

# Nie za krótkie wprowadzenie do systemu L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub>

---

*Albo L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub> w 173 minuty*

**Tobias Oetiker**

**Hubert Partl, Irene Hyna i Elisabeth Schlegl**

PRZEKŁAD POLSKI:

**Tomasz Przechlewski i Ryszard Kubiak**

**Janusz Gołdasz**

**Marcin Serwin**

Wersja 6.4PL1 z 13 maja 2022

Copyright ©1995-2021 Tobias Oetiker and Contributors.

Copyright ©1999, 2007, 2022 for the Polish translation and extension Tomasz Przechlewski, Ryszard Kubiak, Janusz Góldasz and Marcin Serwin.

All rights reserved.

This document is free; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This document is distributed in the hope that it will be useful, but *without any warranty*; without even the implied warranty of *merchantability* or *fitness for a particular purpose*. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this document; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301, USA.

Copyright ©1995–2021 Tobias Oetiker oraz współautorzy.

Copyright ©1998, 2007, 2022 polskiego tłumaczenia i opracowania Tomasz Przechlewski, Ryszard Kubiak, Janusz Góldasz oraz Marcin Serwin.

Wszystkie prawa zastrzeżone.

Niniejszy dokument jest wolno dostępny; można go rozpowszechniać i/lub zmieniać na warunkach Powszechnej Licencji Publicznej GNU, wydanej przez Fundację Wolnego Oprogramowania - według wersji 2-giej tej Licencji lub (wedle własnego uznania) którejś z późniejszych wersji.

Niniejszy dokument jest rozpowszechniany w nadziei, że będzie użyteczny, jednakże *bez jakiegokolwiek gwarancji*, nawet domyślnej gwarancji *przydatności handlowej* albo *przydatności do określonych zastosowań*. Więcej szczegółów znajdziesz w Powszechnej Licencji Publicznej GNU.

Do dokumentu powinna być dołączona kopia Powszechnej Licencji Publicznej GNU (GNU General Public License); jeśli jej nie ma, to napisz do: Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301, USA.

The full text of the GNU General Public License can be found in Appendix **B** on page **147**.

Pełny tekst Powszechnej Licencji Publicznej GNU możesz znaleźć w dodatku **B** na stronie **147**.

# Podziękowania

Większość materiału w niniejszej książce pochodzi z austriackiego wprowadzenia do L<sup>A</sup>T<sub>E</sub>Xa 2.09, napisanego w języku niemieckim przez:

Hubert Partl <[partl@mail.boku.ac.at](mailto:partl@mail.boku.ac.at)>  
*Zentraler Informatikdienst der Universität für Bodenkultur Wien*

Irene Hyna <[Irene.Hyna@bmwf.ac.at](mailto:Irene.Hyna@bmwf.ac.at)>  
*Bundesministerium für Wissenschaft und Forschung Wien*

Elisabeth Schlegl <[noemail](mailto:noemail)>  
*in Graz*

Osoby zainteresowane wersją niemiecką, mogą znaleźć wersję zaktualizowaną do do L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub>, opracowaną przez Jörga Knappena pod adresem [CTAN://info/lshort/german](mailto:CTAN://info/lshort/german).

Niżej wymienione osoby pomogły swoimi poprawkami, sugestiami i materiałami w ulepszaniu niniejszego dokumentu. Przyczyniły się one bardzo do nadania niniejszej książce jej obecnego kształtu. Chciałbym im za to serdecznie podziękować. Za wszelkie błędy, które znajdziesz w tekście, ponoszę oczywiście odpowiedzialność wyłącznie ja sam. Wszystkie natomiast słowa zapisane bez błędu są w książce wyłączną zasługą którejś osoby z poniższej listy.

Jeśli chcesz przyczynić się do rozwoju niniejszej książki możesz znaleźć cały kod źródłowy pod adresem <https://github.com/oetiker/lshort><sup>1</sup>. Twoje pull requesty będą mile widziane.

Eric Abrahamsen, Lenimar Nunes de Andrade, Eilinger August, Rosemary Bailey, Barbara Beeton, Marc Bevand, Connor Blakey, Salvatore Bonaccorso, Pietro Braione, Friedemann Brauer, Markus Brühwiler, Jan Busa, David Carlisle, Neil Carter, Carl Cerecke, Mike Chapman, Pierre Chardaire, Xingyou Chen, Christopher Chin, Diego Clavadetscher, Wim van Dam, Benjamin Deschwenden Jan Dittberner, Michael John Downes, Matthias Dreier, David Dureisseix, Hans Ehrbar, Elliot, Rockrush Engch, William Faulk, Robin Fairbairns, Johan Falk, Jörg Fischer, Frank Fischli, Daniel Flipó, Frank, Mic Milic Frederickx, David Frey, Erik Frisk, Hans Fugal, Robert Funnell, Greg Gamble, Andy Goth, Cyril Goutte, Kasper B. Graversen, Arlo Griffiths, Alexandre Guimond, Neil Hammond, Christoph Hamburger, Rasmus Borup Hansen, Joseph Hilferty, Daniel Hirsbrunner, Martien Hulsen, Björn Hvittfeldt, Morten Høgholm, Werner Icking, Eric Jacoboni, Jakob, Alan Jeffrey, Martin Jenkins, Byron Jones, David Jones, Johannes-Maria Kaltenbach, Nils Kanning, Andrzej Kawalec, Christian Kern, Alain Kessi, Axel Kielhorn, Sander de Kievit, Kjetil Kjernsmo, Tobias Klauser, Jörg Knappen, Michael Koundouros, Matt Kraai, Tobias Krewer, Flori Lambrechts, Mike Lee, Maik Lehradt, Rémi Letot, Axel Liljencrantz, Jasper Loy, Johan Lundberg, Martin Maechler, Alexander Mai, Claus Malten, Kevin Van Maren, Pablo Markin, I. J. Vera Marín, Hendrik Maryns, Chris McCormack, Aleksandar S. Milosevic, Henrik Mitsch, Stefan M. Moser, Armin Müller, Philipp Nagele, Richard Nagy, Manuel Oetiker, Urs Oswald, Hubert Partl, Marcelo Pasin, Martin Pfister, Lan Thuy Pham, Breno Pietracci, Demerson Andre Polli, Maksym Polyakov, Nikos Pothitos, John Reffing, Mike Ressler, Brian Ripley, Kurt Rosenfeld, Bernd Rosenlecher, Chris Rowley, Young U. Ryu, Risto Saarelma, András Salamon, José Carlos Santos, Christopher Sawtell, Gilles Schintgen, Craig Schlenker, Hanspeter Schmid, Baron Schwartz, John Scott, Jordi Serra i Solanich, Miles Spielberg, Susan Stewart, Matthieu Stigler, Geoffrey Swindale, Laszlo Szathmary, Boris Tobotras, Josef Tkadlec, Scott Veirs, Didier Verna, Carl-Gustav Werner, Fabian Wernli, Matthew Widmann, David Woodhouse, Chris York, Rick Zacccone,

<sup>1</sup>Z kolei kod źródłowy tego tłumaczenia znajduje się pod adresem <https://gitlab.com/marcin-serwin/lshort-pl>

Fritz Zaucker, oraz Mikhail Zotov.

**Od autorów polskiej wersji**

Za pomoc redakcyjną dziękujemy Staszce Wawrykiewiczowi i Włodkowi Macewiczowi.



# Przedmowa

$\text{\LaTeX}$  [1] jest systemem składu znakomicie nadającym się do tworzenia publikacji naukowych i technicznych o wysokiej jakości typograficznej. Nadaje się również do przygotowywania dowolnego rodzaju dokumentów, poczynając od prostych listów, a kończąc na grubych książkach. Do formatowania dokumentów  $\text{\LaTeX}$  wykorzystuje program  $\text{\TeX}$  [2].

Niniejsze krótkie wprowadzenie opisuje system  $\text{\LaTeX} 2_{\epsilon}$  i powinno być wystarczające w większości zastosowań. Pełny opis  $\text{\LaTeX}$ a możesz znaleźć w [1, 3].

Wprowadzeni podzielone jest na sześć rozdziałów:

**Rozdział 1** przedstawia ogólną strukturę dokumentów  $\text{\LaTeX}$ owych. Poznasz w nim również nieco historię systemu  $\text{\LaTeX}$ . Po przeczytaniu tego rozdziału zrozumiesz przybliżony sposób działania  $\text{\LaTeX}$ a.

**Rozdział 2** wchodzi w szczegóły dotyczące składania dokumentów. Omówiono w nim większość istotnych instrukcji i otoczeń w  $\text{\LaTeX}$ u. Po przeczytaniu tego rozdziału będziesz w stanie tworzyć swoje pierwsze dokumenty z wypunktowanymi listami, tabelkami, grafikami oraz wstawkami niesztynnymi.

**Rozdział 3** poświęcono składaniu wzorów matematycznych. Zamieszczone w nim przykłady nauczą Cię wykorzystywać jeden z największych atutów  $\text{\LaTeX}$ a. Na końcu rozdziału zamieszczono zestawienie wszystkich dostępnych w  $\text{\LaTeX}$ u symboli matematycznych.

**Rozdział 4** wyjaśnia indeksy (skorowidze), bibliografię oraz kilka drobniejszych aspektów tworzenia PDFów.

**Rozdział 5** pokazuje, jak używać  $\text{\LaTeX}$ a do tworzenia grafiki. Zamiast przygotowywać rysunek w jakimś programie graficznym, zapisać go do pliku, po czym włączyć do dokumentu  $\text{\LaTeX}$ owego, opisujesz rysunek w dokumencie, a  $\text{\LaTeX}$  narysuje go za ciebie.

**Rozdział 6** zawiera potencjalnie niebezpieczne, informacje, mówiące o tym, jak można zmienić standardowy układ graficzny  $\LaTeX$ owych dokumentów. Pokaże on w jaki sposób zmienić ich aspekty tak by piękne wyjście  $\LaTeX$ a stało się ohydne bądź zachwycające, w zależności od Twoich umiejętności.

Ważnym jest by przeczytać wszystkie rozdziały, w powyższej kolejności—książka nie jest w końcu zbyt gruba. Szczególną uwagę zwróć na przykłady, gdyż właśnie w nich zawarto sporo wartościowych informacji.

$\LaTeX$  jest dostępny na większości platform sprzętowych, począwszy od PC i Macintosh, a skończywszy na dużych systemach wyposażonych w system UNIX czy VMS. Na wielu sieciach uniwersyteckich można spotkać gotowe do użytku instalacje  $\LaTeX$ a. Informacje, jak rozpocząć pracę w lokalnej instalacji  $\LaTeX$ a, można znaleźć w *Local Guide (Przewodnik lokalny)* [4]. Jeżeli masz problemy z rozpoczęciem pracy z  $\LaTeX$ em, poproś o pomoc osobę, od której otrzymałeś niniejszą książkę. Celem tego dokumentu *nie* jest przekazanie Ci informacji związanych z instalowaniem i konfigurowaniem systemu  $\LaTeX$ , lecz nauczenie Cię jak pisać dokumenty, aby mogły być przetwarzane przez  $\LaTeX$ a.

Jeśli potrzebujesz jakichkolwiek materiałów dotyczących  $\LaTeX$ a, to zajrzyj do jednego z Comprehensive  $\TeX$  Archive Network (Wszechstronne Archiwum Sieci  $\TeX$ , w skrócie CTAN). Strona główna znajduje się pod adresem <http://www.ctan.org>.

W niniejszej książce znajdziesz wiele odniesień do CTANa, zwłaszcza w postaci odsyłaczy do oprogramowania i dokumentów które możesz chcieć pobrać. Zamiast używać pełnych adresów stron, napisane będzie po prostu CTAN: po którym nastąpi lokalizacja w drzewie CTAN, którą musisz podążyć.

Jeśli chcesz uruchomić  $\LaTeX$ a na własnym komputerze, zobacz na informację dostępne w [CTAN://systems](#).

Jeżeli masz pomysł, co należałoby dodać, usunąć lub zmienić w tym dokumencie – napisz. Jestem szczególnie zainteresowany opiniami początkujących użytkowników  $\LaTeX$ a o tym, które fragmenty są łatwe w zrozumieniu, a które mogłyby być przedstawione lepiej.



Tobias Oetiker <[tobi@oetiker.ch](mailto:tobi@oetiker.ch)>

OETIKER+PARTNER AG  
Aarweg 15  
4600 Olten  
Switzerland

Aktualna wersja tego dokumentu znajduje się w  
<CTAN://info/lshort>



# Spis treści

<b>Podziękowania</b>	<b>iii</b>
<b>Przedmowa</b>	<b>vii</b>
<b>1 Podstawy, które warto znać</b>	<b>1</b>
1.1 Nieco historii . . . . .	1
1.1.1 $\text{\TeX}$ . . . . .	1
1.1.2 $\text{\LaTeX}$ . . . . .	2
1.2 Podstawy . . . . .	2
1.2.1 Autor, redaktor i zecer . . . . .	2
1.2.2 Układ graficzny . . . . .	3
1.2.3 Zalety i wady . . . . .	3
1.3 Pliki źródłowe $\text{\LaTeX}$ a . . . . .	5
1.3.1 Odstępy . . . . .	5
1.3.2 Znaki specjalne . . . . .	5
1.3.3 Polecenia $\text{\LaTeX}$ a . . . . .	6
1.3.4 Komentarze . . . . .	7
1.4 Struktura pliku źródłowego . . . . .	8
1.5 Typowa sesja pracy z $\text{\LaTeX}$ em . . . . .	9
1.6 Układ graficzny dokumentu . . . . .	10
1.6.1 Klasy dokumentów . . . . .	10
1.6.2 Pakiety . . . . .	10
1.6.3 Style strony . . . . .	14
1.7 Pliki z którymi możesz się zetknąć . . . . .	14
1.8 Duże projekty . . . . .	16
<b>2 Składanie tekstu</b>	<b>19</b>
2.1 Struktura tekstu i języka . . . . .	19
2.2 Łamanie linii i łamanie stron . . . . .	21

2.2.1	Wyjustowane akapity . . . . .	21
2.2.2	Przenoszenie wyrazów . . . . .	23
2.3	Predefiniowane napisy . . . . .	24
2.4	Znaki specjalne i symbole . . . . .	24
2.4.1	Cudzysłów . . . . .	24
2.4.2	Dashes and Hyphens . . . . .	25
2.4.3	Pauzy . . . . .	25
2.4.4	Tylda ( $\sim$ ) . . . . .	25
2.4.5	Ukośnik ( $/$ ) . . . . .	25
2.4.6	Oznaczenie stopni ( $\circ$ ) . . . . .	25
2.4.7	Symbol waluty euro ( $\text{€}$ ) . . . . .	26
2.4.8	Wielokropek ( $\dots$ ) . . . . .	27
2.4.9	Ligatury . . . . .	27
2.4.10	Akcenty i znaki specjalne . . . . .	27
2.5	Wsparcie dla innych języków . . . . .	28
2.5.1	Polyglossia w użyciu . . . . .	29
2.6	Odstępy między wyrazami . . . . .	32
2.7	Tytuły, rozdziały i sekcje . . . . .	33
2.8	Odsyłacze . . . . .	35
2.9	Przypisy . . . . .	36
2.10	Wyróżnienia . . . . .	37
2.11	Otoczenia (Środowiska) . . . . .	37
2.11.1	Itemize, enumerate i description . . . . .	38
2.11.2	Flushleft, flushright i center . . . . .	38
2.11.3	Quote, quotation i verse . . . . .	39
2.11.4	Streszczenie . . . . .	40
2.11.5	Pisanie dosłowne . . . . .	40
2.11.6	Tabular . . . . .	41
2.12	Wstawianie grafik i obrazów . . . . .	44
2.13	Wstawki nieszytywne . . . . .	46
<b>3</b>	<b>Skład wyrażeń matematycznych</b> . . . . .	<b>51</b>
3.1	Pakiety $\mathcal{A}\mathcal{M}\mathcal{S}\text{-}\mathcal{L}\mathcal{A}\mathcal{T}\mathcal{E}\mathcal{X}$ . . . . .	51
3.2	Pojedyncze równania . . . . .	52
3.2.1	Tryb matematyczny . . . . .	54
3.3	Części składowe wyrażeń matematycznych . . . . .	55
3.4	Pojedyncze równania które są zbyt długie: <code>multline</code> . . . . .	61
3.5	Wiele równań . . . . .	62
3.5.1	Problemy z tradycyjnymi poleceniami . . . . .	62
3.5.2	Otoczenie <code>IEEEeqnarray</code> . . . . .	64

3.5.3	Typowe użycie	65
3.6	Tablice i macierze	68
3.7	Odstępy w trybie matematycznym	69
3.7.1	Fantomy	70
3.8	Zabawa z fontami matematycznymi	71
3.8.1	Pogrubione symbole	72
3.9	Twierdzenia, lematy, itp.	72
3.9.1	Dowody i symbol końca dowodu	74
3.10	Zestawienie symboli matematycznych	77
<b>4</b>	<b>Specjalności</b>	<b>87</b>
4.1	Bibliografia	87
4.2	Indeksy	88
4.3	Nagłówki i stopki	90
4.4	Pakiet verbatim	92
4.5	Instalacja dodatkowych pakietów	92
4.6	L <sup>A</sup> T <sub>E</sub> X i PDF	94
4.6.1	Łącza hipertekstowe	94
4.6.2	Problemy z łączeniami	97
4.6.3	Problemy z zakładkami	98
4.7	Praca z X <sub>Y</sub> L <sup>A</sup> T <sub>E</sub> X i PDF	98
4.7.1	Fonty	99
4.7.2	Kompatybilność pomiędzy X <sub>Y</sub> L <sup>A</sup> T <sub>E</sub> X, a pdfL <sup>A</sup> T <sub>E</sub> X	101
4.8	Tworzenie prezentacji	101
<b>5</b>	<b>Tworzenie grafiki matematycznej</b>	<b>105</b>
5.1	Przegląd	105
5.2	Otoczenie picture	106
5.2.1	Podstawowe polecenia	106
5.2.2	Odcinki	108
5.2.3	Strzałki	109
5.2.4	Okręgi	110
5.2.5	Tekst i wzory	111
5.2.6	\multiput i \linethickness	112
5.2.7	Owale	113
5.2.8	Wielokrotne użycie pudełek z rysunkami	114
5.2.9	Krzywe Béziera drugiego stopnia	116
5.2.10	Krzywe łańcuchowe	117
5.2.11	Prędkość w Szczególnej Teorii Względności	118
5.3	Pakiety PGF i TikZ	118

<b>6</b>	<b>Adaptowanie L<sup>A</sup>T<sub>E</sub>Xa</b>	<b>123</b>
6.1	Definiowane instrukcje i otoczeń . . . . .	123
6.1.1	Nowe polecenia . . . . .	124
6.1.2	Nowe otoczenia . . . . .	125
6.1.3	Nadmiarowe odstępy . . . . .	126
6.1.4	Wywoływanie L <sup>A</sup> T <sub>E</sub> Xa z linii poleceń . . . . .	127
6.1.5	Własne pakiety . . . . .	127
6.2	Fonty i rozmiary . . . . .	128
6.2.1	Polecenia do zmiany fontu . . . . .	128
6.2.2	Uważaj! Tu mogą być promile . . . . .	131
6.2.3	Rada . . . . .	131
6.3	Odstępy . . . . .	132
6.3.1	Interlinia . . . . .	132
6.3.2	Formatowanie akapitów . . . . .	132
6.3.3	Odstępy poziome . . . . .	133
6.3.4	Odstępy pionowe . . . . .	134
6.4	Układ graficzny strony . . . . .	135
6.5	Więcej o odległościach . . . . .	138
6.6	Pudełka . . . . .	138
6.7	Kreski . . . . .	141
<b>A</b>	<b>Instalacja L<sup>A</sup>T<sub>E</sub>Xa</b>	<b>143</b>
A.1	Co zainstalować . . . . .	143
A.2	Edytor międzyplatformowy . . . . .	144
A.3	T <sub>E</sub> X na macOS . . . . .	144
A.3.1	Dystrybucje T <sub>E</sub> Xa . . . . .	144
A.3.2	Edytor T <sub>E</sub> X dla macOS . . . . .	144
A.3.3	Poczęstuj się PDFView . . . . .	144
A.4	T <sub>E</sub> X na Windows . . . . .	145
A.4.1	Dystrybucje T <sub>E</sub> Xa . . . . .	145
A.4.2	Edytor L <sup>A</sup> T <sub>E</sub> Xa . . . . .	145
A.4.3	Podgląd dokumentu . . . . .	145
A.4.4	Praca z grafiką . . . . .	145
A.5	T <sub>E</sub> X na Linuxie . . . . .	145
<b>B</b>	<b>GNU General Public License, version 2</b>	<b>147</b>
	<b>Bibliografia</b>	<b>157</b>
	<b>Indeks</b>	<b>159</b>

# Spis rysunków

1.1	Minimalny dokument $\LaTeX$ owy. . . . .	8
1.2	Przykład artykułu do czasopisma . . . . .	9
2.1	Preambuła wszystko w jednym . . . . .	29
2.2	Przykład kodu wstawiającego <code>test.png</code> w dokument. . . . .	46
4.1	Przykładowe użycie <code>fancyhdr</code> . . . . .	91
4.2	Prosty kod dla klasy <code>beamer</code> . . . . .	102
6.1	Przykładowy pakiet. . . . .	127
6.2	Argumenty układu dla tej książki. . . . .	136





# Spis tabel

1.1	Klasy dokumentów. . . . .	11
1.2	Opcje klas dokumentów. . . . .	12
1.3	Wybrane pakiety z podstawowej dystrybucji $\text{\LaTeX}$ a . . . . .	13
1.4	Predefiniowane style stron w $\text{\LaTeX}$ u. . . . .	14
2.1	Torba pełna symboli euro . . . . .	26
2.2	Akcenty i znaki specjalne. . . . .	28
2.3	Nazwy kluczy w pakiecie <code>graphicx</code> . . . . .	45
2.4	Przyzwolenia elementów niesztynych. . . . .	47
3.1	Akcenty w trybie matematycznym. . . . .	77
3.2	Litery greckie. . . . .	77
3.3	Relacje binarne. . . . .	78
3.4	Operatory binarne. . . . .	78
3.5	DUŻE operatory. . . . .	79
3.6	Strzałki. . . . .	79
3.7	Strzałki i akcenty. . . . .	79
3.8	Ograniczniki. . . . .	80
3.9	Duże ograniczniki. . . . .	80
3.10	Różne symbole. . . . .	80
3.11	Symbole niematematyczne. . . . .	81
3.12	Ograniczniki w $\mathcal{AMS}$ . . . . .	82
3.13	Litery greckie i hebrajskie w $\mathcal{AMS}$ . . . . .	82
3.14	Alfabety matematyczne. . . . .	82
3.15	Operatory binarne w $\mathcal{AMS}$ . . . . .	82
3.16	Relacje binarne w $\mathcal{AMS}$ . . . . .	83
3.17	Strzałki w $\mathcal{AMS}$ . . . . .	84
3.18	Zanegowane relacje binarne i strzałki w $\mathcal{AMS}$ . . . . .	85
3.19	Różne symbole w $\mathcal{AMS}$ . . . . .	85

4.1	Przykłady użycia wpisu do indeksu. . . . .	89
6.1	Fonty. . . . .	129
6.2	Rozmiary fontów. . . . .	129
6.3	Absolutny rozmiar fontów w klasach standardowych. . . . .	129
6.4	Fonty matematyczne. . . . .	130
6.5	Jednostki $\TeX$ a. . . . .	135

# Rozdział 1

## Podstawy, które warto znać

W pierwszej części tego rozdziału przedstawimy krótko filozofię oraz historię systemu  $\text{\LaTeX} 2_{\epsilon}$ . W części drugiej skoncentrujemy się na podstawowych elementach dokumentu  $\text{\LaTeX}$ owego. Po przeczytaniu tego rozdziału czytelnik powinien z grubsza wiedzieć, jak działa  $\text{\LaTeX}$ , co jest niezbędne do rozumienia materiału prezentowanego w następnych rozdziałach.

