

L^AT_EX- podstawy

Mateusz Kulikowski Adam Ratajczak

Politechnika Wrocławska

Wrocław, 29 listopada 2007

Plan prezentacji

- 1 Wprowadzenie
- 2 Struktura kodu dokumentu
- 3 Formatowanie tekstu
- 4 Ilustracje i tabele

Plan prezentacji

- 5 Skład matematyczny
- 6 Spisy
- 7 Tworzenie prezentacji
- 8 Dodatkowe informacje
- 9 Literatura

Jak to było na początku. . .

T_EX (wym. tech) jest komputerowym systemem profesjonalnego składu drukarskiego. Litery T, E i X reprezentują greckie litery, odpowiednio tau, epsilon oraz chi, tworzące początek greckiego słowa $\tau\acute{\epsilon}\chi\nu\eta$ (technē) oznaczającego sztukę lub rzemiosło, od którego pochodzi wyraz technika. Jego twórcą jest Donald E. Knuth, amerykański matematyk i informatyk. Program powstał, ponieważ prof. Knuth nie był zadowolony z wyglądu swojej książki „The Art of Computer Programming”.

Historię TeX już znamy a co to takiego LaTeX?

LaTeX (wym. latech) (od [Leslie] Lamport TeX) jest to zestaw makr stanowiących nadbudowę nad systemem składu TeX, automatyzujących wiele czynności związanych z procesem poprawnego składania tekstu. Twórcą pierwszej wersji LaTeX-a był Leslie Lamport.

Sprawy techniczne

Zarówno T_EX jak i L^AT_EX są programami działającymi w konsoli, i operujące na gotowych plikach źródłowych, oznacza to tyle, że do napisania samego źródła potrzebujemy edytora tekstu (emacs, notatnik), a także potrzebujemy program, który umożliwi wyświetlanie wyniku (GSView, AcrobatReader).

Środowisko Linux

System L^AT_EX dla systemu operacyjnego Linuks:

L^AT_EX- <http://www.latex-project.org/>

Środowisko Windows

Dwie najbardziej znane dystrybucje systemu L^AT_EX dla Windows to:

- MiKTeX - <http://www.miktex.org/>
- TeXLive - <http://www.tug.org/tex-live/>

L^AT_EX && dvipdfm

System L^AT_EX podczas kompilacji kodu źródłowego dokumentu tworzy plik typu Dvi dlatego aby otrzymać plik w standardzie Pdf należy użyć programu konwertującego, takiego jak na przykład dvipdfm.

PdfL^AT_EX

Program pdfL^AT_EX służy do bezpośredniej kompilacji kodu źródłowego, a następnie automatycznie przetwarza wynik do formatu Pdf.

Wyświetlanie wyniku

Istnieje wiele programów do wyświetlania plików Pdf, jednym z najbardziej znanych jest Adobe Acrobat Reader, inne to X_Pdf, GSview oparty na GhostScript. Ten ostatni potrafi także wyświetlać pliki PostScript'owe. Wadą Adobe Acrobat Reader'a jest blokowanie na zapis otwartego pliku, co uniemożliwia modyfikację dokumentu z jednoczesnym podglądem.

Aspell

Do sprawdzenia pisowni w dokumentach T_EX można wykorzystać aplikację aspell. Sprawdzenie poprawności pliku (oraz możliwość interaktywnego poprawiania błędów) odbywa się przez wywołanie komendy:

```
aspell -c <nazwa>.tex
```

Edytory

Istnieje całkiem sporo różnych edytorów dedykowanych systemom L^AT_EX zawierających w sobie program do pisania kodu, często z podświetlaniem składni, makropolecenia do kompilacji oraz narzędzie wyświetlające wynik.

