |
| `\bigcup` | $\bigcup$ | `\bigcap` | $\bigcap$ | `\approx` | $\approx$ |
| `\cdot` | $\cdot$ | `\dots` | $\dots$ | `\in` | $\in$ |
| `\subset` | $\subset$ | `\supset` | $\supset$ | `\bot` | $\bot$ |
| `\forall` | $\forall$ | `\exists` | $\exists$ | `\pm` | $\pm$ |

Click **here** for more information about symbols.

### 3.3. Mathematical font (in math mode).

We will need `amsymb` **package** for these:

| | | | |
|---|---|---|---|
| `\mathbb F` | $\mathbb F$ | `\mathbb N` | $\mathbb N$ |
| `\mathbb Z` | $\mathbb Z$ | `\mathbb Q` | $\mathbb Q$ |
| `\mathbb R` | $\mathbb R$ | `\mathbb C` | $\mathbb C$ |

Click **here** for more information about mathematical fonts.

### 3.4. **Special characters.**

| **Command** | **Sample** | **Mode** |
|---|---|---|
| `\%` | % | *both* |
| `\$` | $ | *both* |
| `\&` | & | *both* |
| `\_` | _ | *both* |
| `\}` | } | *both* |
| `\{` | { | *both* |
| `\textasciicircum` | ˆ | *text only (as copyable caret)* |
| `$\wedge$` | ∧ | *math only (as copyable caret)* |
| `use package` ***`textcomp`***<br>`\textquotesingle` | ' | *text only (as straight quotes)* |
| `$'$` | ′ | *math only (as straight quotes)* |
| ² `(push ALT+0178)` | ² | *text only* |
| ³ `(push ALT+0179)` | ³ | *text only* |
| ± `(push ALT+0177)` | ± | *text only* |
| º `(push ALT+0186)` | º | *text only* |

## 4. Advanced examples of mathematical expressions

To use various mathematical expressions in your text, the most probably you'll need `amsmath` **package**. After you gain some experience in typesetting formulas, you'll need to use Tabs to get more clear and tidy in your scripts.

### 4.1. **Dealing with systems of equations.**

| | |
|---|---|
| `$\begin{cases} x+y=4 \\x-y=2 \end{cases}$` | $\begin{cases} x+y=4 \\ x-y=2 \end{cases}$ |
| But *try to be tidy here*:<br>`$ x^2-6x+8=0 \Leftrightarrow \begin{cases}`<br>`      x_1=4 \\`<br>`      x_2=2`<br>`      \end{cases}$` | $x^2-6x+8=0 \Leftrightarrow \begin{cases} x_1=4 \\ x_2=2 \end{cases}$ |
| *We define a new environment* ***sqcases*** *to create a new type of braces:*<br>`\makeatletter`<br>`\newenvironment{sqcases}`<br>`{\matrix@check\sqcases\env@sqcases}`<br>`{\endarray\right.}`<br>`\def\env@sqcases{\let\@ifnextchar`<br>`\new@ifnextchar \left\lbrack \def\arraystretch{1.2}\array{@{}l@{\quad}l@{}}}`<br>`\makeatother`<br>`$\begin{sqcases} x+y=4 \\x-y=2 \end{sqcases}$` | $\left[\begin{array}{l} x_1=4 \\ x_2=2 \end{array}\right.$ |
| *Another way to make systems look nice is to use* ***\left*** *and* ***\right*** *actions in* ***array*** *environment*<br>`\[ \left [\begin{array}{c c c c c c c}`<br>`           x & + & y & + & z &= & 0,\\`<br>`           x & + & 2y & + & 3z &= & 0.\\`<br>`           \end{array}\right\}`<br>`\]` | $\left[\begin{array}{ccccccc} x & + & y & + & z & = & 0, \\ x & + & 2y & + & 3z & = & 0. \end{array}\right.$ |
| `$ 2x_1 - x_2 = 0, \qquad x_2 = 2x_1 $` | $2x_1 - x_2 = 0, \qquad x_2 = 2x_1$ |
| `\[`<br>`\left \{ \begin{array}{ccccccc}`<br>`           a & & & + & c & =& 0\\`<br>`           2a & + & b & + & 4c &=& 0\\`<br>`           -3a & + & 3b & + & 8c &=& 0\\`<br>`           \end{array}\right.`<br><br>`\setlength{\unitlength}{0.75pt}`<br>`\begin{picture} (0,0)`<br>`          \put(0,30){\vector(0, -1){ 25}}`<br>`          \put(3,25){\tiny{$\cdot (-2)$}}`<br>`          \put(30,30){\vector(0, -1){40}}`<br>`          \put(33,25){\tiny{$\cdot 3$}}`<br>`           \end{picture}`<br>`\]` | $\left\{\begin{array}{ccccccc} a & & & + & c & = & 0 \\ 2a & + & b & + & 4c & = & 0 \\ -3a & + & 3b & + & 8c & = & 0 \end{array}\right. \; \downarrow{\scriptstyle \cdot(-2)} \; \downarrow{\scriptstyle \cdot 3}$ |