### 1.1 Nieco historii

#### 1.1.1 $\text{\TeX}$

$\text{\TeX}$  jest programem komputerowym stworzonym przez Donalda E. Knutha [2]. Jest przeznaczony do składu tekstów oraz wzorów matematycznych. Knuth rozpoczął pracę nad  $\text{\TeX}$ em w 1977 roku, aby zbadać potencjał składu cyfrowego, stosowanego wówczas na coraz szerszą skalę w poligrafii. Miał przede wszystkim nadzieję, że uda się odwrócić tendencję do pogarszania się jakości typograficznej, co uwidaczniało się w jego własnych książkach i artykułach. W używanej obecnie postaci  $\text{\TeX}$  został udostępniony w roku 1982, a niewielkie rozszerzenie, dotyczące ośmiobitowego kodowania znaków oraz wielu języków, pojawiło się w roku 1989.  $\text{\TeX}$  ma renomę programu nadzwyczaj stabilnego, pracującego na różnego rodzaju sprzęcie oraz praktycznie wolnego od błędów. Numery wersji  $\text{\TeX}$ a zbiegają do liczby  $\pi$ , a obecny wynosi 3,141 592 65.

Słowo  $\text{\TeX}$  należy wymawiać „tech”, gdzie zgłoska „ch” wymawiana jest tak jak w niemieckim słowie „Ach”<sup>1</sup> lub w szkockim „Loch”. Zgłoska „ch” pojawia się tu, ponieważ w alfabecie greckim X oznacza literę „ch” albo „chi”.

---

<sup>1</sup>W języku niemieckim istnieją tak naprawdę dwie wymowy dla „ch” i można rozważyć, czy

$\text{T}_{\text{E}}\text{X}$  jest też pierwszą sylabą greckiego słowa technika. W tekstach ASCII  $\text{T}_{\text{E}}\text{X}$  zapisujemy jako  $\text{TeX}$ .

### 1.1.2 $\text{\LaTeX}$

$\text{\LaTeX}$  umożliwia autorom złożenie i wydrukowanie ich prac na najwyższym poziomie typograficznym przy użyciu ustalonych, profesjonalnych układów. Pierwszą wersję  $\text{\LaTeX}$ a opracował Leslie Lamport [1]. Do formatowania dokumentu  $\text{\LaTeX}$  używa programu  $\text{T}_{\text{E}}\text{X}$ . W czasach obecnych  $\text{\LaTeX}$  jest utrzymywany przez The  $\text{\LaTeX}$  Project.

Słowo  $\text{\LaTeX}$  należy wymawiać „lej-tech” albo „la-tech”. W tekstach ASCII  $\text{\LaTeX}$  zapisujemy jako  $\text{LaTeX}$ .  $\text{\LaTeX} 2_{\epsilon}$  wymawiamy „la-tech dwa i”, a w tekstach ASCII piszemy  $\text{LaTeX2e}$ .

## 1.2 Podstawy

### 1.2.1 Autor, redaktor i zecer

Aby wydać książkę, autor dostarcza maszynopis do wydawnictwa. W wydawnictwie redaktor decyduje o układzie graficznym dokumentu (szerokość szpalty, krój pisma, odstępy przed i po tytułach rozdziałów itp.). Redaktor zapisuje swoje decyzje w maszynopisie, w formie odpowiednich instrukcji, i przekazuje go zecerowi. Na podstawie maszynopisu oraz instrukcji zecer wykonuje skład.

Redaktor-człowiek próbuje odgadnąć, co autor miał na myśli, gdy zapisywał maszynopis. Wykorzystując swoje doświadczenie zawodowe, ustala, które miejsca w maszynopisie oznaczają tytuły rozdziałów, podrozdziałów, cytaty, przykłady, wzory matematyczne itd.

$\text{\LaTeX}$  gra rolę redaktora i wykorzystuje  $\text{T}_{\text{E}}\text{X}$ a jako swojego zecera. Jednakże  $\text{\LaTeX}$  jest „zaledwie” programem komputerowym i dlatego potrzebuje dodatkowej pomocy autora, który powinien dostarczyć niezbędnych do składu informacji o strukturze logicznej dokumentu. Informacje te autor zapisuje w pliku źródłowym dokumentu jako „polecenia  $\text{\LaTeX}$ a”.

---

miękkie „ch” ze słowa „Pech” nie byłoby bardziej odpowiednie. Zapytany o to Knuth napisał na niemieckiej Wikipedii: *Nie złościsz się gdy ludzie wymawiają  $\text{T}_{\text{E}}\text{X}$  w swój ulubiony sposób ... a w Niemczech wielu używa miękkie ch, ponieważ X następuje po samogłosce e, a nie twardsze ch następujące po samogłosce a. W Rosji „tex” jest bardzo pospolitym słowem wymawianym jako „tyekh”. Osobiście uważam, że najbardziej poprawną wymowę można usłyszeć w Grecji, gdzie występuje bardziej szorstkie ch z ach oraz Loch.*

Jest to istotnie różne od podejścia WYSIWYG<sup>2</sup> podjętego przez nowoczesne procesory tekstu, takie jak *MS Word* czy *LibreOffice*. W takich aplikacjach autorzy ustalają układ interaktywnie podczas wprowadzania tekstu do komputera. Na ekranie widzą oni jak będzie wyglądała ostateczna praca gdy zostanie wydrukowana.

Używając  $\LaTeX$ a, nie można na ogół oglądać dokumentu w jego ostatecznej postaci podczas wprowadzania tekstu. Można natomiast obejrzeć dokument na ekranie po przetworzeniu go  $\LaTeX$ em. Po dokonaniu korekty dokument taki można już wysłać do drukowania.

### 1.2.2 Układ graficzny

Projektowanie książek jest sztuką. Niedoświadczeni autorzy często popełniają poważne błędy, zakładając, że zaprojektowanie układu graficznego książki jest jedynie kwestią estetyki (jeżeli dokument ładnie wygląda, to jest dobrze złożony). Jednak jako że dokumenty są przeznaczone do czytania, a nie do wieszania na ścianie w galerii sztuki, to o wiele większe znaczenie niż piękny wygląd ma łatwość czytania i przyswajania tekstu. Przykłady:

- rozmiar fontu oraz numerację nagłówków należy ustalić tak, by czytelnik mógł się szybko zorientować w strukturze dokumentu;
- szerokość szpalty powinna być na tyle wąska, by czytelnik nie musiał wyteżać wzroku, wystarczająco jednak duża, aby tekst elegancko wypełniał stronę.

W systemach WYSIWYG często powstają dokumenty przyjemne dla oka, ale pozbawione struktury albo wykazujące brak konsekwencji w strukturze.  $\LaTeX$  zapobiega powstawaniu takich błędów, zmuszając autora do określenia *logicznej* struktury dokumentu. Do  $\LaTeX$ a należy dobór najodpowiedniejszego dla niej układu graficznego.

### 1.2.3 Zalety i wady

Tematem często dyskutowanym, gdy użytkownicy programów typu WYSIWYG spotykają użytkowników  $\LaTeX$ a, są „zalety  $\LaTeX$ a w porównaniu ze zwykłym procesorem tekstu” albo na odwrót. Najlepiej podczas takich dyskusji siedzieć cicho, gdyż często wymykają się one spod kontroli. Czasami jednak nie ma ucieczki...

---

<sup>2</sup>ang. *What you see is what you get* – dostaniesz to, co widzisz.

Na wszelki wypadek trochę amunicji. Oto najważniejsze zalety  $\LaTeX$ a w porównaniu ze zwykłymi procesorami tekstu:

- Profesjonalnie przygotowane układy, dzięki którym dokumenty wyglądają „jak z drukarni”.
- Wprowadzanie formuł matematycznych jest wspierane i możliwe w wygodny sposób.
- Do rozpoczęcia pracy wystarczy poznać zaledwie kilkanaście łatwych do zrozumienia instrukcji, określających strukturę logiczną dokumentu. Niemal nigdy nie trzeba zaprzętać sobie głowy formatowaniem dokumentu.
- Nawet skomplikowane elementy, takie jak: przypisy, odnośniki, spisy treści oraz spisy bibliograficzne mogą zostać łatwo wygenerowane.
- Wolne i darmowe pakiety znacząco poszerzające typograficzne możliwości  $\LaTeX$ a. Przykładowo istnieją pakiety umożliwiające wstawianie do dokumentów grafiki w formacie POSTSCRIPT czy też przygotowanie spisów bibliograficznych według ściśle określonych reguł. Opis wielu z tych pakietów można znaleźć w podręczniku *The  $\LaTeX$  Companion* [3].
- $\LaTeX$  zachęca autorów do tworzenia dokumentów o dobrze określonej strukturze.
- $\TeX$ , program formatujący używany przez  $\LaTeX 2_{\epsilon}$ , jest dalece przenośny, wolny oraz bezpłatny. Dzięki temu można działać na praktycznie dowolnej platformie systemowo-sprzętowej.

$\LaTeX$  ma także pewne wady, chociaż ciężko mi znaleźć jakąkolwiek istotną. Jestem jednak pewien, że inne osoby wskażą ci ich setki ;-)

- $\LaTeX$  nie działa u tych, którzy zapredali swoje dusze...
- Chociaż przez zmianę niektórych parametrów można dostosowywać predefiniowane układy graficzne do własnych potrzeb, to jednak zaprojektowanie całkowicie nowego układu jest trudne i czasochłonne<sup>3</sup>.

---

<sup>3</sup>Plotki mówią, że jest to jeden z ważniejszych problemów, nad jakim pracują twórcy systemu  $\LaTeX 3$ .

- Trudno jest tworzyć dokumenty o nieokreślonej, bałaganiarskiej strukturze.
- Twój chomik może, pomimo obiecujących pierwszych kroków, nie być w stanie w pełni pojąć koncepcji znakowania logicznego.

## 1.3 Pliki źródłowe L<sup>A</sup>T<sub>E</sub>Xa

Plik źródłowy L<sup>A</sup>T<sub>E</sub>Xa to zwykły plik tekstowy. Na systemach Unix/Linux pliki tekstowe występują bardzo często. Aby stworzyć taki plik na Windowsie można wykorzystać Notatnik. Pliki takie zawierają treść dokumentu oraz instrukcje dla L<sup>A</sup>T<sub>E</sub>Xa określające, jak tekst ma zostać złożony. Jeśli pracujesz w L<sup>A</sup>T<sub>E</sub>Xowym IDE to będzie ono zawierało program do tworzenia plików źródłowych L<sup>A</sup>T<sub>E</sub>Xa w formie tekstowej.

### 1.3.1 Odstępy

Znaki „niedrukowalne”, takie jak odstępy (spacje) lub znaki tabulacji, są przez L<sup>A</sup>T<sub>E</sub>Xa traktowane jednakowo – jako „odstęp”. *Kolejno* po sobie występujące znaki niedrukowalne L<sup>A</sup>T<sub>E</sub>X traktuje jak *pojedynczy* „odstęp”. Znaki niedrukowalne znajdujące się na początku wiersza są prawie zawsze ignorowane. Pojedynczy koniec linii jest traktowany jak odstęp.

Pusty wiersz pomiędzy dwoma wierszami tekstu oznacza koniec akapitu. *Kolejno* występujące puste wiersze L<sup>A</sup>T<sub>E</sub>X traktuje jak *jeden*. Przykładem może być poniższy tekst. Po prawej stronie (w ramce) przedstawiono wynik składu, a po lewej – zawartość pliku źródłowego.

Nie ma znaczenia,  
czy między słowami jest  
jedna czy       więcej spacji.

Pusty wiersz zakończył poprzedni  
akapit.

Nie ma znaczenia, czy między słowami jest  
jedna czy więcej spacji.

Pusty wiersz zakończył poprzedni akapit.

### 1.3.2 Znaki specjalne

Poniższe symbole są znakami zarezerwowanymi – w tym sensie, że albo mają dla L<sup>A</sup>T<sub>E</sub>Xa specjalne znaczenie, albo nie są dostępne we wszystkich standardowych krojach pisma. Użyte dosłownie w pliku źródłowym nie pojawią się na wydruku, lecz spowodują, że L<sup>A</sup>T<sub>E</sub>X zrobi coś niepożądanego.

# \$ % ^ & \_ { } ~ \

Jak wkrótce zobaczysz znaki te można umieścić w dokumencie pod warunkiem, że w pliku źródłowym zostaną poprzedzone ukośnikiem wstecznym.

```
\# \$ \% \^{} \& \_ \{ \} \~{}
\textbackslash
```

```
# $ % ^ & _ { } ~ \
```

Inne symbole oraz wiele innych może zostać wstawione przy użyciu specjalnych poleceń w formułach matematycznych lub jako akcenty. Znak ukośnika wstecznego `\` *nie* może zostać wstawiony przez poprzedzenie go nim samym (`\\`); ta sekwencja znaków jest używana do łamania linii. Aby go wprowadzić użyj polecenia `\textbackslash`.

### 1.3.3 Polecenia $\LaTeX$ a

Polecenia  $\LaTeX$ a mogą wystąpić w dwóch następujących odmianach:

- Zaczynają się ukośnikiem wstecznym `\` po którym następuje ich nazwa składająca się tylko z liter. Nazwy poleceń są zakończony przez spację, liczbę bądź inną „nie-literę”. Wielkość liter ma tutaj znaczenie.
- Składają się z ukośnika wstecznego i jednej „nie-literę”.
- Wiele poleceń posiada wersję z gwiazdką, uzyskiwaną przez dodanie na końcu jej nazwy asterysku (\*).

$\LaTeX$  ignoruje znaki niedrukowalne występujące po nazwie polecenia. Jeśli chcesz uzyskać odstęp po komendzie, musisz wstawić po niej pusty argument `{}` i odstęp lub użyć specjalnych poleceń wstawiających odstępy. Pusty argument `{}` zapobiega zignorowaniu przez  $\LaTeX$ a odstępu po nazwie polecenia.

Nowi użytkownicy systemu `\TeX` mogą zgubić odstęp po poleceniu. Doświadczeni `\TeX` nicy systemu `\TeX{}` wiedzą jak poprawnie używać odstępow.

Nowi użytkownicy systemu `\TeX` mogą zgubić odstęp po poleceniu. Doświadczeni `\TeX` nicy systemu `\TeX` wiedzą jak poprawnie używać odstępow.

Niektóre instrukcje  $\LaTeX$ owe wymagają argumentów. Podaje się je w nawiasach klamrowych `{ }` po nazwie komendy. Instrukcje mogą mieć także argumenty opcjonalne, podawane w nawiasach kwadratowych `[ ]`.



`\polecenie[argument opcjonalny]{argument}`

Poniższe przykłady wykorzystują kilka poleceń L<sup>A</sup>T<sub>E</sub>Xowych. Nie przejmuj się jeśli ich nie rozumiesz, zostaną opisane później.

Możesz na mnie `\textsl{polegać}`!

Możesz na mnie *polegać*!

Proszę, rozpocznij nową linię  
dokładnie tutaj!`\newline`  
Dziękuję.

Proszę, rozpocznij nową linię dokładnie  
tutaj!  
Dziękuję.

### 1.3.4 Komentarze

Po napotkaniu znaku % L<sup>A</sup>T<sub>E</sub>X ignoruje resztę bieżącego wiersza, znak końca wiersza oraz znaki odstępu na początku następnego.

Może być to używane do umieszczania dodatkowych notatek wewnątrz plików źródłowych, które nie zostaną pokazane w wersji wydrukowanej.

To jest % głupi  
% Lepiej: pouczający <----  
przykład: konstantyno%  
                  politańczy%  
                  kowianeczka

To jest przykład: konstantynopolitańczy-  
kowianeczka

Znak % może być również użyty do dzielenia bardzo długich linii w pliku wejściowym, gdy niedozwolone jest użycie spacji lub złamanie wiersza.

W przypadku dłuższych komentarzy można użyć otoczenia `comment` z pakietu `verbatim`. Wystarczy dodać linię `\usepackage{verbatim}` do preambuły twojego dokumentu – jak to zrobić opisane jest później – by móc korzystać z tego polecenia.

Oto następny  
`\begin{comment}`  
nieco głupi,  
lecz pomocny  
`\end{comment}`  
przykład wstawiania  
komentarzy do twojego dokumentu.

Oto następny przykład wstawiania ko-  
mentarzy do twojego dokumentu.

Zwróć uwagę, że sposób ten nie zadziała w skomplikowanych otoczeniach, takich jak np. tryb wprowadzania wzorów matematycznych.

## 1.4 Struktura pliku źródłowego

Gdy  $\text{\LaTeX}$  przetwarza plik źródłowy, oczekuje on, że posiada określoną strukturę. W związku z tym każdy plik musi rozpoczynać się od polecenia

```
\documentclass{...}
```

Instrukcja ta określa rodzaj tworzonego dokumentu. Po niej można umieścić polecenia dotyczące stylu całego dokumentu oraz dołączyć pakiety poszerzające możliwości  $\text{\LaTeX}$ a. Aby je załadować należy użyć polecenia

```
\usepackage{...}
```

Gdy wszystkie style i pakiety zostały ustawione<sup>4</sup>, rozpoczyna się ciało tekstu przy użyciu polecenia

```
\begin{document}
```

Dalej znajduje się tekst dokumentu, wzbogacony o  $\text{\LaTeX}$ owe polecenia. Na końcu dokumentu musi występować polecenie

```
\end{document}
```

które informuje  $\text{\LaTeX}$ a, że to koniec pracy. Wszystko co następuje po tym poleceniu zostanie zignorowane przez  $\text{\LaTeX}$ a.

Rysunek 1.1 pokazuje zawartość minimalnego dokumentu  $\text{\LaTeX}$ owego. Nieco bardziej skomplikowany plik źródłowy pokazany jest w rysunku 1.2.

<sup>4</sup>Obszar pomiędzy poleceniami `\documentclass` i `\begin{document}` nazywa się *preambułą*.

---

```
\documentclass{article}
\begin{document}
Małe jest piękne
\end{document}
```

---

Rysunek 1.1: Minimalny dokument  $\text{\LaTeX}$ owy.