- Texmaker(Linux) - <http://www.xm1math.net/texmaker/>
- Kile(Linux) - <http://kile.sourceforge.net/>
- LEd(Windows) - <http://www.latexeditor.org/>

ImageMagick

ImageMagick jest biblioteką służącą do obróbki grafiki w środowisku unix. Jednym z narzędzi znajdujących się w tej bibliotece jest *convert*. Za jego pomocą można w łatwy sposób konwertować obrazki między jednym formatem a drugim. Przykład użycia:
`convert rysunek.jpg rysunek.pdf`

Zarys struktury pliku źródłowego

prolog

```
\documentclass [lista-opcji] {klasa-dokumentu} [data]
```

preambuła

```
\begin{document}
```

treść-dokumentu

```
\end{document}
```

Standardowe klasy dokumentów

- `article` - przeznaczony do krótkich artykułów, sprawozdań itp.
- `report` - dłuższe dokumenty, często rozbijana na częściowe pliki
- `book` - książki, domyślnie druk dwustronny
- `letter` - wszelkiego rodzaju listy, podania
- `slides` - folie projekcyjne

Pakiety

Aby skutecznie używać poleceń nie wchodzących w skład podstawowej wersji L^AT_EX należy zadeklarować użycie pakietów. Do tego celu służy polecenie:

```
\usepackage{nazwa-pakietu}.
```

Polecenie to można użyć jedynie w preambule dokumentu.

Przykładowe pakiety

- `\usepackage[T1]{fontenc}` - wspiera tworzenie polskich znaków diakrytycznych,
- `\usepackage[english,polish]{babel}` - wspiera dokumenty wielojęzyczne,
- `\usepackage[latin2]{inputenc}` - pozwala używać kodowania znaków latin2,
- `\usepackage{graphicx}` - wspiera umieszczanie grafik w dokumencie,

Przykładowe pakiety c.d.

- `\usepackage{amsmath}` - ułatwia formatowanie formuł matematycznych,
- `\usepackage{amssymb}` - zawiera liczne symbole wykorzystywane w składzie matematycznym
- `\usepackage{fancyhdr}` - umożliwia definiowanie własnych główek i stopek stron,
- `\usepackage{showlabels}` - pokazuje na marginesie argument polecenia `\label{etykieta}`

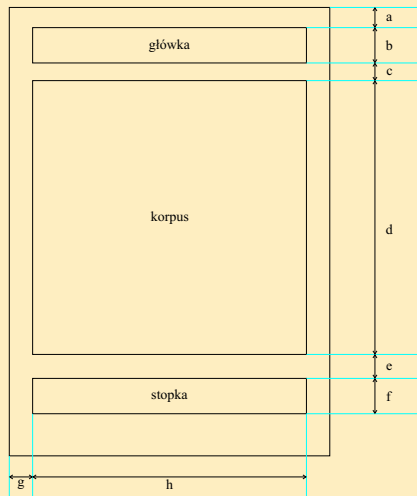
Parametry

a	<code>\topmargin + 1cm</code>
b	<code>\headheight</code>
c	<code>\headsep</code>
d	<code>\textheight</code>
e	<code>\footskip - \footheight</code>
f	<code>\footheight + 1cm</code>
g	<code>\oddsidemargin + 1cm</code>
g	<code>\evensidemargin + 1cm</code>
h	<code>\textwidth</code>

przykład użycia:

```
\setlength{\topmargin}{-1cm}
\setlength{\headheight}{1cm}
```

Układ



Struktura otoczenia

```
\begin{otoczenie} treść-otoczenia \end{otoczenie}
```

Polecenia przydatne na początku dokumentu

- `\title{tytuł}`
- `\author{autor}`
- `\date{data}`
- otoczenie `abstract`
- `\maketitle`

Przykład strony tytułowej

Polecenia podziału dokumentu

- `\part{nazwa}`
- `\chapter{nazwa}` - nie ma w klasie `article`
- `\section{nazwa}`
- `\subsection{nazwa}`
- `\subsubsection{nazwa}`
- `\paragraph{nazwa}`
- `\subparagraph{nazwa}`
- `\appendix{nazwa}`

Kod źródłowy przykładowego podziału

```
\part{Część pierwsza}
\chapter{Rozdział pierwszy}
\section{Sekcja pierwsza}
\subsection{Podsekcja pierwsza}
\subsubsection{Podpodsekcja pierwsza}
\paragraph{Paragraf pierwszy}
\subparagraph{Podparagraf pierwszy}
\section{Sekcja druga}
\subsection{Podsekcja pierwsza}
\appendix
\chapter{Pierwszy dodatek}
\chapter{Drugi dodatek}
```

Przykład podziału

Formatowanie tekstu

Formatowanie tekstu to wszelkiego rodzaju zmiany stylu czcionki, ułożenia tekstu na stronie, wyrównania, wycięcia itp.