### 4.2. **Dealing with matrix.**

| | |
|---|---|
| `\[A= \begin{pmatrix} 1 & 1\\`<br>`          2 & 1\\`<br>`          3 & 1\\`<br>`          \end{pmatrix}`<br>`\]` | $A = \begin{pmatrix} 1 & 1 \\ 2 & 1 \\ 3 & 1 \end{pmatrix}$ |
| `\[A= \begin{vmatrix} 1 & 1\\`<br>`          2 & 1\\`<br>`          3 & 1\\`<br>`          \end{vmatrix}`<br>`\]` | $A = \begin{vmatrix} 1 & 1 \\ 2 & 1 \\ 3 & 1 \end{vmatrix}$ |
| `\[`<br>`\begin{pmatrix}`<br>`          1 & 2 & 2 & 5\\`<br>`          0 & 1 & 3 & 1\\`<br>`          1 & 4 & 8 & 7\\`<br>`         -1 & -1 & 1 & -4\\`<br>`\end{pmatrix} \begin{picture}(0, 0)`<br>`         \put(-91, 20){\framebox(10, 10)}`<br>`         \put(-94, 3){\framebox(39, 30)}`<br>`         \end{picture} \]` | $\begin{pmatrix} 1 & 2 & 2 & 5 \\ 0 & 1 & 3 & 1 \\ 1 & 4 & 8 & 7 \\ -1 & -1 & 1 & -4 \end{pmatrix}$ |
| ` \[`<br>`\begin{pmatrix}`<br>`\alpha_{11} & \alpha_{12} & \cdots &\alpha_{1n}\\`<br>`\alpha_{21} & \alpha_{22} & \cdots &\alpha_{2n}\\`<br>`\cdots & \cdots & \ddots &\cdots\\`<br>`\alpha_{m1} & \alpha_{m2} & \cdots &\alpha_{mn}\\`<br>`\end{pmatrix}`<br>`\]` | $\begin{pmatrix} \alpha_{11} & \alpha_{12} & \cdots & \alpha_{1n} \\ \alpha_{21} & \alpha_{22} & \cdots & \alpha_{2n} \\ \cdots & \cdots & \ddots & \cdots \\ \alpha_{m1} & \alpha_{m2} & \cdots & \alpha_{mn} \end{pmatrix}$ |

Click **here** for more information about **picture** environment.

### 4.3. **Other formulas.**

| | |
|---|---|
| `$ \displaystyle \int_0^1 v(t)(24t-6) dt $` | $\displaystyle \int_0^1 v(t)(24t-6) dt$ |
| `$ \displaystyle \lim_{t\to 1} 24t-6 $` | $\displaystyle \lim_{t\to 1} 24t-6$ |
| `$ \displaystyle \sum_{i=0}^{24} 24i-6 $` | $\displaystyle \sum_{i=0}^{24} 24i-6dt$ |
| `$ \overline{a\& b} =\bar{a} \lor \bar{b} $` | $\overline{a\& b} =\bar{a} \lor \bar{b}$ |