## 1.5 Typowa sesja pracy z L<sup>A</sup>T<sub>E</sub>Xem

Na pewno nie możesz się już doczekać, aby wypróbować drobną plik źródłowy ze strony 8. Oto jak to zrobić: L<sup>A</sup>T<sub>E</sub>X sam z siebie nie posiada żadnego interfejsu graficznego lub wymyślnych przycisków do kliknięcia. To po prostu program, który przetwarza twój plik źródłowy. Niektóre instalacje L<sup>A</sup>T<sub>E</sub>Xa zawierają taki interfejs, w których można znaleźć przycisk do kompilowania pliku źródłowego. Na innych jedynym sposobem będzie wpisanie poleceń. Zwróć uwagę, że poniższe instrukcje zakładają, że działająca instalacja L<sup>A</sup>T<sub>E</sub>Xa znajduje się już na twoim komputerze<sup>5</sup>. Jeśli tak nie jest to możesz chcieć wprawdzie obejrzeć dodatek A na stronie 143.

1. Stwórz/Edytuj swój plik źródłowy L<sup>A</sup>T<sub>E</sub>X. Plik ten musi być plikiem tekstowym. Na systemach unixowych wszystkie edytory stworzą dokładnie taki. Na Windowsie należy się upewnić, by przy zapisywaniu pliku wybrać format *Plik Tekstowy*. Przy wybieraniu nazwy dla pliku upewnij się, że zawiera rozszerzenie `.tex`.

---

<sup>5</sup>Będzie tak w przypadku dobrze zadbanym systemów Unixowych, a... szanujące się osoby korzystają z Unixa, więc... ;-)

---

```
\documentclass[a4paper,11pt]{article}
\usepackage{polski}
% Zdefiniuj tytuł
\author{H.~Partl}
\title{Minimalizm}
\begin{document}
% Generuje tytuł
\maketitle
% Wstawia spis treści
\tableofcontents
\section{Jakieś Interesujące słowa}
A tutaj rozpoczyna się nas cudowny artykuł.
\section{Żegnaj świecie}
\ldots{} a tutaj się kończy.
\end{document}
```

---

Rysunek 1.2: Przykład artykułu do czasopisma. Użyte w nim polecenia zostaną objaśnione w dalszej części.

- Otwórz terminal lub okno poleceń, przejdź do katalogu w którym zapisany jest twój plik (polecenie `cd` – *change directory*) i uruchom na nim  $\text{\LaTeX}$ a. Jeśli operacja się powiedzie, w katalogu pojawi się plik `.pdf`. Może być konieczne wielokrotne uruchomienie  $\text{\LaTeX}$ a by poprawnie wygenerować spis treści i wszystkie odnośniki. Jeśli w twoim kodzie jest błąd  $\text{\LaTeX}$  poinformuje cię o tym i przestanie przetwarzać plik. Naciśnij `ctrl-D` by powrócić do linii poleceń.

```
xelatex foo.tex
```

## 1.6 Układ graficzny dokumentu

### 1.6.1 Klasy dokumentów

Na samym początku przetwarzania pliku źródłowego  $\text{\LaTeX}$  musi się dowiedzieć, jakiego typu dokument autor chce uzyskać. Określone jest to w instrukcji `\documentclass`:

```
\documentclass[opcje]{klasa}
```

gdzie *klasa* oznacza typ dokumentu, który ma zostać utworzony. W tabeli 1.1 zestawiono klasy dokumentów opisane w niniejszym wprowadzeniu. Dystrybucja  $\text{\LaTeX} 2_{\epsilon}$  dostarcza również inne klasy, między innymi do pisania listów czy tworzenia slajdów. Argument *opcje* dostosowuje zachowanie klasy dokumentu. Opcje muszą być oddzielone przecinkami. Najczęstsze opcje standardowych klas dokumentów są zestawione w tabeli 1.2.

Przykład: Plik źródłowy może się rozpoczynać od następującej instrukcji

```
\documentclass[11pt,twoside,a4paper]{article}
```

która instruuje  $\text{\LaTeX}$ a by złożył dokument w klasie *article*, fontem rozmiaru *11 punktów*, przygotowany do wydruku po *dwóch* stronach na kartkach papieru formatu *A4*.

### 1.6.2 Pakiety

Podczas pisania swoich dokumentów, może się okazać, że w czasami potrzebujesz zrobić coś, czego podstawowy  $\text{\LaTeX}$  nie potrafi. Jeśli chcesz włączyć grafikę, kolorowy tekst lub kod źródłowy do twojego dokumentu będziesz

musiał rozszerzyć możliwości  $\LaTeX$ a. Takie rozszerzenia nazywamy pakietami. Dołącza się je poleceniem:

```
\usepackage[opcje]{pakiet}
```

gdzie *pakiet* oznacza nazwę pakietu, a *opcje* – listę rozdzielonych przecinkami opcji, które konfiguruja zachowanie pakietu. Polecenie `\usepackage` umieszcza się zawsze w preambule. Jeśli nie pamiętasz co to zajrzyj do sekcji 1.4.

Część pakietów znajduje się w podstawowej dystrybucji  $\LaTeX 2_{\epsilon}$  (zobacz tabela 1.3), inne są rozpowszechniane oddzielnie. Jeśli  $\LaTeX$ a używamy w systemie, którym zarządza (dobry) administrator, to informacja o dostępnych pakietach powinna się znajdować w *Local Guide (Przewodnik lokalny)* [4]. Podstawowym źródłem informacji o pakietach  $\LaTeX$ a jest *The  $\LaTeX$  Companion* [3]. Zawiera on opis setek pakietów, a także informuje, jak można pisać własne rozszerzenia  $\LaTeX$ a.

Nowoczesne dystrybucje  $\TeX$ a zawierają domyślnie dużą liczbę pakietów. Jeśli pracujesz na systemie unixowym, możesz użyć komendy `texdoc` by zobaczyć ich dokumentację.

Tabela 1.1: Klasy dokumentów.

---

`article` dla artykułów w czasopiśmie naukowych, prezentacji, krótkich sprawozdań, dokumentacji programów, zaproszeń, ...

`proc` klasa dla zbiorów artykułów z konferencji (proceedings) opartych na klasie `article`.

`minimal` jest tak mała jak tylko się da. Ustawia tylko rozmiar strony oraz podstawowy font. Wykorzystywana głównie do wyłapywania błędów.

`report` dla dłuższych sprawozdań zawierających kilka rozdziałów, krótkich książek, prac dyplomowych, ...

`book` dla prawdziwych książek

`slides` dla prezentacji. Klasa ta używa dużych liter bezszeryfowych. Możesz również rozważyć wybór nowszej klasy Beamer.

---

Tabela 1.2: Opcje klas dokumentów.

---

10pt, 11pt, 12pt	Ustawia rozmiar fontu dla tekstu zasadniczego dokumentu. Domyślną wartością jest 10 punktów.
a4paper, letterpaper, ...	Definiuje format arkusza. Domyślną wartością jest letterpaper. Inne dopuszczalne wartości to: a5paper, b5paper, executivepaper i legalpaper.
fleqn	Składanie wyeksponowanych wzorów matematycznych od lewego marginesu zamiast domyślnego centrowania.
leqno	Umieszczanie numerów wzorów matematycznych na lewym marginesie zamiast domyślnie na prawym.
titlepage, notitlepage	Ustala czy nowa strona powinna zostać wstawiona po tytule dokumentu (titlepage) czy nie (notitlepage). Domyślnie klasa article nie rozpoczyna nowej strony, ale report i book już tak.
onecolumn, twocolumn	Ustawia czy tekst powinien być składany w jednej (onecolumn) czy dwóch kolumnach (twocolumn).
twoside, oneside	Ustawia czy wyjście powinno być złożone do druku jednostronnego (oneside) czy dwustronnego (twoside). Klasy article i report są domyślnie jednostronne, a klasa book jest domyślnie dwustronna. Zwróć uwagę, że ta opcja zmienia tylko styl twojego dokumentu. Opcja ta <i>nie</i> poinstruuje twojej drukarki, że chcesz zrobić dwustronny wydruk.
landscape	Zmienia układ strony na poziomy.
openright, openany	Wybranie pierwszej opcji powoduje, że tytuły rozdziałów będą umieszczane tylko na stronach nieparzystych (pusta kartka zostanie dodana w razie potrzeby). W klasie article opcja nie ma znaczenia, gdyż w tej klasie nie jest zdefiniowane pojęcie rozdziału. W klasie report domyślną wartością jest openany, a w klasie book – openright.

---

Tabela 1.3: Wybrane pakiety z podstawowej dystrybucji  $\text{\LaTeX}$ a

---

<code>doc</code>	Służy do drukowania dokumentacji pakietów oraz innych części składowych $\text{\LaTeX}$ a. Opis znajduje się w pliku <code>doc.dtx</code> <sup>a</sup> oraz w <i>The <math>\text{\LaTeX}</math> Companion</i> [3].
<code>exscale</code>	Umożliwia skalowanie fontów matematycznych, tak by optycznie były zgodne z otaczającym tekstem, np. w tytułach rozdziałów. Opis w <code>ltxscale.dtx</code> .
<code>fontenc</code>	Definiuje układ znaków, którego ma używać $\text{\LaTeX}$ . Opis w <code>loutenc.dtx</code> .
<code>ifthen</code>	Umożliwia korzystanie z poleceń typu „jeśli... to zrób... w przeciwnym razie...” Opis w <code>ifthen.dtx</code> i [3].
<code>latexsym</code>	Udostępnia $\text{\LaTeX}$ owy font symboliczny. Opisany w <code>latexsym.dtx</code> oraz w <i>The <math>\text{\LaTeX}</math> Companion</i> [3].
<code>makeidx</code>	Udostępnia polecenia do przygotowywania skorowidzów. Opis w punkcie 4.2 i [3].
<code>syntonly</code>	Powoduje, że dokument jest przetwarzany bez składania go.
<code>inputenc</code>	Definiuje układ znaków w pliku źródłowym, jak: ASCII, ISO Latin-1, ISO Latin-2, 437/850 IBM, Apple Macintosh, Next, ANSI-Windows albo układ zdefiniowany przez użytkownika. Opis w <code>inputenc.dtx</code> .

---

<sup>a</sup>Plik ten powinien być zainstalowany w twoim systemie. Aby otrzymać z niego plik DVI, wystarczy w katalogu z prawem do zapisu napisać `xelatex doc.dtx`. To samo stosuje się do innych pakietów z tej tabeli.

### 1.6.3 Style strony

$\LaTeX$  wspiera trzy predefiniowane kombinacje pagina górna/pagina dolna (nagłówek/stopka) – tak zwane style stron. Argument *styl* polecenia

```
\pagestyle{styl}
```

ustala którego z nich chcesz użyć. Tabela 1.4 zestawia predefiniowane style stron.

Tabela 1.4: Predefiniowane style stron w  $\LaTeX$ u.

---

**plain** pagina górna jest pusta, a pagina dolna zawiera wycentrowany numer strony. Ten styl jest domyślny;

**headings** pagina górna zawiera numer strony oraz tytuł, pagina dolna jest pusta (ten style jest użyty w niniejszym dokumencie);

**empty** pagina górna i dolna są puste.

---

Możliwa jest także zmiana stylu bieżącej strony przy użyciu polecenia

```
\thispagestyle{style}
```

Instrukcje stworzenia własnych stylów można znaleźć w *The  $\LaTeX$  Companion* [3] oraz w sekcji 4.3 na stronie 90.

## 1.7 Pliki z którymi możesz się zetknąć

Pracując z  $\LaTeX$ em, szybko zauważysz, że na dysku pojawia się mnóstwo plików o różnych rozszerzeniach, o których nie masz żadnego pojęcia. W poniższym wykazie objaśniono rozmaite typy plików, z którymi możesz się zetknąć podczas pracy z  $\TeX$ em. Wykaz ten nie pretenduje do kompletnego, dlatego napisz do nas, gdy napotkasz jakieś nowe rozszerzenie, które uznasz za warte opisanie.

**.tex** Plik źródłowy z dokumentem w notacji  $\LaTeX$ a bądź zwykłego  $\TeX$ a. Można go kompilować programem `xelatex`.



- .sty** Pakiet makr  $\LaTeX$ owych. Plik tego typu można dołączać do dokumentu  $\LaTeX$ owego, używając do tego celu instrukcji `\usepackage`.
- .dtx** Documented  $\TeX$ . Udokumentowany  $\TeX$ . Jest to podstawowy format, w jakim dystrybuowane są style  $\LaTeX$ a. Skutkiem kompilacji pliku tego typu jest broszurka z udokumentowanymi makrami.
- .ins** Instalator dla plików `.dtx`. Pobierając pakiet  $\LaTeX$ owy, otrzymasz na ogół pliki `.dtx` i `.ins`. Uruchomienie  $\LaTeX$ a na pliku `.ins` powoduje rozpakowanie pliku `.dtx`.
- .cls** Plik z klasą  $\LaTeX$ a definiującą wygląd składanych w  $\LaTeX$ u dokumentów. Właśnie do tych plików odnosi się występująca na początku dokumentu instrukcja `\documentclass`.
- .fd** Definicja niektórych właściwości fontów  $\LaTeX$ a.

W wyniku kompilacji dokumentu  $\LaTeX$ owego powstają następujące pliki:

- .dvi** *Device Independent File* (plik niezależny od urządzenia), będący wynikiem kompilacji pliku źródłowego przez „tradycyjnego”  $\LaTeX$ a. Możesz mu się przyjrzeć przy użyciu odpowiedniego programu bądź wysłać go do drukarki przy użyciu `dvips` lub podobnej aplikacji. Jeśli używasz `pdf $\LaTeX$`  to nie powinieneś zobaczyć takich plików.
- .log** Zawiera szczegółowy raport z tego, co się wydarzyło podczas kompilacji.
- .toc** Zawiera nagłówki rozdziałów i punktów dokumentu. Jest on czytany przez  $\LaTeX$ a w następnym przebiegu kompilacji, w celu wygenerowania spisu treści.
- .lof** Podobny do pliku `.toc`, z tym że zawiera wykaz ilustracji.
- .lot** Tak samo, lecz dotyczy wykazu tabel.
- .aux** Inny plik pomocniczy, przenoszący informację z jednego przebiegu kompilacji do następnego. Jest używany między innymi do magazynowania informacji związanej z odsyłaczami występującymi w dokumencie.
- .idx** Jeśli dokument zawiera indeks (skorowidz), to w tym pliku  $\LaTeX$  zapisze wszystkie jego hasła. Do przetworzenia tego pliku służy program `makeindex`. Więcej o tworzeniu indeksów przeczytasz w punkcie 4.2 na stronie 88.

- .ind Przetworzony plik .idx, gotowy do włączenia do dokumentu w następnym cyklu kompilacji.
- .ilg Sprawozdanie z tego, co zrobił program makeindex.

## 1.8 Duże projekty

Pracując nad dużym dokumentem, wygodnie jest podzielić plik źródłowy na mniejsze części. W  $\text{\LaTeX}$ u mamy dwie instrukcje ułatwiające pracę z tak podzielonymi dokumentami. Pierwszą z nich jest:

```
\include{nazwa pliku}
```

Włącza ona do dokumentu zawartość pliku o nazwie *nazwa pliku.tex*. Zwróć uwagę, że  $\text{\LaTeX}$  rozpoczyna nową stronę przed przetworzeniem instrukcji z *nazwa pliku.tex*.

Drugie polecenie może być użyte w preambule. Pozwala ono by  $\text{\LaTeX}$  przetworzył tylko niektóre z plików dołączonych poleceniem `\included`.

```
\includeonly{nazwa pliku,nazwa pliku,...}
```

Gdy ta komenda jest obecna w preambule dokumentu, spośród instrukcji `\include` zostaną wykonane tylko te, które dotyczą plików wymienionych w argumencie `\includeonly`.

Polecenie `\include` rozpoczyna skład dołączanego tekstu od nowej strony. W połączeniu z `\includeonly` w preambule, instrukcja `\include` umożliwia przetwarzanie wybranych plików bez zmiany miejsc łamania poszczególnych stron. Czasami jednak rozpoczynanie składu od nowej strony nie jest pożądane. W takim przypadku można użyć polecenia

```
\input{nazwa pliku}
```

Wstawia ono zawartość podanego pliku już bez żadnych dodatkowych efektów.

Aby szybko sprawdzić swój dokument pod kątem błędów możesz użyć pakietu `syntonly`. Sprawia on, że  $\text{\LaTeX}$  przetworzy twój dokument sprawdzając tylko poprawność składni i użyte polecenia, ale nie wyprodukuje żadnego pliku.  $\text{\LaTeX}$  działa szybciej w tym trybie, więc może ci to zaoszczędzić nieco czasu. Używanie jest bardzo proste:

```
\usepackage{syntonly}  
\syntonly
```

Gdy już zechcesz wyprodukować plik wystarczy zakomentować drugą linię (dodając przed nią znak procent).



## Rozdział 2

# Składanie tekstu

Po lekturze poprzedniego rozdziału znasz już podstawy tego jak tworzony jest  $\LaTeX$  2<sub>ε</sub>owy dokument. W tym rozdziale Twoja wiedza wzbogaci się o informacje niezbędne do tworzenia dokumentów z prawdziwego zdarzenia.

### 2.1 Struktura tekstu i języka

Hanspeter Schmid <[hanspi@schmid-werren.ch](mailto:hanspi@schmid-werren.ch)>

Głównym zadaniem słowa pisanego jest przekaz myśli, informacji lub wiedzy. Nadanie zapisowi odpowiedniej struktury pomaga czytelnikowi lepiej zrozumieć przekazywane treści. Skład typograficzny może czytelnikowi tę logiczną i semantyczną strukturę tekstu przybliżyć.

$\LaTeX$  tym się różni od innych systemów składu, że do złożenia tekstu wystarczy mu znajomość logicznej i semantycznej struktury tekstu. Postać typograficzna jest wyprowadzana na podstawie „reguł” zawartych w klasie dokumentu i plikach ze stylami.

Najważniejszą jednostką podziału tekstu w  $\LaTeX$ u (a także w typografii) jest akapit. Jest to „jednostka podziału” głównie dlatego, że według wszelkich kanonów sztuki typograficznej treść akapitu powinna być poświęcona jednej spójnej myśli lub pojęciu. Tak więc, gdy zaczyna się nowa myśl, powinien się zacząć nowy akapit. Kontynuacja dotychczasowej myśli w nowym akapicie jest błędem. Niezgodne z regułami sztuki jest też pojawienie się w tym samym akapicie całkowicie nowej myśli. W następnym punkcie omówimy instrukcje nakazujące  $\LaTeX$ owi złamanie linii bez rozpoczynania nowego akapitu, na przykład poleceniem `\\`, a także sposób rozpoczęcia nowego akapitu, na przykład przez umieszczenie w kodzie źródłowym pustej linii.

Większość ludzi całkowicie lekceważy znaczenie właściwej organizacji akapitów. Co więcej, wiele osób nawet nie zdaje sobie sprawy, czym akapit naprawdę jest, i (szczególnie w  $\text{\LaTeX}$ u) kończy akapit, nawet o tym nie wiedząc. Błąd taki łatwo popełnić zwłaszcza w tekście z równaniami. Zobaczmy, dlaczego w poniższych przykładach w jednej z takich sytuacji należy przejść do nowego akapitu, a w innej nie. (Czytelnik, który nie zna jeszcze wszystkich poleceń użytych w tych przykładach, powinien dokładnie przeczytać ten i następny rozdział, a następnie wrócić do tego punktu i przeczytać go jeszcze raz.)

```
% Przykład 1
\ldots gdy Einstein wprowadził swoje równanie
\begin{equation}
e = m \cdot c^2 \ ; \ ,
\end{equation}
które jest też najbardziej znanym, ale też
najmniej rozumianym równaniem w fizyce.

% Przykład 2
\ldots z czego wynika pierwsze prawo Kirchhoffa:
\begin{equation}
\sum_{k=1}^n I_k = 0 \ ; \ .
\end{equation}

Drugie prawo Kirchhoffa może być wyprowadzone z\ldots

% Przykład 3
\ldots co ma wiele zalet.

\begin{equation}
I_D = I_F - I_R
\end{equation}
jest rdzeniem innego modelu tranzystora. \ldots
```

Mniejszą od akapitu jednostką podziału tekstu jest zdanie. W tekstach anglojęzycznych odstęp po kropce kończącej zdanie jest większy od odstepu po kropce oznaczającej skrót. Zależnie od kontekstu  $\text{\LaTeX}$  stara się użyć krótszego lub dłuższego odstepu. W razie pomyłki z jego strony powinniśmy

mu wskazać nasze intencje. Jak to zrobić, wyjaśniamy w dalszej części tego rozdziału.

Właściwa organizacja tekstu dotyczy nawet fragmentów zdań. Wiele języków ma bardzo skomplikowaną interpunkcję, ale w większości wypadków (wliczając angielski i niemiecki) stawiając przecinek w określonym miejscu w zdaniu, nie popełnimy na ogół błędu, pamiętając o zasadzie, że przecinek oznacza krótką przerwę w wypowiedzi. Jeśli nie masz pewności gdzie postawić przecinki spróbuj przeczytać zdanie i brać wdech w każdym miejscu gdzie pojawia się przecinek. Jeśli zrobienie tego wydaje się dziwne w danym miejscu to usuń tam przecinek, a jeśli gdzieś odczuwasz potrzebę zrobienia wdechu (bądź krótkiej pauzy) wstaw tam przecinek.