Definicja stylu

Definicja stylu składa się z trzech elementów

krój (family)

- antykwia
- bezszeryfowa
- stała szerokość

seria (series)

- półgruba
- normalna

odmiana (shape)

- kursywa
- kapitaliki
- pismo pochyłe

Przykład

Przykład zastosowania poleceń zmiany stylu czcionki

Wcięcia

- `\indent` - wstawia odstęp poziomy równy wcięciu na początku akapitu
- `\noindent` - zapobiega domyślnemu wcięciu na początku akapitu

Justowanie do lewej

```
\begin{flushleft}  
...  
\end{flushleft}
```

Wyśrodkowanie

```
\begin{center}  
...  
\end{center}
```

Justowanie do prawej

```
\begin{flushright}  
...  
\end{flushright}
```

Akapit

Każda pusta linia w kodzie źródłowym powoduje zakończenie bieżącego akapitu i rozpoczęcie nowego

Łamanie wiersza

Do łamania wiersza przydatne są polecenia:

- `\\[dlug]` - zakańcza wiersz (bez „rozstrzeliwania”), opcja *dlug* określa odstęp pionowy do następnego wiersza
- `\newline` - podobne do polecenia `\\`
- `\linebreak` - zakańcza wiersz (z „rozstrzeleniem”)
- `\nolinebreak` - wskazuje miejsce gdzie pod żadnym pozorem nie wolno zakończyć wiersza
- symbol `~` - zakazuje podzieleniu wiersza w miejscu, w którym się znajduje, tzw. „twarda spacja”

Łamanie strony

Polecenia służące do łamania strony:

- `\newpage` - w miejscu umieszczenia zakańcza bieżący akapit i bieżącą stronę
- `\pagebreak` - zakańcza wiersz (z „rozstrzeleniem”) a następnie kończy bieżącą stronę
- `\nopagebreak` - wskazuje miejsce gdzie pod żadnym pozorem nie wolno zakończyć bieżącej strony

Wyliczenia

Do tworzenia wyliczeń służy otoczenie `enumerate`. Otoczenie to można zagnieżdżać.

Źródło

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

Wynik

- 1 ...
 - 1 ...
 - 2 ...
- 2 ...
- 3 ...

Wypunktowania

Do tworzenia wypunktowań służy otoczenie `itemize`. Otoczenie to również można zagnieżdżać.

Źródło

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

Wynik

- ...
 - ...
 - ...
- ...
- ...

Tabele

Do tworzenia wszelkiego rodzaju tabel przeznaczony jest otoczenie tabular.

Źródło

```
\begin{tabular}{||c|r||} \hline
\multicolumn{2}{||c||}{Wspólny wiersz}
      \\ \hline \hline
(1,1) & (1,2) \\
(2,1) & (2,2) \\
(3,1) & (3,2) \\
(4,1) & (4,2) \\
\end{tabular}
```

Wynik

Wspólny wiersz	
(1,1)	(1,2)
(2,1)	(2,2)
(3,1)	(3,2)
(4,1)	(4,2)

Przypisy

Do tworzenia przypisów potrzebne są dwa polecenia:

- `\footnotemark` - generujący symbol odsyłacza ¹
- `\footnote[nr]{treść}` - zawierający odpowiednią etykietę i treść przypisu zamieszczonego stopce strony ^a

^ato jest przypis

Etykieta

Do tworzenia etykiety służy polecenie `\label{etykieta}`. Etykieta może być nadana formule matematycznej, rysunkowi, tabeli a także elementowi podziału dokumentu na sekcje.

Odsyłacz

Aby stworzyć na przykład odsyłacz do rysunku typu: `rys(1)` należy użyć składni:

```
rys(\ref{etykieta})
```

Odsyłacz do bibliografii

Tworzeniu odsyłacza do bibliografii typu: `[1]` służy składnia:

```
\cite{etykieta-pozycji-w-bibliografii}
```

Format otoczeń figure i table

```
\begin{figure}[poz] tekst \end{figure}
```

```
\begin{table}[poz] tekst \end{table}
```

Opcjonalny parametr *poz*

- *b* - ilustracja może pojawić się w dolnej części tekstu,
- *h* - ilustracja może być umieszczona w dokumencie w tym samym miejscu, w którym jest wstawiona w pliku źródłowym,
- *p* - ilustracja może pojawić się na stronie przeznaczonej tylko na ilustracje,
- *t* - ilustracja może pojawić się w górnej części strony tekstu
- *!* - umieści ilustrację tak blisko położenia wskazanego w pliku źródłowym jak tylko możliwe,