## 5. Making it clear

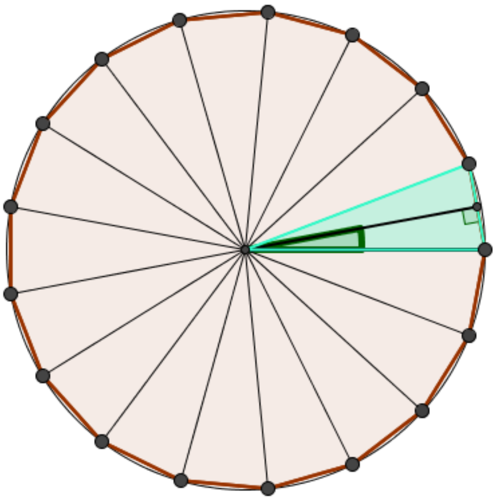
In this section I will run through the main tools that can make your notes extremely clear and also offer an efficient structure that I mostly use.

In order to start a new topic, we begin with theorems, definitions or rules. If this part is going to be long, a clever solution is to save it in a different file and leave just a referrence to this file (*See Next Section*). Otherwise, you may want to keep your explanation clear and brief as much as possible. So far we have seen many environments already such as... More about spaces and boxes More about evironments

### 5.1. **Using minipages.** In order to make your proof easy, sometimes you'll need to provide proofs together with images. In this case you can use a ***minipage*** environment:

| **Command** | **Comments** |
|---|---|
| **`\begin{minipage}`**`[b]{0.48\linewidth}`<br>`Paveikslėlyje pavaizduotas vienetinis apskritimas. Jei įbrėžtinio taisyklingojo daugiakampio viršūnių skaičius artėja į begalybę, tai jo kraštinės ilgis artėja į atkirsto nuo apskritimo lanko ilgiui. Tas pats galioja ir pavaizduotam kampui $\ alpha$. Jo atkirsto lanko ilgis yra lygus $\alpha$, o pusės daugiakampio kraštinės, esančios prieš tą kampą, ilgis yra $\sin \alpha$. Taigi, $\sin \alpha \to \alpha$, kai $\alpha \to 0$`<br>**`\end{minipage}`** | Minipage provides a paragraph box which is a complete mini version of a page;<br>[b] controls the vertical positioning of the text inside the box (valid values: c, t, or b);<br>**`{0.48\linewidth}`** determines the width of the box |
| `\hspace{\fill}` | `\hspace` adds horizontal space between two minipages;<br>**`{\fill}`** sets its lenght (a complete fill) |
| **`\begin{minipage}`**`[b]{0.37\linewidth}`<br>`\includegraphics[width=\textwidth]{"circle".png}`<br>**`\end{minipage}`** | `\includegraphics` includes image;<br>**`[width=\textwidth]`** sets its width to fit into a minipage;<br>**`{"circle".png}`** locates a file; |

*Paveikslėlyje pavaizduotas vienetinis apskritimas. Jei įbrėžtinio taisyklingojo daugiakampio viršūnių skaičius artėja į begalybę, tai jo kraštinės ilgis artėja į atkirsto nuo apskritimo lanko ilgiui. Tas pats galioja ir pavaizduotam kampui $\alpha$. Jo atkirsto lanko ilgis yra lygus $\alpha$, o pusės daugiakampio kraštinės, esančios prieš tą kampą, ilgis yra* $\sin\alpha$. *Taigi,* $\sin\alpha \to \alpha$, *kai* $\alpha \to 0$.



Some issues appears with indentations while using **`minipage`** environment. There is no paragraph indentation in the minipage environment. That is, LaTeX sets **`\parindent`** to zero; however, you may override this with a **`\setlength`** command. To solve confusions with indentation, at the begininig of minipage, use this:

\noindent
\edef\myindent{\the \parindent}
\begin{minipage}[b]{0.78\linewidth}\setlength{\parindent}{\myindent}

5.2. **Using tabulars.** The first way to make your problem explained step by step is to use an ***tabular*** environment. Vertical positioning of text is only possible with ***array*** package. Also, you can put your table to the ***center*** environment to make sure alignment looks nice.