Wreszcie zbiory akapity w tekście powinny mieć narzuconą wyższą hierarchię poprzez ustalenie rozdziałów, sekcji, podsekcji itd. Jednakże efekt w składaniu przykładowo `\section{Struktura tekstu i języka}` są na tyle oczywiste, że reguły ich użycia są łatwo zrozumiałe.

## 2.2 Łamanie linii i łamanie stron

### 2.2.1 Wyjustowane akapity

Książki najczęściej składa się tak, że wszystkie wiersze w akapitach są tej samej długości. Dążąc do optymalnej prezentacji akapitu,  $\text{\LaTeX}$  ustala miejsca łamania linii oraz wielkość odstępów między słowami. W razie potrzeby przenosi wyrazy, których nie jest w stanie zmieścić w wierszu. Sposób składania akapitów zależy od użytej klasy dokumentu. Najczęściej pierwszy wiersz akapitu jest wcięty, a między akapitami nie ma dodatkowych odstępów. Więcej na ten temat można przeczytać w punkcie [6.3.2](#).

W specjalnych przypadkach może być konieczne nakazanie  $\text{\LaTeX}$ owi złamania linii w danym miejscu. Polecenie

```
\ lub \newline
```

rozpoczyna nową linię bez rozpoczynania nowego akapitu, a polecenie

```
\*
```

zakazuje dodatkowo złamania strony (w miejscu złamania linii). Z kolei in-

strukcja:

```
\newpage
```

rozpoczyna nową stronę.

Polecenia

```
\linebreak[n], \nolinebreak[n], \pagebreak[n], \nopagebreak[n]
```

oznaczają, odpowiednio: nakaz złamania wiersza, zakaz złamania wiersza, nakaz złamania strony i zakaz złamania strony (w miejscu ich wystąpienia w dokumencie). Opcjonalny argument  $n$ , o dopuszczalnej wartości od 0 do 4, określa stopień tego nakazu (zakazu). Domyślna wartość 4 to bezwarunkowy zakaz lub nakaz złamania linii/strony. Wartość mniejsza od 4 pozostawia  $\LaTeX$ -owi swobodę zignorowania instrukcji, jeżeli skład otrzymany w jej rezultacie byłby bardzo kiepskiej jakości.

Polecen z grupy `break` nie należy mylić z tymi z grupy `new`. Mimo otrzymania polecenia typu `break`  $\LaTeX$  stara się wyrównać wiersz do prawego marginesu czy też wypełnić stronę do całej jej wysokości. Nietrudno zgadnąć, jakiej instrukcji należy użyć, gdy naprawdę zależy nam na rozpoczęciu nowego wiersza.

$\LaTeX$  zawsze stara się znaleźć najlepszy podział akapitu na wiersze. Kiedy nie potrafi znaleźć podziału, który spełnia jego wysokie wymagania jakościowe, pozwala wtedy niektórym wyrazom wystawać na prawy margines. Będzie on wtedy narzekał komunikatem „`overfull hbox`” podczas przetwarzania pliku źródłowego. Zdarza się tak gdy  $\LaTeX$  nie może znaleźć dobrego miejsca na podzielenie słowa<sup>1</sup>. Aby poinstruować  $\LaTeX$ a by obniżył swoje standardy możesz użyć polecenia `\sloppy`. Zapobiega ono zbyt długim wierszom poprzez zwiększenie odstępów między wyrazami – nawet jeśli ostateczny rezultat nie jest najlepszy. W takim przypadku otrzymamy ostrzeżenie „`underfull hbox`”. W większości przypadków rezultat nie wygląda wtedy najlepiej. Aby przywrócić  $\LaTeX$ a do oryginalnego zachowania należy użyć polecenia `\fussy`.

<sup>1</sup>Pomimo tego, że  $\LaTeX$  wyświetla ostrzeżenie gdy tak się dzieje, a wraz z nim problematyczny wiersz, nie zawsze jest łatwo znaleźć dokładne miejsce. Jeśli użyjesz opcji `draft` w poleceniu `\documentclass` to wiersze te będą oznaczone grubą czarną linią na prawym marginesie.



### 2.2.2 Przenoszenie wyrazów

W razie potrzeby  $\LaTeX$  przenosi (dzieli) wyrazy. Jeżeli algorytm podziału nie znajdzie poprawnych miejsc przenoszenia wyrazu, to można je wskazać poleceniem

```
\hyphenation{lista słów}
```

które sprawia, że słowa z listy można dzielić wyłącznie w miejscach oznaczonych znakiem -. Argumenty tego polecenia powinny składać się wyłącznie z liter, lub dokładniej z rzeczy które  $\LaTeX$  uważa za litery. Wskazówki dzielenia wyrazów są zapisywane dla języka aktywnego w momencie gdy polecenie jest użyte. Oznacza to, że jeśli użyjesz go w preambule to wpłynie to na język angielski. Jeśli wstawisz je po  $\begin{document}$  i korzystasz z jakiegoś pakietu do wspierania innych języków, takiego jak `polyglossia`, to zasady dzielenia będą dodane w języku aktywowanym przez ten pakiet.

Poniższy przykład pozwoli na dzielenie wyrazu „dzielenie” oraz „Dzielenie” oraz zabroni dzielenia wyrazów „FORTRAN”, „Fortran” i „fortran”. Specjalne znaki i symbole nie są dozwolone w argumentach polecenia.

Przykład:

```
\hyphenation{FORTRAN Dzie-le-nie}
```

Polecenie  $\backslash-$  wstawia nieobowiązkowy podział w wyrazie. Staje się on jednocześnie jedynym możliwym miejscem podziału w tym wyrazie. Jest ono szczególnie przydatne w słowach zawierających specjalne znaki (np. z akcentami), ponieważ  $\LaTeX$  nie łamie automatycznie takich słów.

```
Na moście tańczy kon\-stan\-%
ty\-no\-po\-li\-tań\-czy\-%
ko\-wia\-necz\-ka
```

```
Na moście tańczy konstantynopolitańczy-
kowaneczka
```

Kilka słów może być utrzymane razem przy użyciu komendy

```
\mbox{tekst}
```

Wymusza ona by jej argumenty były w jednej linii niezależnie od sytuacji.

```
Numer mojego nowego telefonu
to \mbox{0116 291 2319}.
```

```
Parametr \mbox{\emph{nazwa}} to
nazwa pliku.
```

```
Numer mojego nowego telefonu to
0116 291 2319.
```

```
Parametr nazwa to nazwa pliku.
```

Polecenie `\fbox` jest podobne do `\mbox`, z tym że dodatkowo dookoła zawartości rysuje ramkę.

## 2.3 Predefiniowane napisy

W przykładach na poprzednich stronach pojawiło się kilka prostych instrukcji  $\text{\LaTeX}$ a do składania krótkich napisów.

Polecenie	Przykład	Opis
<code>\today</code>	13 maja 2022	Bieżąca data
<code>\TeX</code>	$\text{\TeX}$	Twój ulubiony system składu
<code>\LaTeX</code>	$\text{\LaTeX}$	Sedno sprawy
<code>\LaTeXe</code>	$\text{\LaTeX} 2_{\epsilon}$	Obecne jej wcielenie

## 2.4 Znaki specjalne i symbole

### 2.4.1 Cudzysłowy

Znaku " *nie* używamy do wstawiania cudzysłowu tak jak robi się to np. na maszynie do pisania.

W publikacjach drukowanych różnie oznacza się początek i koniec cudzysłowu. W  $\text{\LaTeX}$ u używa się podwójnych znaków ``` (grawis) do otwarcia cudzysłowu i dwóch znaków `'` (apostrof) do jego zamknięcia według konwencji angielskiej. Dla wewnętrznych cudzysłowów używa się ich pojedynczo.

```
``Please press the `x' key.``
```

```
"Please press the 'x' key."
```

W języku polskim cudzysłów otwierający oznacza się dwoma przecinkami `,,`, natomiast zamykający – dwoma apostrofami `''`. Gdy zachodzi konieczność użycia cudzysłowu w tekście już objętym cudzysłowem, to stosuje się „cudzysłowy »niemieckie«”, oznaczane w pliku źródłowym znakami, odpowiednio, większości `>>` i mniejszości `<<`.

```
,,Naciśnij przycisk >>x<<.''
```

```
„Naciśnij przycisk »x«.”
```

Nie oznacza to, że znak " bezużyteczny. Jest on wykorzystywany na przykład do wstawienia umlauta. Będzie to opisane dokładniej w sekcji 2.4.9.

### 2.4.2 Dashes and Hyphens

#### 2.4.3 Pauzy

$\LaTeX$  zna cztery rodzaje „pauz” (poziomych kresek). Trzy z nich można uzyskać powtarzając znak pauzy z klawiatury. Czwarty rodzaj nie jest tak naprawdę pauzą lecz znakiem minus:

```
Akademia Górniczo-Hutnicza\\
strony 13--67\\
--- Na pewno? --- spytała.\\
$0$, $1$ i $-1$
```

```
Akademia Górniczo-Hutnicza
strony 13-67
— Na pewno? — spytała.
0, 1 i -1
```

Formalne ich nazwy to: „-” dywiz, „-” półpauza, „—” pauza i „-” znak minus.

#### 2.4.4 Tylda (~)

W adresach internetowych często występuje znak tyldy. W  $\LaTeX$ u można do jego uzyskania użyć instrukcji `\~`, ale wynik: `~` nie jest chyba tym, czego oczekujemy. Lepiej zrobić tak:

```
http://www.rich.edu/~{bush} \\
http://www.clever.edu/$\sim$demo
```

```
http://www.rich.edu/~bush
http://www.clever.edu/~demo
```

#### 2.4.5 Ukośnik (/)

Aby wpisać ukośnik między dwoma słowami można użyć tego znaku wprost, przykładowo `otwórz/zamknij`, ale sprawia to, że  $\LaTeX$  traktuje oba wyrazy jako jeden, a co za tym idzie wyłącza ich dzielenie. Może to doprowadzić do ostrzeżeń „`overflow hbox`”. Aby temu zapobiec użyj `\slash`, przykładowo `otwórz\slash zamknij`. Normalnego ukośnika używaj jeśli chcesz wyrazić ułamki bądź jednostki, np. `5 MB/s`.

#### 2.4.6 Oznaczenie stopni (°)

Poniższy przykład ilustruje, jak w czystym  $\LaTeX$ u uzyskuje się symbol stopni:

Jest  $\$-30\,^{\circ}\mathrm{C}$ .  
Niedługo zacznę nadprzewodzić.

Jest  $-30\text{ }^{\circ}\text{C}$ . Niedługo zacznę nadprzewodzić.

Pakiet `textcomp` udostępnia symbol stopni jako `\textdegree` lub w połączeniu z literą `C` także jako `\textcelsius`.

30 `\textcelsius` to  
86 `\textdegree`F.

30 °C to 86 °F.

### 2.4.7 Symbol waluty euro (€)

Pisząc dziś o pieniądzach, nie można się obejść bez symbolu euro. Znak ten występuje w wielu współczesnych fontach. Po załadowaniu pakietu `textcomp` w preambule:

```
\usepackage{textcomp}
```

użyj polecenia

```
\texteuro
```

aby go wpisać.

Jeśli używany font nie zawiera własnego symbolu euro albo nam się on nie podoba, to mamy dwie dodatkowe możliwości: Pierwszą jest pakiet `eurosym`. Udostępnia on oficjalny znak euro:

```
\usepackage[official]{eurosym}
```

Jeśli wolimy znak euro zgodny optycznie z fontem, to zastępujemy opcję `official` opcją `gen`.

Tabela 2.1: Torba pełna symboli euro

LM+textcomp	<code>\texteuro</code>	€	€	€
eurosym	<code>\euro</code>	€	€	€
[gen]eurosym	<code>\euro</code>	€	€	€

### 2.4.8 Wielokropek (...)

W typowym piśmie maszynowym przecinek oraz kropka zajmują tyle samo miejsca co każdy inny znak. W piśmie drukarskim szerokość tych znaków jest z reguły bardzo mała i dlatego, jeżeli umieścimy je obok siebie, to odstępy między nimi będą zbyt małe. Do uzyskiwania wielokropka (trzech kropek) istnieje specjalna komenda o nazwie

```
\ldots
```

Nie tak ..., lecz raczej tak:\\  
Londyn, Tokio, Warszawa, \ldots

Nie tak ..., lecz raczej tak:  
Londyn, Tokio, Warszawa, ...

### 2.4.9 Ligatury

Niektóre kombinacje liter są składane nie poprzez postawienie dwóch znaków obok siebie lecz przez wstawienie specjalnego znaku.

ff fi fl ffi... zamiast ff fi fl ffi ...

Jeśli nam w danym przykładzie to przeszkadza to tak zwane ligatury mogą być zabronione poprzez wstawienie `\mbox{}` pomiędzy dwoma rzeczonymi literami.

Jak lepiej: geografii czy  
geograf\mbox{ii}?

Jak lepiej: geografii czy geografii?

### 2.4.10 Akcenty i znaki specjalne

W  $\text{\LaTeX}$ u istnieją metody wstawiania znaków akcentowanych oraz spotykanych w różnych językach znaków specjalnych. W tabeli 2.2 zestawiono instrukcje akcentów. Użyto ich do litery „o”, ale można je również stosować do dowolnej innej litery.

W wypadku akcentów nad literami „i” oraz „j” należy znad tych liter usunąć kropkę. Służą do tego instrukcje `\i` i `\j`.

```
H\^otel, na\"i ve, \'el`eve,\\  
sm\o rrebr\o d, !`Se\~norita!,\\  
Sch\"onbrunner Schlo\ss{  
Stra\ss e
```

Hôtel, naïve, élève,  
smørrebrød, ¡Señorita!,  
Schönbrunner Schloß Straße

Tabela 2.2: Akcenty i znaki specjalne.

ò	\`o	ó	\'o	ô	\~o	õ	\~o
ō	\=o	ô	\.o	ö	\"o	ç	\c c
ö	\u o	ö	\v o	ő	\H o	q	\c o
ø	\d o	o	\b o	oo	\t oo		
œ	\oe	Œ	\OE	æ	\ae	Æ	\AE
å	\aa	Å	\AA				
ø	\o	Ø	\O	ł	\l	Ł	\L
ı	\i	ı	\j	ı	!\`	ı	?`

## 2.5 Wsparcie dla innych języków

Axel Kielhorn <[A.Kielhorn@web.de](mailto:A.Kielhorn@web.de)>

Domyślnym językiem systemu  $\LaTeX$  jest język angielski. Aby składać dokumenty w innym języku  $\LaTeX$ a należy skonfigurować w trzech miejscach:

1. Wszystkie automatycznie wstawiane napisy<sup>2</sup> muszą zostać zmienione.
2.  $\LaTeX$  musi znać odpowiednie zasady dzielenia wyrazów dla danego języka.
3. Zasady typograficzne charakterystyczne dla danego języka. Przykładowo w języku polskim spacje po kropce powinny być tej samej długości co między wyrazami (w angielskim są one nieco dłuższe).

Co więcej wprowadzanie tekstu w danym języku przy użyciu poleceń opisanych w tabeli 2.2 może być nieco nieporęczne. Aby temu zapobiec do niedawna trzeba było zagłębiać się w kodowania specyficzne dla danego języka oraz fontów. W dzisiejszych czasach nowe systemy  $\TeX$ a rozumieją już UTF-8, dzięki czemu problemy to zostały w dużej mierze zażegnane.

Pakiet `polyglossia`[5] zastępuje wiekowy już pakiet `babel`. Zajmuje się on większością konfiguracji specyficznych dla języka.

Pakiet `fontspec`[6] obsługuje ładowanie fontów w  $X_{\text{L}}\LaTeX$  i  $\text{Lua}\TeX$ . Domyślnym fontem jest Latin Modern Roman.

<sup>2</sup>Takie jak „Spis treści”, „Rozdział” itp.

### 2.5.1 Polyglossia w użyciu

Zależnie od użytego silnika  $\TeX$  nieco inne polecenia są wymagane w preambule aby poprawnie włączyć obsługę języków. Rysunek 2.1 pokazuje w jaki sposób napisać preambułę by działała ona w obu obsługiwanych silnikach.

Główną przewagą silników ze wsparciem Unicode jest bezbolesna integracja z językami które używają znaków spoza alfabetu łacińskiego takich jak rosyjski czy grecki. Dzięki nim dokument można po prostu pisać przy użyciu odpowiedniego alfabetu, nie martwiąc się o kodowanie, a  $\TeX$  poprawnie je obsłuży.

Pisanie w innych językach również jest bardzo proste; wystarczy wskazać języki w preambule. Poniższy przykład korzysta z pakietu `csquotes`, który generuje odpowiednie cudzysłowy dla danego języka. Zwróć uwagę, że powinien być załadowany *przed* wsparciem języka.

```
\usepackage[autostyle=true]{csquotes}
\setdefaultlanguage{english}
\setotherlanguage{german}
```

Aby napisać akapit po niemiecku wystarczy użyć odpowiednie środowisko:

```
Tekst po polsku.
\begin{german}
Deutscher \enquote{Text}.
\end{german}
Więcej \enquote{tekstu}.
```

```
Tekst po polsku. Deutscher „Text“. Więcej „tekstu“.
```

---

```
\usepackage{iftex}
\ifXeTeX
  \usepackage{fontspec}
\else
  \usepackage{luatextra}
\fi
\defaultfontfeatures{Ligatures=TeX}
\usepackage{polyglossia}
```

---

Rysunek 2.1: Preambuła wszystko w jednym obsługująca zarówno Lua $\LaTeX$  jak i Xe $\LaTeX$

Jeśli potrzebujesz pojedynczego słowa w obcym języku możesz skorzystać z polecenia `\textjęzyk`:

Wiedziałaś, że  
`\textgerman{Gesundheit}` jest  
 niemieckim słowem?

Wiedziałaś, że Gesundheit jest niemieckim słowem?

Może się to wydawać niepotrzebne jako, że jedyną zaletą jest poprawne dzielenie wyrazów, jednak w przypadku bardziej egzotycznych języków będzie to bardzo pomocne.

Czasami główny font użyty w dokumencie nie zawiera glifów, które są wymagane w innym języku. Przykładowo Latin Modern nie zawiera liter z cyrylicy. Najprostszym rozwiązaniem jest zdefiniowany fontu dla danego języka. Za każdym razem gdy nowy język jest zmieniany `polyglossia` sprawdzi czy dany język ma zdefiniowany font. Jeśli podoba ci się font Computer Modern, możesz wypróbować „Computer Modern Unicode” dodając następujące polecenia do preambuły.

W Lua $\text{\LaTeX}$  jest to całkiem proste

```
\setmainfont{CMU Serif}
\setsansfont{CMU Sans Serif}
\setmonofont{CMU Typewriter Text}
```

W X $\text{\LaTeX}$  jest nieco więcej pisania:

```
\setmainfont{cmun}[
  Extension=.otf,UprightFont=*rm,ItalicFont=*ti,
  BoldFont=*bx,BoldItalicFont=*bi,
]
\setsansfont{cmun}[
  Extension=.otf,UprightFont=*ss,ItalicFont=*si,
  BoldFont=*sx,BoldItalicFont=*so,
]
\setmonofont{cmun}[
  Extension=.otf,UprightFont=*btl,ItalicFont=*bto,
  BoldFont=*tb,BoldItalicFont=*tx,
]
```

Z załadowanymi fontami możesz teraz napisać:



`\textrussian{Правда}` jest rosyjską gazetą. `\textgreek{ἀλήθεια}` oznacza prawdę bądź objawienie w filozofii.

Правда jest rosyjską gazetą. ἀλήθεια oznacza prawdę bądź objawienie w filozofii.

Pakiet `xgreek`[7] pomaga pisać w pisaniu zarówno starożytnego jak i nowoczesnego (monotonicznego i politonicznego) greckiego.

### Języki piszące od prawa do lewa (RTL)

Niektóre języki są pisane od lewa do prawa, inne od prawa do lewa. `polyglossia` potrzebuje pakietu `bidi`[8]<sup>3</sup> by wspierać te drugie. Pakiet `bidi` powinien być ostatnim załadowanym pakietem, nawet po `hyperref`, które zwykle jest ostatnie. (Ponieważ `polyglossia` ładuje `bidi` oznacza to, że `polyglossia` powinna być ładowana jako ostatnia.)

Pakiet `xepersian`[9] pozwala na wsparcie dla języków perskich. Dostarcza on zlokalizowane komendy takie jak `\section` w języku perskim dzięki którym  $\LaTeX$  jest łatwiejszy do opanowania dla znających język. `xepersian` jest jedynym pakietem pozwalającym na pisanie kashida w  $X_{\text{E}}\LaTeX$ . Pakiet dla Syriac korzystający z podobnego algorytmu jest w trakcie rozwoju.

Font `IranNastaliq` dostarczany przez SCICT<sup>4</sup> jest dostępny na ich stronie <http://www.scict.ir/Portal/Home/Default.aspx>.

Pakiet `arabxetex`[10] wspiera wiele języków korzystających z alfabetu arabskiego:

- arab (Arabski)
- persian
- urdu
- sindhi
- pashto
- ottoman (turecki)
- kurdish (kurdyjski)
- kashmiri

<sup>3</sup>`bidi` nie wspiera  $\text{Lua}\TeX$ .