Inne polecenia zarządzające rozmieszczeniem ilustracji na stronie

- `\suppressfloats [poz]` zakazuje umieszczania ilustracji na bieżącej stronie
 - `b` - zakazuje umieszczania w dolnej części strony,
 - `t` - zakazuje umieszczania w górnej części strony,
 - `!` użyty w agrumencie `poz` w otoczeniu `figure` lub `table` unieważnia wszystkie polecenia `\suppressfloats`.
- `\clearpage` - zakańcza bieżący akapit i bieżącą stronę. Wszystkie ilustracje trzymane „w pamięci” zostaną umieszczone w dokumencie.

Kod źródłowy

- `\begin{flushleft}`
 ...
`\end{flushleft}`
- `\begin{center}`
 ...
`\end{center}`
- `\begin{flushright}`
 ...
`\end{flushright}}`

Wynik



Grafika wektorowa (EPS)

Zapisywane jako zbiór obiektów i powiązań między nimi.

Właściwości:

- mały rozmiar pliku
- „nieskończona” skalowalność przy zachowaniu ostrości
- łatwe dalsze edytowanie

Grafika rastrowa (JPG)

Zapisywane jako macierz pikseli o określonych kolorach.

Właściwości:

- jedyny sposób na przechowywanie zdjęć
- skalowanie wiąże się z utratą jakości
- stosunkowy duży rozmiar

Grafika w L^AT_EX

Pakiety `graphics`/`graphicx` wraz z poleceniem `\includegraphics` oferują zunifikowane metody włączania plików graficznych.

Składnia polecenia `\includegraphics`

```
\includegraphics [parametry] {nazwa-pliku}
```

- `width` - szerokość grafiki,
- `height` - szerokość grafiki,
- `keepaspectratio` - zachowuje proporcje, nie przekraczając wartości podanych w parametrach `width` i `height`,
- `scale` - skalowanie całego obiektu
- `angle` - określa kąt obrotu.

Różnice pomiędzy L^AT_EX 2_ε a PdfL^AT_EX

W L^AT_EX 2_ε za pomocą polecenia `\includegraphics` można dołączyć pliki typu `*.eps` jak i `*.jpg` natomiast w PdfL^AT_EX należy format `*.eps` przekształcić do `*.pdf` i dopiero wówczas go dołączyć.

Przykład kodu źródłowego dołączającego grafikę

```
\begin{figure}[ht]
  \begin{center}
    \includegraphics[width=7cm]{img/model}
    \caption{Rysunek modelu}{Schematyczny rysunek modelu}
    \label{rys_model}
  \end{center}
\end{figure}
```

składnia polecenia `\caption`:

```
\caption{nazwa-w-spisie}{nazwa-pełna}
```

Pakiety matematyczne

L^AT_EX posiada wbudowany zestaw symboli i formuł matematycznych.

Dodatkowo powstały pakiety rozszerzające te możliwości:

- **amsmath** - ułatwiający formatowanie formuł matematycznych,
- **amssymb** - zawierający pokaźną kolekcję symboli,
- **amscd** - służący do tworzenia diagramów przemiennych,
- **amsfonts** - zawierający dodatkowe czcionki matematyczne.

Otwieranie otoczenia

Jest kilka sposobów otwierania otoczenia matematycznego:

- W wierszu:

Należy pamiętać, że $x=2+3y$ niezależnie od pogody.

Należy pamiętać, że $x = 2 + 3y$ niezależnie od pogody.

- W nowej linii:

W wyniku obliczeń otrzymujemy: $[x=2+3y]$

W wyniku obliczeń otrzymujemy:

$$x = 2 + 3y$$

Otwieranie otoczenia c.d.