| Command | Comments |
|---|---|
| `\begin{tabular}{\|r l \| p{0.60\textwidth}\|}`<br>`\multicolumn{3}{c}{Lygties sprendimas}\\`<br>`\hline`<br><br>`$\frac{x-2}{3}+1$ & $=\frac{2x}{7}$ &`<br>`Įsitikiname, kad trupmenų vardiklių sandauga 21 nelygi 0, tada iš jos padauginame\\`<br><br>`$7(x-2)+21$ & $=6x$ & \\`<br>`$7x-14+21$ & $=6x$ & \\`<br><br>`$7x+7$ & $=6x$ &`<br>`Vienanarius surenkame į kairę lygties pusę, o laisvuosius narius į dešinę pusę\\`<br><br>`$7x-6x$ &$=-7$ & Sutvarkome lygtį\\`<br>`$x$ & $=-7$ & \\`<br>`\hline`<br><br>`\end{tabular}` | Tabular is the default method to create tables;<br>• 'r' and 'l' aligns the column horizontally (valid values: c, l, r);<br>• '\|' draws a vertical line (valid values: c, l, or r);<br>• 'p' aligns the column vertically (valid values: p, m, b) and sets its width;<br>`\hline` draws a horizontal line; `\multicolumn{3}{c}{txt}` joins columns in a given row:<br>• '{c}' centers its text (valid values: c, l, r);<br>• '{3}' is a number of rows included;<br>• '{txt}' prints a text;<br>Inside tabular:<br>• & separates columns;<br>• \\begins a new line; |

Lygties sprendimas

| | | |
|---:|:---|:---|
| $\frac{x-2}{3}+1$ | $=\frac{2x}{7}$ | Įsitikiname, kad trupmenų vardiklių sandauga 21 nelygi 0, tada iš jos padauginame |
| $7(x-2)+21$ | $=6x$ | |
| $7x-14+21$ | $=6x$ | |
| $7x+7$ | $=6x$ | Vienanarius surenkame į kairę lygties pusę, o laisvuosius narius į dešinę pusę |
| $7x-6x$ | $=-7$ | Sutvarkome lygtį |
| $x$ | $=-7$ | |

5.3. **Using aligns.** The second and easier (especially for equations) way to comment your solution is to use ***aligned*** environment inside formula or ***align**** environment outside formula.

$$\begin{aligned} (a+b)^2 &= (a+b)(a+b) && \text{atskliausime} \\ &= a^2+ab+ab+b^2 && \text{sutrauksime panašius narius} \\ &= a^2+2ab+b^2 \end{aligned}$$

5.4. **Using lists.** Click **here**

5.5. **Using frames.**

5.6. **Visualise yourself!**

- Be colourful! Include a package: `\usepackage[dvipsnames]{xcolor}` and define your own commands `\red{your text}` and `\blue{your text}` supported in both text and math modes:

```
\newcommand{\red}[1]{%
{\color{OrangeRed}#1}}
```

```
\newcommand{\blue}[1]{%
{\color{MidnightBlue}#1}}
```

- Don't hesitate to ~~cancel out~~ your expressions!

| | |
|---|---|
| `$\cancel{\frac{4}{2}}=2$` | $\cancel{\frac{4}{2}}=2$ |
| `$\bcancel{\frac{4}{2}}=2$` | $\bcancel{\frac{4}{2}}=2$ |
| `$\frac{\cancelto{33}{99}}{\cancelto{22}{66}}$` | $\frac{\cancelto{33}{99}}{\cancelto{22}{66}}$ |

Even better, we can choose a color of our marker if we will define a new command:

```
\newcommand\ccancel[2][black]{\renewcommand\CancelColor{\color{#1}}\cancel{#2}}
```

| | |
|---|---|
| `$\ccancel[green]{\frac{444}{222}}=2$` | $\cancel{\frac{444}{222}}=2$ |

- Make your answers boxed! If you've got your answer given as formula use `\fbox` or `\boxed` (inside equations)

| | |
|---|---|
| `\fbox{$x=\{1, -3\}$` | $\boxed{x=\{1,-3\}}$ |
| `\fbox{nėra realiųjų sprendinių}` | $\boxed{\text{nėra realiųjų sprendinių}}$ |
| `$x-2c=0, \quad \boxed{x=2c}$` | $x-2c=0, \quad \boxed{x=2c}$ |

- Make your formulas bolded! E.g. \bm{a=\sin(\alpha)} will print $\boldsymbol{a=\sin(\alpha)}$
- Use Tooltips! [not supported yet..]
- Control the size of your brackets!
- Put a lot of exercises (in one row?)!
- For simple frames, use *framed* environment.