<sup>4</sup>Supreme Council of Information and Communication Technology

- malay (jawi)
- uighur

Zawiera on mapowanie fontu pozwalające X<sub>Y</sub>LaTeXowi na przetwarzanie wejścia napisanego przy użyciu ArabTeXowej transkrypcji ASCII.

Fonty wspierające języki arabskie są oferowane przez IRMUG<sup>5</sup> na [http://wiki.irmug.org/index.php/X\\_Series\\_2](http://wiki.irmug.org/index.php/X_Series_2).

Nie istnieje osobny pakiet dla języka hebrajskiego, ponieważ nie jest potrzebny. Wsparcie dla hebrajskiego w `polyglossia` powinno być wystarczające. Potrzebny jest jednak odpowiedni font ze wsparciem dla hebrajskiego Unicode. SBL Hebrew jest darmowym fontem do celów niekomercyjnych i dostępna na <http://www.sbl-site.org/educational/biblicalfonts.aspx>. Innym fontem dostępnym na licencji Open Font License jest Ezra SIL dostępna na [http://www.sil.org/computing/catalog/show\\_software.asp?id=76](http://www.sil.org/computing/catalog/show_software.asp?id=76).

Pamiętaj o wybraniu odpowiedniego skryptu:

```
\newfontfamily\hebrewfont [Script=Hebrew]{SBL Hebrew}
\newfontfamily\hebrewfont [Script=Hebrew]{Ezra SIL}
```

## Chiński, japoński i koreański (CJK)

Pakiet `xeCJK`[11] zajmie się wybraniem fontów oraz zasadami interpunkcji tych języków.

## 2.6 Odstępy między wyrazami

Aby wyrównać prawy margines, LaTeX wstawia między słowami odstępy różnej wielkości. Odstęp wstawiany na końcu zdania jest trochę większy, ponieważ tak składa się książki w krajach anglosaskich. LaTeX zakłada, że zdania mogą się kończyć kropką, znakiem zapytania lub wykrzyknikiem. Jeżeli bezpośrednio przed kropką znajduje się wielka litera, to LaTeX nie traktuje takiego miejsca jako końca zdania, lecz jako kropkę po skrótce.

Wyjątki od powyższych zasad trzeba wyraźnie zaznaczyć w tekście. Znak `\` poprzedzający spację oznacza odstęp normalnej wielkości. Tylda `~` również wstawia taki odstęp, z tym że LaTeXowi nie wolno na nim złamać wiersza. Umieszczenie instrukcji `\@` przed kropką jest dla LaTeX'a wskazówką, że ta kropka kończy zdanie, nawet jeśli następuje po dużej literze.

<sup>5</sup>Iranian Mac User Group

J.~Kowalska ucieszyła  
 się na jej widok zob.~Rys.~5\\  
 Podoba mi się BASIC\@. A tobie?

J. Kowalska ucieszyła się na jej widok  
 zob. Rys. 5  
 Podoba mi się BASIC. A tobie?

Dodatkowa spacja po kropkach może zostać wyłączona przy użyciu polecenia

```
\frenchspacing
```

która instruuje  $\text{\LaTeX}$ a, aby *nie* wstawiał większej spacji po kropce niż po innym wyrazie. Jeśli użyjesz polecenia `\frenchspacing`, to instrukcja `\@` nie jest potrzebna.

## 2.7 Tytuły, rozdziały i sekcje

Podzielenie dokumentu na rozdziały, sekcje, podsekcje itd. pomaga czytelnikom lepiej orientować się w tekście. Do dzielenia dokumentu na hierarchiczne części służą odpowiednie instrukcje  $\text{\LaTeX}$ owe przyjmujące nazwę sekcji jako argument. Do autora należy używanie tych poleceń w odpowiednim porządku.

W klasie `article` mamy do dyspozycji następujące instrukcje podziału hierarchicznego:

```
\section{...}
\subsection{...}
\subsubsection{...}
\paragraph{...}
\subparagraph{...}
```

Jeśli chcesz podzielić dokument na części bez naruszania numeracji sekcji, to można użyć polecenia:

```
\part{...}
```

W klasach `report` (raport) i `book` (książka) występuje dodatkowo polecenie tworzące rozdziały:

```
\chapter{...}
```

Ponieważ w klasie `article` najwyższą jednostką w hierarchii podziału jest `\section` (czyli *sekcja*), łatwo tworzy się książki (klasa `book`), w których rozdziałami są poszczególne artykuły.  $\LaTeX$  dobierze za nas odpowiednie odstępy między rozdziałami oraz wielkość i krój pisma w śródtytułach.

Dwie instrukcje podziału działają nieco inaczej niż pozostałe:

- instrukcja `\part` nie ma wpływu na numerację rozdziałów;
- instrukcja `\appendix` nie ma argumentów. Jest to deklaracja zmieniająca sposób numerowania z cyfr na litery. Dotyczy to rozdziałów w klasach `book` i `report`<sup>6</sup>.

$\LaTeX$  tworzy spis treści przy użyciu tytułów sekcji i numerów stron z poprzedniego etapu kompilacji. Instrukcja:

```
\tableofcontents
```

wstawia spis treści w miejscu jej użycia. Aby w spisie treści otrzymać poprawne numery stron, trzeba dokument przetworzyć („zlatechować”) dwukrotnie aby zadziałała poprawnie. Czasami niezbędna jest nawet trzecia kompilacja. Kolejny przebieg jest potrzebny, gdy pod koniec przetwarzania dokumentu  $\LaTeX$  ostrzeże Cię wówczas odpowiednim komunikatem.

Wymienione wyżej instrukcje podziału hierarchicznego posiadają także wersje „z gwiazdką”. Nazwa instrukcji w wersji „z gwiazdką” składa się z „normalnej” nazwy, po której występuje znak „\*”. W wyniku działania takiej instrukcji tytuł rozdziału lub punktu zostanie umieszczony w dokumencie, ale nie w spisie treści; tytuł nie zostanie też objęty numeracją. Przykładowo, wersją „z gwiazdką” instrukcji `\section{Pomoc}` jest `\section*{Pomoc}`.

Najczęściej hasła w spisie treści pokrywają się z tytułami rozdziałów czy punktów. Czasami jednak nie jest to pożądane, na przykład wówczas, gdy tekst hasła jest zbyt długi. W takich wypadkach hasło do spisu treści można podać jako opcjonalny argument instrukcji podziału hierarchicznego.

```
\chapter[Krótki i ekscytujący rozdział]{To jest
bardzo długi i wyjątkowo nudny rozdział}
```

<sup>6</sup>W klasie `article` zmienia numerowanie sekcji

L<sup>A</sup>T<sub>E</sub>X składa część tytułową dokumentu, napotkawszy instrukcję

```
\maketitle
```

Jej zawartość należy zdefiniować przy użyciu poleceń

```
\title{...}, \author{...} oraz opcjonalnie \date{...}
```

przed jej wywołaniem. Aby umieścić kilku autorów w odpowiedniej komendzie należy użyć polecenia `\and` do ich oddzielenia.

Przykłady użycia powyższych poleceń można znaleźć w rysunku 1.2 na stronie 9.

W L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub> istnieją trzy dodatkowe instrukcje dotyczące struktury dokumentu, dostępne jednak wyłącznie w klasie `book`. Są użyteczne do podzielenia książki. Zmieniają one tytuły rozdziałów, numeracje stron i działają tak jak można by się spodziewać w książce:

**\frontmatter** powinno być pierwszą poleceniem po rozpoczęciu dokumentu (`\begin{document}`). Zmieni ono numerowanie stron na liczby rzymskie, a sekcje będą nienumerowane tak jak przy użyciu wersji z gwiazdką (np. `\chapter*{Preface}`) jednak pojawią się one w spisie treści.

**\mainmatter** powinno zostać wstawione przed pierwszym rozdziałem książki. Przełącza ona sposób oznaczania numerów stron na arabski, zerując zarazem licznik stron.

**\appendix** oznacza rozpoczęcie dodatkowych materiałów w twojej książce. Po tym poleceniu rozdziały będą numerowane literami.

**\backmatter** powinno wystąpić przed ostatnimi fragmentami książki, takim jak spis literatury albo skorowidz. Nie ma żadnych efektów wizualnych w standardowych klasach dokumentu.

## 2.8 Odsyłacze

Książki, raporty i artykuły często zawierają odsyłacze do rysunków, tabel i innych fragmentów tekstu. Z odsyłaczami związane są w L<sup>A</sup>T<sub>E</sub>Xu następujące trzy instrukcje:

```
\label{etykieta}, \ref{etykieta} and \pageref{etykieta}
```

Argument *etykieta* jest ciągiem liter, cyfr lub znaków interpunkcyjnych. Na-

zwy etykiet ustala sam autor.  $\LaTeX$  zamienia `\ref{etykieta}` na numer tego rozdziału, punktu, rysunku, tabeli czy też równania matematycznego, bezpośrednio za którym umieszczona została instrukcja `\label` zawierająca identyczną *etykieta*. Instrukcja `\pageref{etykieta}` działa identycznie jak `\ref`, z tym że wstawia numer strony, na której znajduje się element oznaczony etykietą<sup>7</sup>. Podobnie jak w przypadku spisu treści wykorzystywane są dane z poprzedniego cyklu kompilacji.

Odsyłacz do tego punktu  
`\label{sec:this}` wygląda tak:  
 „patrz punkt `\ref{sec:this}` na  
 stronie `\pageref{sec:this}`.”

Odsyłacz do tego punktu wygląda tak:  
 „patrz punkt 2.8 na stronie 36.”

## 2.9 Przypisy

Przy użyciu polecenia

```
\footnote{tekst przypisu}
```

drukowana jest notatka w stopce aktualnej strony. Przypisy powinny być umieszczane bezpośrednio po zdaniu bądź wyrazie do którego się odnoszą. W krajach anglosaskich przypisy odnoszące się do całego zdania lub jego części umieszcza się natychmiast po kropce lub przecinku. W Polsce najczęściej umieszcza się je *przed* znakiem przestankowym<sup>8</sup>.

Przypisy `\footnote{To jest  
właśnie przypis.}` są często  
stosowane przez  
użytkowników `\LaTeX{}`a.

Przypisy<sup>a</sup> są często stosowane przez użyt-  
kowników  $\LaTeX$ a.

<sup>a</sup>To jest właśnie przypis.

<sup>7</sup>Warto pamiętać, że te instrukcje „nie wiedzą”, do czego tak naprawdę się odnoszą. Zadaniem instrukcji `\label` jest przechowanie związku wygenerowanej automatycznie liczby z miejscem w tekście.

<sup>8</sup>Zwróć uwagę, że przypisy odciągają uwagę od tekstu głównego. Jak wiemy wszyscy czytają przypisy, ponieważ jesteśmy ciekawskim gatunkiem. Dlaczego więc nie umieścić wszystkiego w głównym tekście dokumentu?<sup>9</sup>

<sup>9</sup>Drogowskaz niekoniecznie stawia się tam gdzie wskazuje :-).

## 2.10 Wyróżnienia

W tekstach pisanych na maszynie fragmenty, które mają zostać wyróżnione są podkreślone.

```
\underline{tekst}
```

W dokumentach drukowanych wyróżnienie fragmentu odbywa się przez złożenie go *kursywą*. Jako autora nie powinno cię to jednak obchodzić. Ważne jest to, że dane słowo powinno zostać wyróżnione. Służy do tego L<sup>A</sup>T<sub>E</sub>Xowa instrukcja

```
\emph{tekst}
```

To co faktycznie ona robi jest zależne od kontekstu:

```
\emph{Jeśli użyjesz wyróżnienia
w tekście napisanym kursywą
to zostanie on wyróżniony
przez zapisanie go
\emph{zwykłym} krojem pisma.}
```

*Jeśli użyjesz wyróżnienia w tekście napisanym kursywą to zostanie on wyróżniony przez zapisanie go zwykłym krojem pisma.*

Jeśli jest to dla Ciebie za mało więcej informacji o kontrolowaniu rozmiaru i stylu fontu znajdziesz w sekcji 6.2 na stronie 128.

## 2.11 Otoczenia (Środowiska)

```
\begin{otoczenie} tekst \end{otoczenie}
```

Gdzie *otoczenie* jest nazwą otoczenia które chcemy użyć. Otoczenia mogą być zagnieżdżone w sobie pod warunkiem, że kolejność zagnieżdżenia jest odpowiednia.

```
\begin{aaa}... \begin{bbb}... \end{bbb}... \end{aaa}
```

W kolejnych sekcjach przedstawiamy częściej używane otoczenia.

### 2.11.1 Itemize, enumerate i description

Otoczenie `itemize` służy do tworzenia list punktowanych, `enumerate` do list numerowanych, a `description` do wyliczania definicji. W każdym z nich element wyliczenia zaczyna się od instrukcji .

```
\begin{enumerate}
\item Możesz zagnieżdzać
otoczenia do woli:
\begin{itemize}
\item Ale może to wyglądać
nieco komicznie.
\item[-] Od pauzy.
\end{itemize}
\item Więc zapamiętaj:
\begin{description}
\item[Głupoty] nie staną się
mądrościami, gdy się je wyliczy.
\item[Mądrości] można elegancko
zestawiać w wyliczeniach.
\end{description}
\end{enumerate}
```

1. Możesz zagnieżdzać otoczenia do woli:

- Ale może to wyglądać nieco komicznie.
- Od pauzy.

2. Więc zapamiętaj:

**Głupoty** nie staną się mądrościami, gdy się je wyliczy.

**Mądrości** można elegancko zestawiać w wyliczeniach.

### 2.11.2 Flushleft, flushright i center

W otoczeniach `flushleft` i `flushright` akapity są składane z wyrównaniem, odpowiednio, do lewego bądź prawego marginesu. Wewnątrz otoczenia `center` każdy wiersz akapitu jest wyśrodkowany. Tak jak zawsze, `LaTeX` dzieli akapity na wiersze automatycznie, można jednak w obrębie powyższych otoczeń wymusić zmianę wiersza poleceniem `\\`.

```
\begin{flushleft}
To jest tekst\\
wyrównany do lewej.
\LaTeX{} nie składa tu wierszy\\
z zachowaniem jednakowej
długości.
\end{flushleft}
```

To jest tekst  
wyrównany do lewej. `LaTeX` nie składa tu wierszy z zachowaniem jednakowej długości.



```
\begin{flushright}
To jest tekst\\
wyrównany do prawej.
\LaTeX{} nie składa tu wierszy\\
z zachowaniem jednakowej
długości.
\end{flushright}
```

To jest tekst  
wyrównany do prawej.  $\LaTeX$  nie składa  
tu wierszy  
z zachowaniem jednakowej długości.

```
\begin{center}
To jest tekst\\wyśrodkowany.
\end{center}
```

To jest tekst  
wyśrodkowany.

### 2.11.3 Quote, quotation i verse

Otoczenie `quote` przydaje się do składania cytatów oraz przykładów.

```
Jeżeli chodzi o długość wierszy,
to regułą kciuka jest, że:
\begin{quote}
Przeciętnie wiersz nie powinien
zawierać więcej niż 66 znaków.
\end{quote}
Dlatego w  $\LaTeX$ u standardowe
strony mają szerokie marginesy.
Dlatego też w gazetach stosuje
się druk wielołamowy.
```

Jeżeli chodzi o długość wierszy, to regułą  
kciuka jest, że:

Przeciętnie wiersz nie powi-  
nien zawierać więcej niż 66  
znaków.

Dlatego w  $\LaTeX$ u standardowe strony ma-  
ją szerokie marginesy. Dlatego też w gaze-  
tach stosuje się druk wielołamowy.

Istnieją ponadto dwa otoczenia o podobnym zastosowaniu: `quotation` oraz `verse`. Pierwsze z nich przydaje się do formatowania cytatów dłuższych niż jeden akapit. W przeciwieństwie do otoczenia `quote`, wewnątrz `quotation`  $\LaTeX$  rozpoczyna poszczególne akapity od wcięcia akapitowego. Otoczenie `verse` służy do składania wierszy. Poszczególne linijki zwrotek należy kończyć instrukcją `\\`, poszczególne zaś zwrotki – oddzielać pustą linią.

```
Na pamięć znam tylko
jeden wierszyk o kotku.
\begin{flushleft}
\begin{verse}
Wlazł kotek na płotek i mruga,\
piękna to piosneczka niedługa.\

Wlazł kurek na murek i pieje;\
niech się nikt z tych piosnek
nie śmieje.
\end{verse}
\end{flushleft}
```

Na pamięć znam tylko jeden wierszyk o kotku.

Wlazł kotek na płotek  
i mruga,  
piękna to piosneczka  
niedługa.  
Wlazł kurek na murek  
i pieje;  
niech się nikt z tych piosnek  
nie śmieje.

#### 2.11.4 Streszczenie

Publikacje naukowe zaczynają się zazwyczaj od streszczenia – przeglądu tego, co czytelnik napotka w dalszej części. W  $\LaTeX$ u do wyróżniania streszczeń służy otoczenie `abstract`. Używa się go na ogół w dokumentach klasy `article`.

```
\begin{abstract}
Streszczenie streszczenia.
\end{abstract}
```

Streszczenie streszczenia.

#### 2.11.5 Pisanie dosłowne

Tekst zawarty między `\begin{verbatim}` a `\end{verbatim}` jest składany przez  $\LaTeX$ a dosłownie, czyli tak, by wyglądał jak napisany na maszynie, z zachowaniem zmian wiersza i odstępów z pliku źródłowego oraz ignorując polecenia  $\LaTeX$ a.

Wewnątrz akapitów podobny wynik uzyskuje się za pomocą instrukcji

```
\verb+tekst+
```

Znak `+` jest tu tylko przykładowym znakiem ograniczającym. Możesz użyć dowolnego znaku oprócz liter, `*` i spacji. Polecenie to jest często wykorzystywane w tej książce do przedstawiania przykładów poleceń  $\LaTeX$ a.

Polecenie `\verb|\ldots| \ldots`

```
\begin{verbatim}
10 PRINT "HELLO WORLD ";
20 GOTO 10
\end{verbatim}
```

Polecenie `\ldots ...`

```
10 PRINT "HELLO WORLD ";
20 GOTO 10
```

```
\begin{verbatim*}
wersja z gwiazdką
otoczenia verbatim
wyróżnia spacje
w tekście
\end{verbatim*}
```

```
wersja_z_gwiazdką
otoczenia_verbatim
wyróżnia_spacje
w_tekście
```

Podobnie działa wersja polecenia `\verb`.

`\verb*|w ten sposób :-)|`

```
w_ten_sposób_:-)
```

Otoczenie `verbatim` oraz polecenie `\verb` nie mogą być wykorzystywane jako argumenty innych poleceń.

### 2.11.6 Tabular

Otoczenie `tabular` służy do składania pięknych tabel z opcjonalnymi liniami poziomymi i pionowymi.  $\LaTeX$  automatycznie ustala szerokość poszczególnych rubryk tabeli.

Argument *specyfikacja tabeli* w poleceniu

```
\begin{tabular}[pozycja]{specyfikacja tabeli}
```

definiuje format tabeli. Użyj `l` aby kolumna była wyrównana do lewej, `r` by była wyrównana do prawej, a `c` dla wyśrodkowania. Z kolei `p{szerość}` wstawi kolumnę wyjustowaną o zadanej szerokości. `l` wstawia linie pionowe.

Jeśli tekst kolumny jest zbyt długi,  $\LaTeX$  zawinie go automatycznie tylko w przypadku użycia `p{szerość}`. Kolumna ta będzie miała wówczas określoną szerokość, a tekst będzie zawijany podobnie jak w akapitach.

Argument *pozycja* definiuje pionowe położenie tabeli względem linii tekstu. Użyj pozycji `t`, `b` oraz `c`, by ustawić położenie odpowiednio względem góry, dołu oraz środka.