- W nowej linii, numerowane :

Ostatecznie wyrażenie przyjmuje postać:

```
\begin{equation} x=3 \end{equation}
```

Ostatecznie wyrażenie przyjmuje postać:

$$x = 3 \tag{1}$$

- W nowej linii, z własnym oznaczeniem (amsmath):

Poniższe wyrażenie sponsorowane jest przez literki S oraz A:

```
\begin{equation} \tag{SA} x=3 \pi^2 \end{equation}
```

Poniższe wyrażenie sponsorowane jest przez literki S oraz A:

$$x = 3\pi^2 \tag{SA}$$

„Zwyczajne” znaki (w składzie matematycznym nie są rozdzielane odstępami) - alfabet łaciński, oraz

0 1 2 3 4 5 6 7 8 9 ! ? . | / @ " ‘

Alfabet grecki, mały i duży

γ	<code>\gamma</code>	Γ	<code>\Gamma</code>
ρ	<code>\rho</code>	P	<code>\mathrm{P}</code>
ϕ	<code>\phi</code>	Φ	<code>\Phi</code>
		(...)	

Matematyczne wersje małych greckich liter

∂	<code>\partial</code>	ϑ	<code>\vartheta</code>	ϱ	<code>\varrho</code>
ε	<code>\varepsilon</code>	ϖ	<code>\varpi</code>	φ	<code>\varphi</code>

Pismo kaligraficzne, tablicowe (amsfonts) i gotyckie (amsfonts)

\mathcal{A}	<code>\mathcal{A}</code>	\mathbb{A}	<code>\mathbb{A}</code>	\mathfrak{A}	<code>\mathfrak{A}</code>
				\mathfrak{a}	<code>\mathfrak{a}</code>
\mathcal{B}	<code>\mathcal{B}</code>	\mathbb{B}	<code>\mathbb{B}</code>	\mathfrak{B}	<code>\mathfrak{B}</code>
\mathcal{C}	<code>\mathcal{C}</code>	\mathbb{C}	<code>\mathbb{C}</code>	\mathfrak{C}	<code>\mathfrak{C}</code>
\mathcal{D}	<code>\mathcal{D}</code>	\mathbb{D}	<code>\mathbb{D}</code>	\mathfrak{D}	<code>\mathfrak{D}</code>
				(...)	

Standardowy zestaw symboli matematycznych

\forall	<code>\forall</code>	\exists	<code>\exists</code>	\nexists	<code>\nexists</code>
\neg	<code>\neg</code>	\emptyset	<code>\emptyset</code>	∞	<code>\infty</code>
\mathbb{P}	<code>\mathbb{P}</code>	\mathbb{S}	<code>\mathbb{S}</code>	\eth	<code>\eth</code>
\Re	<code>\Re</code>	\Im	<code>\Im</code>	\mho	<code>\mho</code>
				(...)	

Znaki diakrytyczne

Znaki diakrytyczne - należy pamiętać, że są one stałej szerokości:

a'	a'	b''	b''	\bar{c}	<code>\bar{c}</code>
\dot{d}	<code>\dot{d}</code>	\ddot{e}	<code>\ddot{e}</code>	\tilde{f}	<code>\tilde{f}</code>
\hat{g}	<code>\hat{g}</code>	\check{h}	<code>\check{h}</code>	\vec{v}	<code>\vec{v}</code>

Podkreślenia

Podkreślenia - pozwalają oznaczyć wyrażenie dowolnej wielkości:

$\overline{x+y}$	<code>\overline{x+y}</code>
$\overrightarrow{x+y}$	<code>\overrightarrow{x+y}</code>
$\underline{x+y}$	<code>\underline{x+y}</code>
$\underbrace{x+y}$	<code>\underbrace{x+y}</code>

Ułamki

Do tworzenia ułamków służy polecenie `\frac`:

```
\[ \frac{x + y}{x + 1} \]
```

$$\frac{x + y}{x + 1}$$

Indeksy

Indeksy górne i dolne tworzymy zgodnie z intuicją:

```
\[ z_{12}^3 = (x + 3y)^{2 \pi^2} \]
```

$$z_{12}^3 = (x + 3y)^{2\pi^2}$$

Opisywanie formuł

Opisywanie formuł najprościej zrealizować za pomocą polecenia `\overbrace` lub `\underbrace`, oraz odpowiednich indeksów:

```
\[ \ddot{q} = \ddot{q}_d
      -K_d (\underbrace{\dot{q} - \dot{q}_d}_{\dot{e}})
      -K_p (\underbrace{q - q_d}_e)\]
```

$$\ddot{q} = \ddot{q}_d - K_d \underbrace{(\dot{q} - \dot{q}_d)}_{\dot{e}} - K_p \underbrace{(q - q_d)}_e$$

Granice

Do tworzenia granic służy operator `\lim`:

```
\[ \lim_{n \to \infty} \frac{1}{n}=0 \]
```

$$\lim_{n \rightarrow \infty} \frac{1}{n} = 0$$

Całki, sumy ...