For colourful frames, use *mdframed* environment:

> In any right triangle, the area of the square whose side is the hypotenuse is equal to the sum of the areas of the squares whose sides are the two legs.

```
\begin{mdframed}[backgroundcolor=red!5]
        In any right triangle, the area of the square whose side is the hypotenuse is equa
\end{mdframed}
```

For special frames (that has title, color and other parameters), use *tcolorbox* package and define *newtcolorbox*:

**A physical explanation the *dynamic matrix***

lots of text
a new line
equation

$$\boldsymbol{CDC}^{\dagger} = \boldsymbol{\Omega} = \begin{pmatrix} \omega_1^2 & 0 & \dots & 0 \\ 0 & \omega_2^2 & \dots & 0 \\ \vdots & & \ddots & \vdots \\ 0 & 0 & \dots & \omega_{Nd}^2 \end{pmatrix},$$

where $\boldsymbol{C}$ is a unitary matrix (each column is one of the eigenvectors of the dynamic matrix $\boldsymbol{D}$), $Nd$ is the product of the number of particlces, $N$, and the number of dimensions, $d$.

```
\newtcolorbox{mybox}[1]{colback=red!5!white,colframe=red!75!black,
        fonttitle=\bfseries,title=#1}
\begin{mybox}{A physical explanation the \emph{dynamic matrix}}
lots of text\\
a new line\\
equation
\begin{equation}
    %\label{eq:dynamic_diag}
    \nonumber
```

```
        \bm C \bm D \bm C^{\dagger}=\bm \Omega = \left(\begin{array}{cccc}
            \omega^2_1 & 0 & ... & 0\\
            0 & \omega^2_2 & ... & 0\\
            \vdots &  & \ddots & \vdots \\
            0 & 0 & ... & \omega^2_{Nd}
        \end{array}\right),
\end{equation}
where $\bm C$ is a unitary matrix (each column is one of the eigenvectors of the dynamic m
\end{mybox}
```

Attention! If you use commas in the title you should include *tikz* package Look for colourful frames as well.

- Use games!
- Use minted package to highlight your scripts. IT's WORKING ONLY IF YOU PUT fvextra.sty and usepackage fvextra.

```
import numpy as np

def incmatrix(genl1,genl2):
    m = politics
    n = len(genl2)
    M = None #to become the incidence matrix
    VT = np.zeros((n*m,1), int)  #dummy variable

    #compute the bitwise xor matrix
    M1 = bitxormatrix(genl1)
    M2 = np.triu(bitxormatrix(genl2),1)

    for i in range(m-1):
        for j in range(i+1, m):
            [r,c] = np.where(M2 == M1[i,j])
            for k in range(len(r)):
                VT[(i)n + r[k]] = 1;
                VT[(i)n + c[k]] = 1;
                VT[(j)n + r[k]] = 1;
                VT[(j)n + c[k]] = 1;

                if M is None:
                    M = np.copy(VT)
                else:
                    M = np.concatenate((M, VT), 1)

                VT = np.zeros((n*m,1), int)

    return M
```

Tooltip is missing... Update: minted didn't work! Use listings package instead:

```
$x$ & $=−7$ &\\
\hline
\end{tabular}
```

- Use references! Meanwhile package hyperref (must be loaded) has changed and allows now links to a special page of a pdf files with the command: \href[page=5]{./doc/mydoc.pdf}{thedoc, page 5}
- Use Geogebra generated scripts as images and refer them to .ggb files...
- I have declared some very special commands (copy them from its declarations please). Here goes some examples of how to use them:

| | |
|---|---|
| $a^2+b^2+c^2<br>\e[=][https://lt.wikipedia.org/wiki/Pitagoro_teorema] [a^2+b^2=h^2][Pitagoro teoremą][]<br>[h\text{ yra įstrižainės projekcija į pagrindą}]<br>h^2+c^2<br>\e[=][https://lt.wikipedia.org/wiki/Pitagoro_teorema]<br>[h^2+c^2=H^2][Pitagoro teoremą]<br>H^2$ | $a^2+b^2+c^2$ |

- Here is a link where you can learn about painting cool animations:
- some clues about accessing variables using SciPy
- problems when trying to apply a command line

  Once again:
  Twice again
  Capturing variables from command line.