Wewnątrz otoczenia `tabular`, `&` przechodzi do następnej kolumny, `\\` rozpoczyna następny wiersz, a `\hline` wstawia poziomą linię na całą długość tabeli. Możesz dodać częściowe linie przy użyciu `\cline{i-j}`, gdzie  $i$  i  $j$  są numerami kolumn nad którymi powinna się ona rozciągać.

```
\begin{tabular}{|r|l|}
\hline
7C0 & szesnastkowo \\
3700 & ósemkowo \\
11111000000 & dwójkowo \\
\hline \hline
1984 & dziesiętnie \\
\hline
\end{tabular}
```

7C0	szesnastkowo
3700	ósemkowo
11111000000	dwójkowo
1984	dziesiętnie

```
\begin{tabular}{|p{4.7cm}|}
\hline
Ten akapit jest wewnątrz
pudełka. Mamy nadzieję, że
uzyskany efekt się podoba.\\
\hline
\end{tabular}
```

Ten akapit jest wewnątrz pudełka. Mamy nadzieję, że uzyskany efekt się podoba.
--------------------------------------------------------------------------------

Instrukcją `@{...}` określamy odstęp między kolumnami. Zastępuje ona domyślny odstęp międzykolumnowy treścią umieszczoną między nawiasami klamrowymi. Jednym zastosowaniem jest wyrównanie liczb względem przecinka, czego przykład pokazany jest poniżej. Można ją także wykorzystać do usunięcia odstępów w pierwszej i ostatniej kolumnie tabeli przy użyciu `@{}`.

```
\begin{tabular}{@{} l @{}}
\hline
bez odstępów na brzegach\\
\hline
\end{tabular}
```

bez odstępów na brzegach
--------------------------

```
\begin{tabular}{|l}
\hline
odstępy na brzegach tabeli\\
\hline
\end{tabular}
```

odstępy na brzegach tabeli
----------------------------

W  $\LaTeX$ u nie ma mechanizmu pozwalającego wyrównywać zestawienia liczbowe według cyfr znaczących<sup>10</sup>, ale efekt ten można uzyskać, składając liczbę w dwóch kolumnach: część całkowitą w kolumnie wyrównywanej do prawego brzegu i część dziesiętną w kolumnie wyrównywanej do lewego. Za pomocą instrukcji `@{,}` zastępujemy przecinkiem odstęp wstawiany normalnie między kolumnami. Trzeba jednak pamiętać o konieczności wpisywania znaku `&` zamiast przecinków w liczbach. Rubryki rozciągające się na kilka kolumn, jak nagłówek w poniższym przykładzie, tworzymy poleceniem `\multicolumn`:

```
\begin{tabular}{c r @{.} l}
Wyrażenie Pi      & &
\multicolumn{2}{c}{Wartość} \\
\hline
 $\pi$  & 3&1416 & \\
 $\pi^\pi$  & 36&46 & \\
 $(\pi^\pi)^\pi$  & 80662&7 & \\
\end{tabular}
```

Wyrażenie Pi	Wartość
$\pi$	3.1416
$\pi^\pi$	36.46
$(\pi^\pi)^\pi$	80662.7

```
\begin{tabular}{|c|c|}
\hline
\multicolumn{2}{|c|}{Ene due} \\
\hline
like & fake! \\
\hline
\end{tabular}
```

Ene due	
like	fake!

Tekst zawarty wewnątrz otoczenia tabular zawsze jest na jednej stronie. Jeśli chcesz złożyć długą tabelę, zobacz na pakiet `longtable`.

Czasami domyślne tabele w  $\LaTeX$ u wydają się nieco ciasne. Aby zwiększyć nieco ich odstęp, ustaw `\arraystretch` oraz `\tabcolsep` na większe wartości.

<sup>10</sup>Do wyrównywania cyfr można skorzystać z pakietu `dcolumn` z zestawu „tools”.

```

\begin{tabular}{|l|}
\hline
Te linie\\\hline
są ciasne\\\hline
\end{tabular}

{\renewcommand{\arraystretch}{1.5}
\renewcommand{\tabcolsep}{0.2cm}
\begin{tabular}{|l|}
\hline
Mniej stłoczony\\\hline
układ tabeli\\\hline
\end{tabular}}

```

Te linie
są ciasne
Mniej stłoczony
układ tabeli

Jeśli chcesz zwiększyć rozmiar pojedynczego wiersza w tabeli dodaj niewidzialną pionową linię<sup>11</sup>. Użyj `\rule` aby faktycznie to wykorzystać.

```

\begin{tabular}{|c|}
\hline
\rule{1pt}{4ex}Pitprop \ldots\\
\hline
\rule{0pt}{4ex}Strut\\
\hline
\end{tabular}

```

Pitprop ...
Strut

pt oraz ex w powyższym przykładzie to jednostki  $\text{T}_{\text{E}}\text{X}$ a. Możesz przeczytać o nich więcej w tabeli 6.5 na stronie 135.

Kilka dodatkowych poleceń rozszerzających otoczenie tabular można znaleźć w pakiecie `booktabs`. Ułatwia on tworzenie profesjonalnie wyglądających tabeli z odpowiednimi odstępami.

## 2.12 Wstawianie grafik i obrazów

Jak wyjaśniono w poprzedniej sekcji,  $\text{L}_{\text{A}}\text{T}_{\text{E}}\text{X}$  pozwala na pracę z nieszttywnymi elementami, takimi jak obrazki czy grafiki przy użyciu otoczeń `figure` i `table`.

Solidnym zbiorem poleceń pozwalającym na dołączenie obrazków do nieszttywnych otoczeń jest pakiet `graphicx` autorstwa D. P. Carlisle. Należy on do rodziny pakietów o nazwie „graphics”<sup>12</sup>.

<sup>11</sup>Profesjonalnie taki obiekt nazywa się `strut`.

<sup>12</sup>[CTAN://pkg/graphics](http://CTAN://pkg/graphics)

Tabela 2.3: Nazwy kluczy w pakiecie `graphicx`.

<code>width</code>	skaluje grafikę do wskazanej szerokości
<code>height</code>	skaluje grafikę do wskazanej wysokości
<code>angle</code>	obraca grafikę przeciwnie do wskazówek zegara
<code>scale</code>	skaluje grafikę

Aby wstawić grafikę do dokumentu wystarczy podać za następującymi krokami:

1. Wyeksportuj obraz z Twojego programu graficznego do formatu EPS, PDF, PNG lub JPEG.
2. Jeśli Twoja grafika jest w formacie wektorowym EPS to musisz najpierw skonwertować ją na PDF przed jej użyciem. Istnieje narzędzie `epstopdf` które służy do tego celu. Zwróć uwagę, że czasami eksport do EPS może być logicznym krokiem nawet gdy program potrafi eksportować do PDF, ponieważ PDF mają rozmiar strony, a pliki EPS mogą mieć mniejsze otoczenie, bardziej dopasowane do faktycznej grafiki.
3. Dodaj pakiet `graphicx` do swojej preambuły przy użyciu polecenia

```
\usepackage{graphicx}
```

4. Użyj polecenia

```
\includegraphics[klucz=wartość, ...]{nazwa pliku}
```

aby wstawić plik *nazwa pliku* do twojego dokumentu. Opcjonalny argument to lista oddzielonych przecinkami *kluczy* i przypisanym ich *wartości*. Za jego pomocą można zmienić wysokość, szerokość oraz rotację grafiki. Tabela 2.3 przedstawia najważniejsze klucze.

Przykład kodu w rysunku 2.2 na stronie 46 przedstawia sposób użycia. Wstawia on grafikę z pliku `test.png`. Grafika jest *najpierw* obracana o 90 stopni, a *następnie* przeskalowana do końcowej szerokości równej połowie szerokości standardowego akapitu. Ponieważ wysokość nie jest podana,

---

```
\includegraphics[angle=90,width=\textwidth]{test.png}
```

---

Rysunek 2.2: Przykład kodu wstawiającego `test.png` w dokument.

zostanie ona automatycznie dobrana tak by nie zniekształcała obrazka. Argumenty wysokości i szerokości mogą być podane również przy użyciu jednostek absolutnych. Informacje o nich znajdziesz w tabeli 6.5 na stronie 135. Więcej informacji o tym temacie możesz znaleźć w [12].

## 2.13 Wstawki nieszttywne

Współczesne publikacje zawierają dużo rysunków i tabel. Elementów tych nie należy dzielić między strony i dlatego wymagają specjalnego potraktowania. Jednym z rozwiązań byłoby rozpoczynanie za każdym razem nowej strony, jednak pozostawiałyby to zbyt wiele pustego miejsca i wyglądało nieelegancko.

Rozwiązanie tego problemu jest by nie traktować położenia tabel i grafik „sztywno”, lecz pozwolić im przenieść się na następną stronę, a poprzednią wypełnić tekstem.  $\LaTeX$  oferuje dwa otoczenia dla wstawek nieszttywnych. Otoczenie `figure` służy do tworzenia rysunków, a do tabel otoczenie `table`. Aby w pełni wykorzystać potencjał wstawek nieszttywnych warto znać zarys sposobu ich działania w  $\LaTeX$ u. Pomaga to uniknąć frustracji gdy nie są one tam gdzie byśmy chcieli.

Zacznijmy od przyjrzenia się wspomnianym poleceniom  $\LaTeX$ a:

Wszystkie elementy zawarte w otoczeniach `figure` lub `table` będą traktowane jako nieszttywne. Oba otoczenia mają jeden parametr opcjonalny

```
\begin{figure}[specyfikacja położenia] lub \begin{table}[...]
```

nazywany *specyfikacja położenia*. Ten parametr mówi  $\LaTeX$ owi o miejscach do których dany element nieszttywny może zostać przeniesiony. *Specyfikacja położenia* zbudowana jest z ciągu *przyzwoleń wstawek*. Zobacz tabelę 2.4.

Przykładowa tabela może się zaczynać następująco

```
\begin{table} [!hbp]
```

Argument `[!hbp]` oznacza, że tabelę można umieścić w miejscu, w którym pojawia się w pliku źródłowym (h), albo na dole strony (b), albo wreszcie



na osobnej stronie zawierającej wyłącznie wstawki (p). Ponadto „!” oznacza, że  $\LaTeX$  ma pominąć większość parametrów sterujących umieszczaniem wstawek. Jeżeli otoczenia `table` użyto bez opcjonalnego argumentu, to jego domyślnymi wartościami są `[tbp]`.

$\LaTeX$  umieszcza każdą wstawkę zgodnie ze specyfikacją autora podaną w argumencie *specyfikacja położenia*. Jeżeli nie może umieścić wstawki na bieżącej stronie, to dołącza ją albo do *kolejki rysunków*, albo do *kolejki tabel*<sup>13</sup>. Na początku składania nowej strony  $\LaTeX$  sprawdza, czy można ją zappełnić wstawkami czekającymi w kolejce. Jeśli nie jest to możliwe, to pierwsza wstawka każdej z kolejek traktowana jest tak, jak gdyby właśnie pojawiła się w tekście:  $\LaTeX$  stara się ją umieścić zgodnie z wartościami parametru *specyfikacja położenia* (za wyjątkiem `h`, gdyż nie jest to już oczywiście możliwe). Nowe wstawki dołączane są na koniec odpowiednich kolejek.  $\LaTeX$  dba o właściwy porządek wstawek każdego typu. Może się zdarzyć, że pojedynczy rysunek, którego z jakichś względów nie można poprawnie wstawić, „ciągnie” za sobą wszystkie późniejsze rysunki, nawet aż na koniec dokumentu. Dlatego:

Jeżeli  $\LaTeX$  nie umieszcza wstawek nieszytywnych zgodnie z oczekiwaniem, to z reguły tylko jedna z nich blokuje którąś z kolejek.

Pomimo tego, że  $\LaTeX$ owi można podać jednoliterowe specyfikacje położenia, będzie to powodować problemy. Jeśli wstawka nie mieści się we wska-

<sup>13</sup>Są to kolejki typu FIFO (pierwsze weszło – pierwsze wyjdzie).

Tabela 2.4: Przyzwolenia elementów nieszytywnych.

Znak	Dopuszczalne miejsce umieszczenia elementu ...
h	bez przemieszczenia, dokładnie w miejscu użycia. Użyteczne w wypadku niewielkich wstawek.
t	na górze strony
b	na dole strony
p	na stronie zawierającej wyłącznie elementy nieszytywne.
!	ignorując większość parametrów kontrolujących <sup>a</sup> , które mogą nie pozwolić na umieszczanie ich na stronie.

<sup>a</sup>Takie jak maksymalna liczba elementów nieszytywnych na jednej stronie.

zanej lokacji, zablokuje ona zarówno siebie jak i następne wstawki. W szczególności unikaj korzystania tylko z opcji [h]. Jest ona tak restrykcyjna, że w nowszych wersjach L<sup>A</sup>T<sub>E</sub>Xa jest ona automatycznie zastępowana przez [ht].

Wyjaśnwszy ów cokolwiek zawiły problem umieszczania wstawek, przejdźmy do omówienia kilku pozostałych spraw z nimi związanych. Poleceniem

```
\caption{tytuł}
```

wstawiamy tytuł rysunku lub tabeli. Kolejny numer rysunku bądź tabeli oraz słowo „Rysunek” bądź „Tabela” (lub „Tablica” – zależnie od używanego pakietu polonizacyjnego) zostaną wstawione automatycznie przez L<sup>A</sup>T<sub>E</sub>Xa.

Następujące instrukcje

```
\listoffigures oraz \listoftables
```

działają analogicznie do instrukcji \tableofcontents, wstawiając do dokumentu, odpowiednio, spis rysunków oraz spis tabel. Poszczególnymi pozycjami tych spisów będą tytuły rysunków bądź tabel będące argumentami instrukcji \caption. Jeżeli tytuł jest długi, to do spisu można przesłać jego wersję skróconą, podaną jako opcjonalny argument instrukcji \caption:

```
\caption[Krótkie]{DDDDłłłłłłuuuuuuuugggggiiiiieeee}
```

Za pomocą instrukcji \label oraz \ref można tworzyć odsyłacze do tabel i rysunków. Polecenie \label należy umieszczać *bezpośrednio* za instrukcją \caption, ponieważ chcemy by odnosiła się do numeru przez nią wygenerowanego.

W poniższym przykładzie rysowany jest kwadrat, a następnie wstawiony do dokumentu. Ten sposób postępowania można wykorzystać w celu zarezerwowania miejsca na rysunki, które zostaną wklejone później do gotowego dokumentu.

```
Rysunek~\ref{white} jest przykładem Pop-Artu.
\begin{figure}[!hbt]
\includegraphics[angle=90,width=\textwidth]{white-box.pdf}
\caption{Biały kwadrat autorstwa
Peter Markus Paulian\label{white}}
\end{figure}
```

Zakładając w tym przykładzie, że kolejka rysunków jest pusta,  $\LaTeX$  najpierw będzie się *bardzo* (!) starał umieścić rysunek bez przesuwania go do-  
kądkolwiek (h). Jeżeli okaże się to niemożliwe, to spróbuje go umieścić na  
dole strony (b). Jeżeli i to okaże się niewykonalne, to będzie się starał umie-  
ścić rysunek na stronie zawierającej wyłącznie wstawki (p). Jeżeli w kolej-  
kach rysunków i tabel nie ma wstawek pozwalających wypełnić stronę, to  
 $\LaTeX$  rozpocznie nową stronę i spróbuje umieścić na niej rysunek, traktując  
go znowu tak, jakby właśnie pojawił się w tekście.

Czasami może wystąpić konieczność wykonania instrukcji

```
\clearpage albo nawet \cleardoublepage
```

Rozkazuje ona  $\LaTeX$ owi natychmiast umieścić wszystkie wstawki z kolejki  
i rozpocząć nową stronę. `\cleardoublepage` upewnia się, że nowa strona  
będzie po prawej stronie książki.



## Rozdział 3

# Skład wyrażen matematycznych

Nareszcie! W tym rozdziale poznasz najlepszą stronę  $\text{T}_{\text{E}}\text{X}$ a, czyli skład wzorów matematycznych. Ostrzegamy jednak, że przedstawimy tu jedynie absolutne podstawy. Chociaż wystarczają one większości użytkowników, to nie załamuj rąk, jeśli nie poradzisz sobie z jakimś skomplikowanym wzorem, lecz zapoznaj się z możliwościami  $\mathcal{A}\mathcal{M}\mathcal{S}\text{-}\mathbb{L}\text{A}\text{T}_{\text{E}}\text{X}$ a.

### 3.1 Pakiety $\mathcal{A}\mathcal{M}\mathcal{S}\text{-}\mathbb{L}\text{A}\text{T}_{\text{E}}\text{X}$

Aby składać (zaawansowaną) matematykę warto użyć  $\mathcal{A}\mathcal{M}\mathcal{S}\text{-}\mathbb{L}\text{A}\text{T}_{\text{E}}\text{X}$ . Jest to kolekcja pakietów i klas przeznaczona do matematycznego składu. Będziemy głównie zajmować się pakietem `amsmath`, który jest jej częścią.  $\mathcal{A}\mathcal{M}\mathcal{S}\text{-}\mathbb{L}\text{A}\text{T}_{\text{E}}\text{X}$  powstało pod egidą Amerykańskiego Towarzystwa Matematycznego (*American Mathematical Society*) i jest powszechnie wykorzystywane.  $\mathbb{L}\text{A}\text{T}_{\text{E}}\text{X}$  dostarcza niektóre podstawowe funkcje i otoczenia do składania matematyki, ale są one ograniczone (inaczej mówiąc  $\mathcal{A}\mathcal{M}\mathcal{S}\text{-}\mathbb{L}\text{A}\text{T}_{\text{E}}\text{X}$  jest *nieograniczone!*) i czasami niekonsekwentne.

Kolekcja  $\mathcal{A}\mathcal{M}\mathcal{S}\text{-}\mathbb{L}\text{A}\text{T}_{\text{E}}\text{X}$  jest jednym z wymaganych elementów w dystrybucjach i powinien być dostarczony wraz z  $\mathbb{L}\text{A}\text{T}_{\text{E}}\text{X}$ em<sup>1</sup>. W tym rozdziale zakładamy, że pakiet `amsmath` jest załadowany w preambule.

```
\usepackage{amsmath}
```

---

<sup>1</sup>Jeśli w twojej ich nie ma, to znajdziesz je pod adresem [CTAN://pkg/amslatex](http://ctan.org/pkg/amslatex)

### 3.2 Pojedyncze równania

Formuła matematyczna może być złożona wewnątrz tekstu akapitu (*styl tekstowy*), lub akapit może być złamany by była ona wyświetlona osobno (*styl pokazowy*). Matematyczne równania *wewnątrz* akapitu są wstawiane pomiędzy \$ i \$:

\$a\$ do kwadratu plus  $b$  do kwadratu równa się  $c$  do kwadratu. Albo, stosując bardziej matematyczne podejście:  $c^2 = a^2 + b^2$ .

$a$  do kwadratu plus  $b$  do kwadratu równa się  $c$  do kwadratu. Albo, stosując bardziej matematyczne podejście:  $c^2 = a^2 + b^2$ .

$\TeX$  należy wymawiać jako  $\tau\epsilon\chi$   
 $100\text{ m}^3$  wody  
 To płynie z mojego  $\heartsuit$ .

$\TeX$  należy wymawiać jako  $\tau\epsilon\chi$   
 $100\text{ m}^3$  wody  
 To płynie z mojego  $\heartsuit$ .

Składając większe wzory, powinniśmy je eksponować, to znaczy wstawiać między akapitami, w osobnym wierszu. Takie wzory umieszcza się pomiędzy  $\begin{equation}$  i  $\end{equation}$ <sup>2</sup>. Możesz im wówczas nadać etykietę i odnieść się do nich w innym miejscu przy użyciu polecenia  $\eqref$ . Jeśli chcesz nazwać równanie w jakiś konkretny sposób możesz użyć polecenia  $\tag$ .

<sup>2</sup>To otoczenie pochodzi z `amsmath`. Jeśli z jakiegoś dziwnego powodu nie masz dostępu do tej paczki możesz użyć  $\LaTeX$ owego otoczenia `displaymath`.

```

$a$ do kwadratu plus~$b$
do kwadratu równa się~$c$
do kwadratu. Albo, stosując
bardziej matematyczne
podejście
\begin{equation}
a^2 + b^2 = c^2
\end{equation}
Einstein powiedział
\begin{equation}
E = mc^2 \label{mądre}
\end{equation}
Nie powiedział
\begin{equation}
1 + 1 = 3 \tag{głupie}
\end{equation}
To odnosi się do
\eqref{mądre}.

```

*a* do kwadratu plus *b* do kwadratu równa się *c* do kwadratu. Albo, stosując bardziej matematyczne podejście

$$a^2 + b^2 = c^2 \quad (3.1)$$

Einstein powiedział

$$E = mc^2 \quad (3.2)$$

Nie powiedział

$$1 + 1 = 3 \quad (\text{głupie})$$

To odnosi się do (3.2).

Jeśli nie chcesz by  $\LaTeX$  numerował równania użyj wersji z gwiazdką otoczenia `equation`, `equation*`, lub krócej wprowadź je pomiędzy `\[ i \]`<sup>3</sup>:

```

$a$ do kwadratu plus~$b$
do kwadratu równa się~$c$
do kwadratu. Albo, stosując
bardziej matematyczne
podejście:
\begin{equation*}
a^2 + b^2 = c^2
\end{equation*}
możesz też napisać mniej
i uzyskać ten sam efekt:
\[ a^2 + b^2 = c^2 \]

```

*a* do kwadratu plus *b* do kwadratu równa się *c* do kwadratu. Albo, stosując bardziej matematyczne podejście:

$$a^2 + b^2 = c^2$$

możesz też napisać mniej i uzyskać ten sam efekt:

$$a^2 + b^2 = c^2$$

`\[` jest krótkie i słodkie, ale nie pozwala na łatwe przełączanie pomiędzy numerowanymi i nienumerowanymi równaniami, tak jak `equation` i `equation*`.

Zwróć uwagę na różnicę w składaniu równań w trybie tekstowym i pokazowym:

<sup>3</sup> To również pochodzi z `amsmath`. Standardowy  $\LaTeX$  posiada tylko otoczenie `equation` bez gwiazdki.