W podobny sposób tworzymy całki, sumy, itp:

```
\[ \int_{x=0}^{\infty} f(x)dx +  
\int\limits_{x=2}^{998} g(x) dx +  
\sum\limits_{i=1}^{-\infty} x^i \]
```

$$\int_{x=0}^{\infty} f(x)dx + \int_{x=2}^{998} g(x)dx + \sum_{i=1}^{-\infty} x^i$$

Operatory jednoargumentowe

Operatory jednoargumentowe - są to przeważnie funkcje jednej zmiennej. Można wydzielić dwa rodzaje w zależności od sposobu dołączania indeksów:

- Operatory z indeksami w stylu wstawionym (nad/pod operatorem):

det	<code>\det</code>	lim	<code>\lim</code>	sup	<code>\sup</code>
min	<code>\min</code>	max	<code>\max</code>	lim sup	<code>\limsup</code>

- Operatory z indeksami w stylu dołączonym (obok operatora):

sin	<code>\sin</code>	cosh	<code>\cosh</code>	exp	<code>\exp</code>	lg	<code>\lg</code>
ker	<code>\ker</code>	dim	<code>\dim</code>	arccos	<code>\arccos</code>	deg	<code>\deg</code>

Operatory dwuargumentowe

Operatory dwuargumentowe - tworzą wyrażenie z dwóch otaczających je wyrażeń, rozdzielając je krótkim odstępem.

Przykładowe operatory to:

$x + y$	<code>x+y</code>	$x \cdot y$	<code>x \cdot y</code>	$x \bullet y$	<code>x \bullet y</code>
$\pm x$	<code>\pm x</code>	$\mp x$	<code>\mp x</code>	$X \times Y$	<code>X \times Y</code>
$A \cup B$	<code>A \cup B</code>	$A \oplus B$	<code>A \oplus B</code>	$A \boxdot B$	<code>A \boxdot B</code>

Symbole relacji

Symbole relacji - odstępy między wyrażeniami są szerokie. Przykładowe symbole to:

$$\begin{array}{cccccc}
 x < y & x < y & x \ll y & x \lll y & x \lll y & x \lllless y \\
 x \leq y & x \leq y & x \in y & x \in y & x \supset y & x \supseteq y \\
 x \sim y & x \sim y & x \approx y & x \approx y & x = y & x = y
 \end{array}$$

Symbole relacji - zaprzeczenia

Istnieje też zestaw zaprzeczeń większości powyższych symboli:

$$\begin{array}{cccccc}
 x \not< y & x \not< y & x \notin y & x \notin y & x \neq y & x \neq y \\
 x \not\leq y & x \not\leq y & x \not\parallel y & x \not\parallel y & x \not\geq y & x \not\geq y
 \end{array}$$

Strzałki

Dostępna jest również pokaźna kolekcja strzałek:

$x \leftarrow y$	<code>x \leftarrow y</code>	$x \twoheadrightarrow y$	<code>x \twoheadrightarrow y</code>
$x \mapsto y$	<code>x \mapsto y</code>	$x \rightrightarrows y$	<code>x \rightrightarrows y</code>
$x \rightsquigarrow y$	<code>x \rightsquigarrow y</code>	$x \multimap y$	<code>x \multimap y</code>
$x \nrightarrow y$	<code>x \nrightarrow y</code>	$x \nRightarrow y$	<code>x \nRightarrow y</code>

Macierze

Macierze można konstruować za pomocą otoczenia `array`:

```
\[A=\left[ \begin{array}{cccc}
  a_{11} & a_{12} & \cdots & a_{1q} \\
  a_{21} & a_{22} & \cdots & a_{2q} \\
  \vdots & \vdots & \ddots & \vdots \\
  a_{p1} & a_{p2} & \cdots & a_{pq}
\end{array} \right]
```

$$A = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1q} \\ a_{21} & a_{22} & \cdots & a_{2q} \\ \vdots & \vdots & \ddots & \vdots \\ a_{p1} & a_{p2} & \cdots & a_{pq} \end{bmatrix}$$