## 6. **THE NEWS**

Copying all the Macros to a new file sucks. Thus we copy our default .STY FILE instead where we add new updates of Macros that are created every time and include `main` package. Bugs also.

## 7. Notes on programming

basic commands of tikz:
https://www.sharelatex.com/blog/2013/08/27/tikz-series-pt1.html
bounding boxes:
https://tex.stackexchange.com/questions/58292/a-line-of-length-textwidth-in-tikz
tcbsidebyside option:
https://tex.stackexchange.com/questions/99224/putting-text-at-the-side-of-a-tikzpicture
an option tree, pretty nice:
https://tex.stackexchange.com/questions/197903/positioning-nodes-in-tikz-below-left
sun rotates about the moon:
https://tex.stackexchange.com/questions/140273/wrong-frame-size-in-animation
tikzpicture to gif?
https://tex.stackexchange.com/questions/152358/animations-in-latex
ploting sine and converting to a gif:
https://tex.stackexchange.com/questions/291627/draw-an-animated-gif-of-trigonometry-function
how to plot a function correctly?
http://pgfplots.net/tikz/examples/fill-between-plots/

Note that the expansion in `\edef` is done at definition time:

```
\def\a{a}
\def\b{\a}
\edef\c{\a}
\def\a{A}
\a \b \c
```

returns $AAa$

Way to concatenate text:

```
\newcommand{\longertext}[1]{
    \def\myresult{}
    \foreach \x in {1,...,#1}{\global\edef\myresult{a\myresult}}}
\longertext{2}
\myresult
\longertext{12}
\myresult
```

returns aa aaaaaaaaaaaa

A way to use boxes repeatedly (savebox creates a link to a box)

```
\newsavebox{\thisismybox}
\savebox{\thisismybox}{\fbox{\parbox{\textwidth}{t \\ c}}}
\usebox{\thisismybox}
```

Way to capture heights

```
\newlength{\alef}
\settoheight{\alef}{\hbox{t \\ c \\ k}}
\the\alef
```

A way to write to .txt

```
\newwrite\tempfile
\section{First}
\immediate\openout\tempfile=\jobname.tmp
\immediate\write\tempfile{\unexpanded{Section \thesection}}
\immediate\closeout\tempfile

\section{Second}
Something:
\input{\jobname.tmp}
```

A way to read the nth row of .txt

```
\makeatletter
\newread\myread
\newcount\linecnt
\openin\myread=mynodes/array.txt
\@whilesw\unless\ifeof\myread\fi{%
  \advance\linecnt by \@ne
  \readline\myread t\expandafter o\csname line-\number\linecnt\endcsname
}
\makeatother

\csname line-12\endcsname
\csname line-1\endcsname
\csname line-2\endcsname
```

A way to use arrays

```
\newcounter{mycounter}\setcounter{mycounter}{1}

% ary is any prefix you want, it should not exist as a command.

\expandafter\newcommand\csname ary\the\value{mycounter} \endcsname{myfirstelement}
\stepcounter{mycounter}
\expandafter\newcommand\csname ary\the\value{mycounter} \endcsname{mysecondelement}

\csname ary1 \endcsname or

\newcounter{index}\setcounter{index}{2}
\csname ary\the\value{index} \endcsname
```

## 8. What is left

Expand the brackets

Add the fractions

Complete a square

how to create your new macros (including naming theorems, examples and so on), how to include images and use picture vs tikzpicture, how to use command line to create something amazing, links to games that can apply everything, animated pictures (if its possible) and, finally, make an example of any great work (Ishkur's guide).