To jest w stylu tekstowym:

```
\lim_{n \to \infty}
\sum_{k=1}^n \frac{1}{k^2}
= \frac{\pi^2}{6}.
```

To jest w stylu pokazowym:

```
\begin{equation}
\lim_{n \to \infty}
\sum_{k=1}^n \frac{1}{k^2}
= \frac{\pi^2}{6}
\end{equation}
```

To jest w stylu tekstowym:  
 $\lim_{n \rightarrow \infty} \sum_{k=1}^n \frac{1}{k^2} = \frac{\pi^2}{6}$ . To jest w stylu pokazowym:

$$\lim_{n \rightarrow \infty} \sum_{k=1}^n \frac{1}{k^2} = \frac{\pi^2}{6} \quad (3.3)$$

W stylu tekstowym wysokie lub głębokie wyrażenia warto zawrzeć w poleceniu `\smash`. To polecenie sprawia, że  $\text{\LaTeX}$  ignoruje wysokość wyrażenia wewnątrz. Dzięki temu odległości między wierszami są wyrównane.

`\$d_{o_{l_{n_e}}}` wyrażenie matematyczne za którym następuje `\$w^{y^{s^{o^{k^i}}}}` wzór.

W porównaniu do:

```
\smash{\$d_{o_{l_{n_e}}}}
wyrażenie za którym następuje
\smash{w^{y^{s^{o^{k^i}}}}}
wzór.
```

$d_{o_{l_{n_e}}}$  wyrażenie matematyczne za którym następuje  $w^{y^{s^{o^{k^i}}}}$  wzór.

W porównaniu do:  $d_{o_{l_{n_e}}}$  wyrażenie za którym następuje  $w^{y^{s^{o^{k^i}}}}$  wzór.

### 3.2.1 Tryb matematyczny

Istnieją również różnice pomiędzy *trybem matematycznym*, a *trybem tekstowym*. Przykładowo w *trybie matematycznym*:

1. Większość odstępów i złamań linii nie ma żadnego znaczenia, ponieważ odstęp są wnioskowane z wyrażeń bądź wstawione przy użyciu specjalnych poleceń takich jak `\`, `\quad` lub `\qquad` (będą one opisane dokładniej w sekcji 3.7).
2. Puste linie nie są dozwolone. Każda formuła może zawierać tylko jeden akapit.
3. Litery we wzorach służą do oznaczania nazw zmiennych; zmienne składamy inaczej niż zwykły tekst. Jeżeli częścią wzoru ma być zwykły tekst (niepochylona czcionka i zwykłe odstęp), to należy się posłużyć instrukcją `\text{...}` (zobacz także sekcje 3.8 na stronie 71).



```
\forall x \in \mathbf{R} \colon
\quad x^2 \geq 0
```

$$\forall x \in \mathbf{R}: \quad x^2 \geq 0$$

```
x^2 \geq 0 \quad \text{for all } x \in \mathbf{R}
```

$$x^2 \geq 0 \quad \text{for all } x \in \mathbf{R}$$

Matematycy potrafią być niezwykle wybredni jeśli chodzi o używane symbole. Na przykład we wzorach, w których występują oznaczenia zbiorów (jak powyższy), często stosuje się krój, w którym te oznaczenia przypominają odmianę „grubą”, pisaną kredą na tablicy. Uzyskuje się je przy użyciu polecenia `\mathbb` z pakietu `amssymb`<sup>4</sup>. Ostatni przykład wygląda wtedy następująco

```
x^2 \geq 0 \quad \text{for all } x
\in \mathbb{R}
```

$$x^2 \geq 0 \quad \text{for all } x \in \mathbb{R}$$

Zobacz tabelę 3.14 na stronie 82 oraz tabelę 6.4 na stronie 130 w celu porównania innych fontów matematycznych.

### 3.3 Części składowe wyrażeń matematycznych

W tej sekcji opiszemy ważniejsze instrukcje do składu wyrażeń. Większość opisywanych tu poleceń nie wymaga pakietu `amsmath` (w przeciwnym razie będzie to wyraźnie zaznaczone), ale i tak go załaduj.

**Małe litery alfabetu greckiego** wprowadzamy, używając instrukcji typu: `\alpha`, `\beta`, `\gamma` itd., a **duże**: `\Delta`, `\Gamma` itd.<sup>5</sup>:

Pełną listę liter greckich możesz znaleźć w tabeli 3.2 na stronie 77.

```
\lambda, \xi, \pi, \theta, \mu,
\Phi, \Omega, \Delta
```

$$\lambda, \xi, \pi, \theta, \mu, \Phi, \Omega, \Delta$$

**Wykładniki, indeksy górne i dolne** mogą być uzyskane przy użyciu znaków `^` oraz `_`. Argumentem większości instrukcji do składu matematyki

<sup>4</sup>`amssymb` nie jest częścią kolekcji `AMS-LATEX`, ale prawdopodobnie jest częścią Twojej dystrybucji `LATEX`a. Sprawdź swoją dystrybucję lub przejdź do `CTAN:/fonts/amsfonts/latex/` aby go uzyskać.

<sup>5</sup>W `LATEX`u nie ma wielkich liter `Alpha`, `Beta`, ..., ponieważ wyglądają one identycznie jak litery `A`, `B`, ... alfabetu łacińskiego.

jest tylko jeden znak, więc jeżeli polecenie ma dotyczyć grupy znaków, to należy je umieścić wewnątrz pary nawiasów klamrowych  $\{ \dots \}$ .

W tabeli 3.3 na stronie 78 znajduje się lista wielu relacji binarnych takich jak  $\subseteq$  czy  $\perp$ .

```
$p^3_{ij} \quad \backslash\text{Knuth} \quad \backslash\sum_{k=1}^3 k \quad \backslash\neq a^{x+y} \quad e^{x^2} \quad \backslash\neq e^{x^2}$
```

$$p_{ij}^3 \quad m_{\text{Knuth}} \quad \sum_{k=1}^3 k$$

$$a^x + y \neq a^{x+y} \quad e^{x^2} \neq e^{x^2}$$

**Pierwiastek kwadratowy** jest wprowadzany przy użyciu polecenia `\sqrt{}`; pierwiastek stopnia  $n$  jest wprowadzany przy użyciu `\sqrt[n]{}`. Wielkość znaku pierwiastka jest przez L<sup>A</sup>T<sub>E</sub>Xa ustalana automatycznie. Zapis samego znaku pierwiastka umożliwia instrukcja `\surd`.

W tabeli 3.6 na stronie 79 można znaleźć wiele przykładów strzałek takich jak  $\leftrightarrow$  czy  $\Leftrightarrow$ .

```
$$\sqrt{x} \Leftrightarrow x^{1/2} \quad \sqrt[3]{2} \quad \sqrt{x^2 + \sqrt{y}} \quad \sqrt{x^2 + y^2} \quad \surd[x^2 + y^2]$
```

$$\sqrt{x} \Leftrightarrow x^{1/2} \quad \sqrt[3]{2} \quad \sqrt{x^2 + \sqrt{y}} \quad \sqrt{x^2 + y^2}$$

Znak mnożenia **kropka** zwyczajowo jest pomijany, jednak czasami jest wstawiany dla wskazania grupowania w formule. Użyj polecenia `\cdot` aby wstawić pojedynczą wyśrodkowaną kropkę. `\cdots` wstawia trzy wyśrodkowane **kropki** podczas gdy `\ldots` wstawia trzy kropki wyrównane do linii tekstu. Oprócz tego istnieją również polecenia `\vdots` dla kropek pionowych oraz `\ddots` dla kropek przekątnych. Więcej przykładów znajdziesz w sekcji 3.6.

```
$$\Psi = v_1 \cdot v_2 \quad \cdots \quad n! = 1 \cdot 2 \cdots (n-1) \cdot n \quad \vdots \quad \ddots$
```

$$\Psi = v_1 \cdot v_2 \cdots \quad n! = 1 \cdot 2 \cdots (n-1) \cdot n$$

Polecenia `\overline{}` oraz `\underline{}` umieszczają nad i pod wyrażeniami poziome kreski:

```
0{,}(3) = \underline{\underline{1/3}}
```

$$0,(3) = \underline{\underline{1/3}}$$

Instrukcje `\overbrace` oraz `\underbrace` umieszczają nad i pod wyrażeniami poziome klamry:

```
\underbrace{\overbrace{a+b+c}^6
\cdot \overbrace{d+e+f}^7}
\_text{meaning of life} = 42
```

$$\underbrace{a + b + c \cdot d + e + f}_{\text{meaning of life}} = 42$$

Akcenty matematyczne, takie jak **małe strzałki** czy **tyldy** nad zmiennymi, umieszczamy we wzorze poleceniami z tabeli 3.1 na stronie 77. Szerokie daszki i tyldy, obejmujące wiele symboli, wstawiamy za pomocą instrukcji `\widetilde` oraz `\widehat`. Zwróć uwagę na różnice pomiędzy `\hat` i `\widehat`, oraz na umieszczenie `\bar` przy zmiennej z indeksem dolnym. Znakiem ' oznaczamy symbol „prim”:

```
f(x) = x^2 \quad f'(x)
= 2x \quad f''(x) = 2\[[5pt]
\hat{XY} \quad \widehat{XY}
\quad \bar{x}_0 \quad \bar{x}_{0}
```

$$f(x) = x^2 \quad f'(x) = 2x \quad f''(x) = 2$$

$$\hat{XY} \quad \widehat{XY} \quad \bar{x}_0 \quad \bar{x}_0$$

**Wektory** oznacza się niekiedy akcentem w postaci strzałki nad nazwą zmiennej. Służą do tego polecenie `\vec`. Natomiast do oznaczenia wektora od punktu  $A$  do punktu  $B$  korzystamy z poleceń `\overrightarrow` oraz `\overleftarrow`:

```
\vec{a} \quad \vec{AB} \quad \overrightarrow{AB}
```

$$\vec{a} \quad \vec{AB} \quad \overrightarrow{AB}$$

Nazwy funkcji typu „logarytm” należy składać odmianą prostą, nie zaś kursywą, zarezerwowaną dla nazw zmiennych, więc L<sup>A</sup>T<sub>E</sub>X definiuje wiele poleceń służących do składu najczęściej używanych funkcji matematycznych:

```
\arccos \cos \csc \exp \ker \limsup
\arcsin \cosh \deg \gcd \lg \ln
\arctan \cot \det \hom \lim \log
\arg \coth \dim \inf \liminf \max
\sinh \sup \tan \tanh \min \Pr
\sec \sin
```

```
\begin{equation*}
  \lim_{x \rightarrow 0}
  \frac{\sin x}{x}=1
\end{equation*}
```

$$\lim_{x \rightarrow 0} \frac{\sin x}{x} = 1$$

Jeśli wymienione funkcje są dla Ciebie niewystarczające, użyj polecenia `\DeclareMathOperator`. Istnieje również wersja z gwiazdką dla funkcji z granicami. Polecenie to działa tylko w preambule więc w poniższym przykładzie są podane w komentarzach.

```
%\DeclareMathOperator{\argh}{argh}
%\DeclareMathOperator*{\nut}{Nut}
\begin{equation*}
  3\argh = 2\nut_{x=1}
\end{equation*}
```

$$3 \operatorname{argh} = 2 \operatorname{Nut}_{x=1}$$

Dla funkcji typu modulo istnieją dwie instrukcje: `\bmod` dla binarnego operatora „ $a \bmod b$ ” oraz `\pmod` do składu takich wyrażeń jak „ $x \equiv a \pmod{b}$ ”.

```
$a\bmod b \\\
x\equiv a \pmod{b}$
```

$$a \bmod b \\ x \equiv a \pmod{b}$$

**Ułamki piętrowe** składa się poleceniem `\frac{...}{...}`. W stylu tekstowym ułamki są nieco niższe by zmieścić się w wierszu. Aby wprowadzać zmniejszone ułamki w trybie pokazowym należy użyć polecenia `\tfrac`. W celu uzyskania efektu odwrotnego, tj. ułamki nieściśnięte w trybie tekstowym, użyj polecenia `\dfrac`. Często preferowana jest forma  $1/2$ , ponieważ wygląda lepiej zwłaszcza w wypadku niewielkich porcji „materiału ułamkowego”.

```
W stylu pokazowym:
\begin{equation*}
  3/8 \quad \frac{3}{8}
  \quad \quad \tfrac{3}{8}
\end{equation*}
```

W stylu pokazowym:

$$3/8 \quad \frac{3}{8} \quad \tfrac{3}{8}$$

```
W stylu tekstowym:
$1\frac{1}{2}$~godziny \quad
$1\dfrac{1}{2}$~godziny
```

W stylu tekstowym:  $1\frac{1}{2}$  godziny  
 $1\frac{1}{2}$  godziny

Poniżej użyto polecenia `\partial` do złożenia pochodnych częściowych :

```
\begin{equation*}
\sqrt{\frac{x^2}{k+1}}\quad
x^{\frac{2}{k+1}}\quad
\frac{\partial^2 f}{\partial x^2}
\end{equation*}
```

$$\sqrt{\frac{x^2}{k+1}} \quad x^{\frac{2}{k+1}} \quad \frac{\partial^2 f}{\partial x^2}$$

Aby złożyć współczynnik dwumianowy lub podobną konstrukcję, użyj polecenia `\binom` z `amsmath`:

```
Reguła Pascala twierdzi
\begin{equation*}
\binom{n}{k} = \binom{n-1}{k}
+ \binom{n-1}{k-1}
\end{equation*}
```

Reguła Pascala twierdzi

$$\binom{n}{k} = \binom{n-1}{k} + \binom{n-1}{k-1}$$

Czasami do stworzenia relacji binarnych może się okazać użyteczne złożenie dwóch symboli jeden nad drugim. Polecenie `\stackrel{\#1}{\#2}` składa swój pierwszy argument czcionką pomniejszoną, jaka stosowana jest do indeksów, i umieszcza go nad drugim argumentem, złożonym czcionką normalnej wielkości.

```
\begin{equation*}
f_n(x) \stackrel{*}{\approx} 1
\end{equation*}
```

$$f_n(x) \stackrel{*}{\approx} 1$$

**Znak całki** składamy poleceniem `\int`, **znak sumowania** instrukcją `\sum`, zaś **operator iloczynu** za pomocą instrukcji `\prod`. Górne granice całkowania i sumowania określamy za pomocą `^`, a dolne znakiem `_`, czyli podobnie jak w wypadku indeksów górnych i dolnych:

```
\begin{equation*}
\sum_{i=1}^n \quad
\int_0^{\frac{\pi}{2}} \quad
\prod_{\epsilon}
\end{equation*}
```

$$\sum_{i=1}^n \quad \int_0^{\frac{\pi}{2}} \quad \prod_{\epsilon}$$

Pakiet `amsmath` zawiera dodatkowe narzędzie do sterowania położeniem indeksów w złożonych wyrażeniach: instrukcję `\substack`:

```
\begin{equation*}
\sum_{\substack{0 < i < n \\ j \subseteq i}} P(i,j) = Q(i,j)
\end{equation*}
```

$$\sum_{\substack{0 < i < n \\ j \subseteq i}}^n P(i,j) = Q(i,j)$$

Do składu **nawiasów** i innych **ograniczników** typu ( [ < || ↓ mamy różnorodność symboli. Nawiasy okrągłe i kwadratowe wstawiamy bezpośrednio z klawiatury. Do nawiasów klamrowych stosujemy \{ oraz \}. Wszystkie inne ograniczniki wstawiamy, używając specjalnych poleceń, np. \updownarrow.

```
\begin{equation*}
\{a,b,c\} \neq \{a,b,c\}
\end{equation*}
```

$$a, b, c \neq \{a, b, c\}$$

Poprzedzenie otwierającego ogranicznika poleceniem \left, a zamykającego poleceniem \right powoduje automatyczne ustalenie jego rozmiaru w zależności od wielkości zawartego między nimi wyrażenia. Zwróć uwagę, że każde użycie \left oraz ogranicznika wymaga nawiasu zamykającego poprzedzonego poleceniem \right. Gdy ogranicznik ma się pojawić tylko po jednej stronie, wówczas po drugiej należy użyć niewidzialnego ogranicznika \right.:

```
\begin{equation*}
1 + \left(\frac{1}{1-x^2}\right)^3 \quad \left. \ddagger \frac{\sim}{\sim} \right)
\end{equation*}
```

$$1 + \left(\frac{1}{1-x^2}\right)^3 \quad \ddagger -$$

W pewnych sytuacjach trzeba samemu określić właściwą wielkość ogranicznika. Do tego celu służą instrukcje \big, \Big, \bigg oraz \Bigg, poprzedzające odpowiedni ogranicznik:

```
\Big((x+1)(x-1)\Big)^2 \\
\big(\ \Big(\ \bigg(\ \Bigg(\ \quad \\
\big\} \ \Big\} \ \bigg\} \ \Bigg\} \quad \\
\big\| \ \Big\| \ \bigg\| \ \Bigg\| \quad \\
\big\downarrow \ \Big\downarrow \ \bigg\downarrow \ \Bigg\downarrow \\
\bigg\downarrow \ \Bigg\downarrow
```

$$\left(\left(\left(\left(x+1\right)\left(x-1\right)\right)\right)\right)^2$$

Zestawienie dostępnych ograniczników znajduje się w tabeli 3.8 na stronie 80.

### 3.4 Pojedyncze równania które są zbyt długie: multiline

Jeśli równanie jest zbyt długie musimy je jakoś podzielić. Niestety podzielone równania są zwykle trudniejsze do przeczytania niż niepodzielone. Aby zwiększyć ich czytelność istnieją pewne zasady co do ich dzielenia:

1. Równania powinny być zwykle dzielone **przed** znakiem równości lub operatorem.
2. Podział przed znakiem równości jest lepsze niż podział przed innymi operatorami.
3. Podział przed operatorem dodawania lub odejmowania jest lepszy niż przed operatorem mnożenia.
4. Wszystkie inne podziały powinny być unikane w miarę możliwości.

Najprostszym sposobem podzielenia równania jest użycie otoczenia `multiline`<sup>6</sup>:

```
\begin{multiline}
  a + b + c + d + e + f
  + g + h + i
  \\
  = j + k + l + m + n
\end{multiline}
```

$$a + b + c + d + e + f + g + h + i = j + k + l + m + n \quad (3.4)$$

W odróżnieniu od otoczenia `equation` pozwala ono na łamanie linii (nawet wielokrotne) w dowolnym miejscu. Wykonujemy je za pomocą `\\` w miejscu gdzie chcemy by równanie zostało podzielone. Podobnie do `equation*` istnieje również otoczenie `multiline*`, które nie numeruje równania.

Często otoczenie `IEEEeqnarray` (więcej w sekcji 3.5) okaże się być lepszym wyborem. Rozważmy na przykład następującą sytuację:

```
\begin{equation}
  a = b + c + d + e + f
  + g + h + i + j
  + k + l + m + n + o + p
  \label{eq:equation_too_long}
\end{equation}
```

$$a = b + c + d + e + f + g + h + i + j + k + l + m + n + o + p \quad (3.5)$$

<sup>6</sup>Otoczenie `multiline` pochodzi z `amsmath`.

Tutaj problem jest tak naprawdę z prawą stroną równania, która nie mieści się w jednej linii. Użycie otoczenia `multline` skutkuje następującym rezultatem:

```
\begin{multline}
a = b + c + d + e + f
+ g + h + i + j \\
+ k + l + m + n + o + p
\end{multline}
```

$$a = b + c + d + e + f + g + h + i + j + k + l + m + n + o + p \quad (3.6)$$

Wygląda to o wiele lepiej niż (3.5), ale ma tę wadę, że znak równości nie jest już tak wyróżniony względem operatora dodawania przed  $k$  pomimo większego znaczenia w równaniu. Lepszym rozwiązaniem będzie tu użycie otoczenia `IEEEeqnarray` o którym opowiemy w sekcji 3.5.

## 3.5 Wiele równań

W najbardziej ogólnej sytuacji mamy kilka kolejnych równań, które nie mieszczą się w jednej linii. Aby je złożyć musimy uważać na wyrównanie pionowe, aby nasza szereg równań był zarówno czytelny jak i elegancki.

Przed faktycznym rozwiązaniem pokażemy kilka złych rozwiązań by zaprezentować największe wady kilku popularnych rozwiązań.

### 3.5.1 Problemy z tradycyjnymi poleceniami

Aby zgrupować kilka równań można użyć otoczenia `align`<sup>7</sup>:

```
\begin{align}
a &= b + c \\
&= d + e
\end{align}
```

$$a = b + c \quad (3.7)$$

$$= d + e \quad (3.8)$$

podejście to nie sprawdza się jednak gdy linia jest zbyt długa:

<sup>7</sup>Otoczenie `align` może być również użyte do zgrupowania kilku bloków równań obok siebie. `IEEEeqnarray` również z tym sobie poradzi przy użyciu argumentu `{rCl+rCl}`.



```
\begin{align}
a &= b + c \\
&= d + e + f + g + h + i \\
&+ j + k + l \nonumber \\
&+ m + n + o \\
&= p + q + r + s
\end{align}
```

$$a = b + c \quad (3.9)$$

$$= d + e + f + g + h + i + j + k + l \\ + m + n + o \quad (3.10)$$

$$= p + q + r + s \quad (3.11)$$

Tutaj  $+m$  powinno być poniżej  $d$ , a nie poniżej znaku równości. Doświadczony TeXnik zwróci uwagę, że `\mathrel{\phantom{=}} \negmedspace {}`, dodałoby wymagany odstęp przed  $+m+n+o$ , jednak wielu użytkowników nie jest na tyle doświadczonych i prostsze rozwiązanie byłoby dla nich przydatne.