Definiowanie przez przypadki

Do rozpatrywania różnych przypadków najprościej wykorzystać otoczenie `cases` (`amsmath`):

```
\[sgn(x) = \begin{cases}
    1, & \text{dla } \$x > 0\$ \\
    0, & \text{dla } \$x = 0 \$ \\
    -1, & \text{dla } \$x < 0 \$ \\
\end{cases}
\]
```

$$sgn(x) = \begin{cases} 1, & \text{dla } x > 0 \\ 0, & \text{dla } x = 0 \\ -1, & \text{dla } x < 0 \end{cases}$$

Wyprowadzenia

Wyprowadzenia można przeprowadzać w otoczeniach `eqnarray*` (bez numeracji), oraz `eqnarray` (z numeracją). Otwierają one tryb matematyczny, etykiety można umieścić w każdej linii:

```
\begin{eqnarray}
f(x,y) & = & (x-y)(x+y)^2 \label{AA} \\
& = & 0. \label{DD}
\end{eqnarray} \quad \ref{DD} \text{ jest prawdziwe np. gdy } \$x=y=0\$.
```

$$f(x,y) = (x-y)(x+y)^2 \quad (2)$$

$$= 0. \quad (3)$$

(3) jest prawdziwe np. gdy $x = y = 0$.

Konstrukcje ułamkopodobne

Polecenie `genfrac` (`amsmath`) oraz jego uproszczone wersje pozwalają tworzyć konstrukcje ułamkopodobne, na przykład:

```
\[ \binom{n}{i}=\frac{n!}{i!(n-i)!} \]
```

```
\[ \genfrac{}{}{0pt}{}{n}{i}=
\genfrac{}{}{}{}{n!}{i!(n-i)!} \]
```

$$\binom{n}{i} = \frac{n!}{i!(n-i)!}$$

$$\binom{n}{i} = \frac{n!}{i!(n-i)!}$$

Wielopoziomowe ułamki

Budowanie wielopoziomowych ułamków nie jest problematyczne:

```
\[
f(x) = \frac{ \frac{x^5+3x^2}{x+1} }
{ \frac{x^3+3x}{x+2} + \frac{\log(x)}{x-1+
\frac{x+1}{x-1}} } +
\frac{1}{x + \frac{2}{x + \frac{3}{x + \dots}}}
\]
```

$$f(x) = \frac{\frac{x^5+3x^2}{x+1}}{\frac{x^3+3x}{x+2} + \frac{\log(x)}{x-1+\frac{x+1}{x-1}}} + \frac{1}{x + \frac{2}{x + \frac{3}{x + \dots}}}$$

Spis treści

Tworzeniu spisu treści służy jedno jedyne polecenie `\tableofcontents`. Polecenie to w miejscu w którym się znajduje tworzy spis treści na podstawie poleceń podziału dokumentu na części i sekcje.

Spis ilustracji

Aby stworzyć spis ilustracji wystarczy użyć polecenia `\listoffigures`. W miejscu zastosowania zostaje utworzona lista ilustracji.

Spis tabel

Podobnie aby stworzyć spis tabel należy użyć polecenia `\listoftables`. W miejscu, w którym się znajduje utworzona zostanie lista tabel.

Tworzenie bibliografii

Do tworzenia spisu literatury służy otoczenie `thebibliography`

Źródło

```
\begin{thebibliography}  
\bibitem{Brzechwa}  
,,Bajki'', Jan Brzechwa  
\bibitem{Tuwim}  
,,Lokomotywa'', Julian Tuwim  
\end{thebibliography}
```

Wynik



„Bajki”, Jan Brzechwa



„Lokomotywa”, Julian
Tuwim

Własności

- Beamer jest małym pakietem służącym do tworzenia interaktywnych prezentacji.
- Wygenerowane prezentacje mają postać plików pdf.
- Generalnie nie nadają się one do druku.
- W dokumentacji Beamera znajduje się dużo przykładów, oraz gotowych szablonów, z których można, a nawet trzeba skorzystać.
- Nie wszystkie polecenia działające w środowisku L^AT_EX działają w Beamerze...

```
%klasa - beamer
\documentclass {beamer}
%ulatwia prace z obrazkami
\usepackage{pgf}
%motyw
\usetheme{Warsaw}
\useoutertheme{smoothtree}
\usecolortheme{crane}
%tytul, autor
\author{Mateusz Kulikowski \and Adam Ratajczak}
\title{Beamer - podstawy}
\institute{Politechnika Wrocławska}
\date {Wrocław, \today}
%\titlegraphics{}
%\keywords{}
```

Tworzenie slajdu

```
\begin{frame}
  \frametitle{Tytuł ramki}
  Tekst w ramce
  \begin{block}{Tytuł bloku}
    Zawartość bloku
  \end{block}
  \begin{block}{Blok 2}
    \begin{itemize}
      \item a
      \item b
      \item c
    \end{itemize}
  \end{block}
\end{frame}
```

Tytuł ramki

Tekst w ramce

Tytuł bloku

Zawratość bloku

Blok 2

- a
- b
- c

Kolumny

```
\begin{columns}
  \column{5cm}
  \begin{block}{Blok a}
    Tekst a.
  \end{block}
  \column{3cm}
  \begin{block}{Blok b}
    Tekst b.
  \end{block}
  \column{2cm}
  \begin{block}{Blok c}
    Tekst c.
  \end{block}
\end{columns}
```

Kolumny - przykład

Blok a

Tekst a.

Blok b

Tekst b.

Blok c

Tekst c.

Przejścia

```
\begin<2->{block}{Blok C}  
  \begin{itemize}  
    \item<2->a  
    \item<3>c  
    \item<4->b  
    \item<5->c  
  \end{itemize}  
\end{block}
```

Przejścia - przykład

Blok C

- a
- b
- c
- d

Założenia

Zamierzamy podzielić dokument na pliki tak, aby w każdy rozdział był w osobnym pliku. Dla każdego rozdziału stwórzmy *plik szkieletowy (root file)* oraz *plik danych (data file)*. Nazwijmy je odpowiednio `rootXXa.tex` i `dataXXa.tex` (w miejsce `XX` należy wstawić kolejny numer). Ponadto utwórzmy dwa dodatkowe pliki `nasz_styl.sty` oraz `common.tex`.

Przykładowa zawartość plików

nasz_styl.tex

```
% w tym pliku należy  
% umieścić wszelkie  
% definicje stylu  
\usepackage{...  
...  
...
```

common.tex

```
\include{data00a}  
\include{data01a}  
\include{data02a}  
\include{data03a}
```

Przykładowa zawartość plików c.d.

root00a.tex

```
\documentclass{report}
\usepackage{nasz_styl}
\title{Tytuł}
\author{Nazwisko i imię}
\includeonly{data00a}
\begin{document}
\maketitle
\tableofcontents
\listoffigures
\listoftables
\include{data00a}
\input{common}
\end{document}
```

data00a.tex

```
\chapter*{Przedmowa}
\addcontentsline{toc}
    {chapter}{Przedmowa}
%
Treść przedmowy...
%
\chapter*{Podziękowania}
\addcontentsline{toc}
    {chapter}{Podziękowania}
%
Treść podziękowań...
%
```

Przykładowa zawartość plików c.d.

root01a.tex

```
\documentclass{report}
\usepackage{nasz_styl}
\includeonly{data01a}
\begin{document}
\input{common}
\end{document}
```

⋮

root03a.tex

```
\documentclass{report}
\usepackage{nasz_styl}
\includeonly{data01a}
\begin{document}
\input{common}
\end{document}
```

data01a.tex

```
\chapter{Rozdział pierwszy}
\typeout{Rozdział pierwszy}
%
Treść rozdziału
pierwszego ...
```

⋮






data03a.tex

```
\chapter{Rozdział trzeci}
\typeout{Rozdział trzeci}
%
Treść rozdziału
trzeciego ...
```

Pliki tworzone przez L^AT_EX

- aux - informacje o różnego rodzaju odsyłaczach, numerach stron itp.
- log - zapis terminala podczas pracy L^AT_EX,
- toc - plik spisu treści,
- lof - plik spisu rysunków,
- lot - plik spisu tabel,
- idx - plik skorowidza,
- glo - podobnie jak idx,

Literatura

-  „L^AT_EX wiersz po wierszu”, Antoni Diller
-  „Nie za krótkie wprowadzenie do systemu L^AT_EX_{2 ϵ} ”, Tobias Oetiker
-  „Włączanie grafik do tekstów w L^AT_EX_{2 ϵ} ”, Wojciech Myszka
-  Przykładowy dokument, Krzysztof Arent
-  Przykładowy dokument, Bogdan Kreczmer

Dziękujemy
Owocnej pracy z \LaTeX