W tym momencie otoczenie `eqnarray` wchodzi na scenę:

```
\begin{eqnarray}
a &= & b + c \\
&= & d + e + f + g + h + i \\
&+ j + k + l \nonumber \\
&& +\!: m + n + o \\
&= & p + q + r + s
\end{eqnarray}
```

$$a = b + c \quad (3.12)$$

$$= d + e + f + g + h + i + j + k + l \\ + m + n + o \quad (3.13)$$

$$= p + q + r + s \quad (3.14)$$

Jest lepiej ale wciąż nieidealnie. Odstępy między znakami równości są zbyt duże. Konkretnie **nie** są takie same jak w otoczeniach `multline` oraz `equation`:

```
\begin{eqnarray}
a &= & a = a
\end{eqnarray}
```

$$a = a = a \quad (3.15)$$

...przez co wyrażenia czasami kolidują z numerem równania pomimo tego, że powinno być dla nich wystarczająco dużo miejsca po lewej:

```
\begin{eqnarray}
a &= & b + c \\
& \\
&= & d + e + f + g + h^2 \\
&+ i^2 + j \\
\label{eq:faultyeqnarray}
\end{eqnarray}
```

$$a = b + c \quad (3.16)$$

$$= d + e + f + g + h^2 + i^2 + j \quad (3.17)$$

Pomimo tego, że otoczenie oferuje polecenie `\lefteqn`, które można użyć gdy lewa strona równania jest zbyt długa:

```
\begin{eqnarray}
\lefteqn{a + b + c + d
+ e + f + g + h}\nonumber\\
&= & i + j + k + l + m \\
&= & n + o + p + q + r + s \\
\end{eqnarray}
```

$$\begin{aligned}
 a + b + c + d + e + f + g + h \\
 &= i + j + k + l + m \quad (3.18) \\
 &= n + o + p + q + r + s \quad (3.19)
 \end{aligned}$$

to również nie jest to idealne, ponieważ prawa strona jest zbyt krótka przez co całość nie jest poprawnie wyśrodkowana:

```
\begin{eqnarray}
\lefteqn{a + b + c + d
+ e + f + g + h}
\nonumber \\
&= & i + j \\
\end{eqnarray}
```

$$\begin{aligned}
 a + b + c + d + e + f + g + h \\
 &= i + j \quad (3.20)
 \end{aligned}$$

Ponieważ ponarzekaliśmy już wystarczająco na konkurencję, możemy przejść teraz do pokazania Ci wspaniałe...

### 3.5.2 Otoczenie `IEEEeqnarray`

Otoczenie `IEEEeqnarray` jest bardzo potężnym narzędziem z wieloma opcjami. Tutaj poruszymy tylko temat jego podstawowych funkcji. Więcej informacji znajdziesz w jego instrukcji<sup>8</sup>.

Przed wszystkim by być w stanie używać otoczenia `IEEEeqnarray` musimy załadować pakiet<sup>9</sup> `IEEEtrantools`. Wstaw następującą linię w preambule Twojego dokumentu:

```
\usepackage{IEEEtrantools}
```

Siłą `IEEEeqnarray` jest możliwość ustalenia liczby *kolumn* w tablicy równań. Zwykle specyfikacją będzie `{rCl}`, *czyli*, trzy kolumny pierwsza wyrównana do prawej, druga wyśrodkowana z nieco większym odstępem (dlatego używamy wielkiej litery C zamiast małej c) a lewa jest wyrównana do lewej:

<sup>8</sup>Oficjalna instrukcja nazywa się [CTAN://macros/latex/contrib/IEEEtran/IEEEtran\\_HOWTO.pdf](http://ctan.org/tex-archive/macros/latex/contrib/IEEEtran/IEEEtran_HOWTO.pdf). Część o otoczeniu `IEEEeqnarray` można znaleźć w dodatku F.

<sup>9</sup>Pakiet `IEEEtrantools` może nie być dołączony w twojej dystrybucji. W takiej sytuacji możesz znaleźć go na CTAN.

```
\begin{IEEEeqnarray}{rCl}
a & = & b + c \\
\\
& = & d + e + f + g + h \\
& + & i + j + k \nonumber \\
& & \negmedspace {} + l \\
& + & m + n + o \\
\\
& = & p + q + r + s \\
\end{IEEEeqnarray}
```

$$a = b + c \quad (3.21)$$

$$= d + e + f + g + h + i + j + k \\ + l + m + n + o \quad (3.22)$$

$$= p + q + r + s \quad (3.23)$$

Możemy podać dowolną liczbę kolumn: `{c}` da nam tylko jedną wyśrodkowaną kolumnę, `{rCl1}` dodałoby czwartą kolumnę wyrównaną do lewej by dodać komentarze. Co więcej, oprócz `l`, `c`, `r`, `L`, `C`, `R` do wprowadzania w trybie matematycznym, dostępne są również `s`, `t`, `u` do wprowadzania tekstu wyrównanego odpowiednio do lewej, do środka i do prawej. Dodatkowe odstępy mogą być dodane przy użyciu `.`, `/` oraz `?` (później wymienione są większe)<sup>10</sup>. Zwróć uwagę, na odstępy wokół znaku równości w porównaniu do otoczenia `eqnarray`.

### 3.5.3 Typowe użycie

Poniżej opiszemy jak używać `IEEEeqnarray` do rozwiązywania najczęściej występujących problemów.

Jeśli linia przykrywa numer równania tak jak w przykładzie (3.17), to polecenie

```
\IEEEeqnarraynumspace
```

może zostać użyte; musi ono zostać dodane do odpowiedniej linii i upewnia się ono, że cała tablica równań przesunięta jest odpowiednio by zmieścić numer równania (wartość przesunięcia zależy oczywiście od numeru!): zamiast

```
\begin{IEEEeqnarray}{rCl}
a & = & b + c \\
\\
& = & d + e + f + g + h \\
& + & i + j + k \\
\\
& = & l + m + n \\
\end{IEEEeqnarray}
```

$$a = b + c \quad (3.24)$$

$$= d + e + f + g + h + i + j + k \quad (3.25)$$

$$= l + m + n \quad (3.26)$$

otrzymujemy

<sup>10</sup>Więcej informacji o rodzajach odstępu znajdziesz w sekcji 3.9.1.

```
\begin{IEEEeqnarray}{rCl}
  a & = & b + c \\
  \\
  & = & d + e + f + g + h \\
  & + & i + j + k \\
  \IEEEeqnarraynumspace \\
  & = & l + m + n.
\end{IEEEeqnarray}
```

$$a = b + c \quad (3.27)$$

$$= d + e + f + g + h + i + j + k \quad (3.28)$$

$$= l + m + n. \quad (3.29)$$

Jeśli lewa strona jest zbyt długa, zamiast używać wadliwego `\lefteqn`, `IEEEeqnarray` dostarcza polecenie `\IEEEeqnarraymulticol`, które działa w każdej sytuacji:

```
\begin{IEEEeqnarray}{rCl}
  \IEEEeqnarraymulticol{3}{1}{
    a + b + c + d + e + f \\
    + g + h \\
  }\nonumber \\
  & = & i + j \\
  \\
  & = & k + l + m
\end{IEEEeqnarray}
```

$$a + b + c + d + e + f + g + h$$

$$= i + j \quad (3.30)$$

$$= k + l + m \quad (3.31)$$

Jego użycie jest identyczne do polecenia `\multicolumns` w otoczeniu tabular. Pierwszym argumentem, `{3}`, określa, że trzy kolumny powinny być połączone w jedną, która będzie wyrównana do lewej, `{1}`.

Zwróć uwagę, że wstawiając polecenia `\quad` można łatwo dopasować głębokość znaków równości<sup>11</sup>, przykładowo

```
\begin{IEEEeqnarray}{rCl}
  \IEEEeqnarraymulticol{3}{1}{
    a + b + c + d + e + f \\
    + g + h \\
  }\nonumber \\
  & = & i + j \\
  \\
  & = & k + l + m
\end{IEEEeqnarray}
```

$$a + b + c + d + e + f + g + h$$

$$= i + j \quad (3.32)$$

$$= k + l + m \quad (3.33)$$

Jeśli równanie podzielone jest na jedną lub więcej linii, `LATEX` interpretuje pierwszy `+` lub `-` jako znak zamiast operator. Skutkiem tego jest konieczność dodania pustej grupy `{}` przed operatorem: zamiast

<sup>11</sup>W opinii autora jeden `\quad` najczęściej wygląda najlepiej.

```
\begin{IEEEeqnarray}{rCl}
a & = & b + c \\
\\
& = & d + e + f + g + h \\
& + & i + j + k \nonumber \\
& & + l + m + n + o \\
\\
& = & p + q + r + s \\
\end{IEEEeqnarray}
```

$$a = b + c \quad (3.34)$$

$$= d + e + f + g + h + i + j + k \\ + l + m + n + o \quad (3.35)$$

$$= p + q + r + s \quad (3.36)$$

powinniśmy napisać

```
\begin{IEEEeqnarray}{rCl}
a & = & b + c \\
\\
& = & d + e + f + g + h \\
& + & i + j + k \nonumber \\
& & \negmedspace {} + l \\
& + & m + n + o \\
\\
& = & p + q + r + s \\
\end{IEEEeqnarray}
```

$$a = b + c \quad (3.37)$$

$$= d + e + f + g + h + i + j + k \\ + l + m + n + o \quad (3.38)$$

$$= p + q + r + s \quad (3.39)$$

Zwróć uwagę na różne odstępy pomiędzy + i ! Konstrukcja `{}` + `l` zmusza by znak + był traktowany jako operator, a nie jako znak liczby, z kolei niechciany odstęp pomiędzy `{}` i + jest uniknięta dzięki negatywnemu odstępowi średniemu `\negmedspace`.

Jeśli któraś linia równania nie powinna być numerowana, możemy temu zapobiec przy użyciu polecenia `\nonumber` (lub `\IEEEnonumber`). Jeśli na takiej linii zdefiniowana jest etykieta `\label{eq:...}`, to będzie ona odnosić się do następnego równania, którego numer nie jest wyłączony. Etykiety warto wstawiać tuż przed złamaniem linii `\\` lub zaraz po równaniu do których należą. Oprócz lepszej czytelności kodu źródłowego, zapobiega to również błędowi kompilacji gdy `\IEEEmulticol` następuje po definicji etykiety.

Istnieje również wersja z gwiazdką, w której wszystkie numery równań są wyłączone. W takim przypadku możemy dodać numer równania przy użyciu polecenia `\IEEEyesnumber`:

```
\begin{IEEEeqnarray*}{rCl}
a & = & b + c \\
& = & d + e \IEEEyesnumber \\
& = & f + g \\
\end{IEEEeqnarray*}
```

$$a = b + c \\ = d + e \quad (3.40) \\ = f + g$$

Używając `\IEEEyessubnumber` możliwe jest również definiowanie podnumerów równań:

```
\begin{IEEEeqnarray}{rCl}
a & = & b + c
\IEEEyessubnumber\
& = & d + e
\nonumber\
& = & f + g
\IEEEyessubnumber
\end{IEEEeqnarray}
```

$$a = b + c \quad (3.40a)$$

$$= d + e$$

$$= f + g \quad (3.40b)$$

### 3.6 Tablice i macierze

Do składania **tablic** użyj otoczenia `array`. Działa ono podobnie do otoczenia `tabular`. Polecenie `\` jest używane do łamania wierszy:

```
\begin{equation*}
\mathbf{X} = \left(
\begin{array}{ccc}
x_1 & x_2 & \dots \\
x_3 & x_4 & \dots \\
\vdots & \vdots & \ddots
\end{array}
\right)
\end{equation*}
```

$$\mathbf{X} = \begin{pmatrix} x_1 & x_2 & \dots \\ x_3 & x_4 & \dots \\ \vdots & \vdots & \ddots \end{pmatrix}$$

Otoczenie `array` może być też użyte do składania funkcji częściowych używając „.” jako niewidzialny prawy ogranicznik `\right.`:

```
\begin{equation*}
|x| = \left\{
\begin{array}{rl}
-x & \text{dla } x < 0, \\
0 & \text{dla } x = 0, \\
x & \text{dla } x > 0.
\end{array}
\right.
\end{equation*}
```

$$|x| = \begin{cases} -x & \text{dla } x < 0, \\ 0 & \text{dla } x = 0, \\ x & \text{dla } x > 0. \end{cases}$$

Otoczenie `cases` z `amsmath` upraszcza nieco składnię, więc jemu również warto się przyjrzeć:

```

\begin{equation*}
|x| =
\begin{cases}
-x & \text{dla } x < 0, \\
0 & \text{dla } x = 0, \\
x & \text{dla } x > 0.
\end{cases}
\end{equation*}

```

$$|x| = \begin{cases} -x & \text{dla } x < 0, \\ 0 & \text{dla } x = 0, \\ x & \text{dla } x > 0. \end{cases}$$

Macierze mogą być również składane przy użyciu `array`, ale `amsmath` umożliwia lepsze rozwiązania przy użyciu otoczeń `matrix`. Jest sześć wersji z innymi ogranicznikami: `matrix` (brak), `pmatrix` (, ), `bmatrix` [, ], `Bmatrix` {, }, `vmatrix` | oraz `Vmatrix` ||. Nie musisz określać liczby kolumn tak jak w `array`. Maksymalna liczba kolumn to 10, jednak można to zmienić (czego prawdopodobnie nie będziesz potrzebować!):

```

\begin{equation*}
\begin{matrix}
1 & 2 \\
3 & 4
\end{matrix} \quad \quad \quad
\begin{bmatrix}
p_{11} & p_{12} & \dots & p_{1n} \\
& p_{21} & & \\
p_{21} & p_{22} & \dots & p_{2n} \\
& p_{2n} & & \\
\vdots & \vdots & \ddots & \\
& \vdots & & \\
p_{m1} & p_{m2} & \dots & \\
& p_{mn} & & 
\end{bmatrix}
\end{equation*}

```

$$\begin{matrix} 1 & 2 \\ 3 & 4 \end{matrix} \quad \begin{bmatrix} p_{11} & p_{12} & \dots & p_{1n} \\ p_{21} & p_{22} & \dots & p_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ p_{m1} & p_{m2} & \dots & p_{mn} \end{bmatrix}$$

### 3.7 Odstępy w trybie matematycznym

Zdarzają się sytuacje, kiedy wielkość odstępów wewnątrz wyrażeń matematycznych jest niepoprawna. Można je skorygować odpowiednimi instrukcjami. Do wprowadzania niewielkich odstępów służy kilka poleceń: `\`, wstawia odstęp równy  $\frac{3}{18}$  em (u), `\:` pozwala uzyskać odstęp równy  $\frac{4}{18}$  em (u) a `\;` – odstęp równy  $\frac{5}{18}$  em (u). Użycie instrukcji `\_` (spacja po znaku `\`) prowadzi do utworzenia zwykłego odstępu międzywyrazowego; `\quad` – odstępu

równego 1 em ( $\_$ ), a  $\quad$  – dwóm em ( $\_$ ). Rozmiar  $\quad$  jest równy szerokości litery „M” aktualnego fontu.  $\!$  wstawia ujemny odstęp równy  $-\frac{3}{18}$  em ( $\_$ ).

```
\begin{equation*}
\int_1^2 \ln x \mathrm{d}x
\quad
\int_1^2 \ln x \ , \mathrm{d}x
\end{equation*}
```

$$\int_1^2 \ln x dx \quad \int_1^2 \ln x dx$$

Zwróć uwagę, że „d” w równaniach różniczkowych jest zwyczajowo zapisywane normalnym fontem. W następnych przykładach definiujemy nowe polecenie  $\ud$ , które tworzy „d” (zwróć uwagę na odstęp  $\_$  przed d), dzięki czemu nie musimy pisać go za każdym razem. Polecenie  $\newcommand$  może zostać wstawione w preambule.

```
\newcommand{\ud}{\ , \mathrm{d}}
```

```
\begin{equation*}
\int_a^b f(x)\ud x
\end{equation*}
```

$$\int_a^b f(x) dx$$

Jeśli spróbujesz złożyć kilka całek to odkryjesz, że odstęp między nimi są zbyt duże. Możesz je poprawić przy użyciu  $\!$ , ale  $\amsmath$  zapewnia prostszy sposób na dopasowanie odstępów, a mianowicie polecenia  $\iint$ ,  $\iiint$ ,  $\dotsint$ .

```
\newcommand{\ud}{\ , \mathrm{d}}
```

```
\begin{IEEEeqnarray*}{c}
\int\int f(x)g(y)
\ud x \ud y \ \
\int\!\!\!\!\int
f(x)g(y) \ud x \ud y \ \
\iint f(x)g(y) \ud x \ud y
\end{IEEEeqnarray*}
```

$$\int \int f(x)g(y) dx dy$$

$$\iint f(x)g(y) dx dy$$

$$\iint f(x)g(y) dx dy$$

Więcej informacji znajdziesz w `testmath.tex` (udostępnianym w  $\mathcal{A}\mathcal{M}\mathcal{S}$ - $\mathcal{L}\mathcal{T}\mathcal{E}\mathcal{X}$ ) lub rozdziale ósmym *The L<sup>A</sup>T<sub>E</sub>X Companion* [3].

### 3.7.1 Fantomy

Podczas wyrównywania w pionie tekstu z indeksami  $\hat{\phantom{x}}$  bądź  $\_ \mathcal{L}\mathcal{T}\mathcal{E}\mathcal{X}$  bywa nadgorliwy. Polecenie  $\phantom$  pozwala rezerwować miejsce na znaki, któ-



re nie mają się pojawić w ostatecznym wydruku. Najłatwiej to zrozumieć, analizując przykład:

```
\begin{equation*}
{}^{\text{14}}_6\text{C}
\quad \text{kontra} \quad
{}^{\text{14}}_6\text{C}
\end{equation*}
```

$${}^{14}_6\text{C} \quad \text{kontra} \quad {}^{14}_6\text{C}$$

W przypadku gdy spodziewasz się pisania sporej liczby izotopów warto przyjrzeć się pakietowi `mhchem`, który znacząco ułatwia składania zarówno ich jak i formuł chemicznych.

### 3.8 Zabawa z fontami matematycznymi

Różne fonty matematyczne są zestawione w tabeli 3.14 na stronie 82.

```
\Re \quad \quad \quad
\mathcal{R} \quad \quad \quad
\mathfrak{R} \quad \quad \quad
\mathbb{R} \quad \quad \quad
```

$$\Re \quad \mathcal{R} \quad \mathfrak{R} \quad \mathbb{R}$$

Ostatnie dwa przykłady wymagają `amssymb` lub `amsfonts`.

Czasami musisz poinformować `LATEX`a, który rozmiar fontu chcesz użyć. W trybie matematycznym służą do tego następujące cztery polecenia:

```
\displaystyle (123), \textstyle (123), \scriptstyle (123) and
\scriptscriptstyle (123).
```

Jeśli  $\sum$  jest składana wewnątrz ułamka, będzie ona złożona w stylu tekstowym chyba, że poinstruujesz `LATEX`a inaczej:

```
\begin{equation*}
P = \frac{\displaystyle
\sum_{i=1}^n (x_i - x)
(y_i - y)}
{\displaystyle\left[
\sum_{i=1}^n (x_i - x)^2
\sum_{i=1}^n (y_i - y)^2
\right]^{1/2}}
\end{equation*}
```

$$P = \frac{\sum_{i=1}^n (x_i - x)(y_i - y)}{\left[ \sum_{i=1}^n (x_i - x)^2 \sum_{i=1}^n (y_i - y)^2 \right]^{1/2}}$$

Zmiana stylów znacząco wpływa na składanie dużych operatorów i ich limitów.

### 3.8.1 Pogrubione symbole

W L<sup>A</sup>T<sub>E</sub>Xu trudno jest uzyskać symbole pogrubione; jest to celowy zabieg, ponieważ amatorzy często ich nadużywają. Polecenie zmiany fontu `\mathbf` sprawia, że litery są pogrubione, jednak są one niepochylone, podczas gdy symbole matematyczne są zwykle zapisane kursywą. Co więcej nie działa ono dla małych liter greckich. Istnieje polecenie `\boldmath`, ale może *być ono wykorzystywane tylko poza trybem matematycznym*. Działa wtedy również dla symboli:

```
$\mu, M \quad \quad \quad
\mathbf{\mu}, \mathbf{M}$
\quad \boldmath{\mu}, M$
```

$\mu, M$	$\mu, \mathbf{M}$	$\boldsymbol{\mu}, M$
----------	-------------------	-----------------------

Pakiet `amsbsy` (dołączany do `amsmath`) jak również pakiet `package bm` z kolekcji `tools` znacząco to ułatwiają, ponieważ zawierają one polecenie `\boldsymbol`:

```
$\mu, M \quad \quad \quad
\boldsymbol{\mu}, \boldsymbol{M}$
```

$\mu, M$	$\boldsymbol{\mu}, M$
----------	-----------------------

## 3.9 Twierdzenia, lematy, itp.

W pracach matematycznych występuje potrzeba wyróżniania w składzie zapisu „lematów”, „definicji”, „aksjomatów” i tym podobnych elementów. Do zdefiniowania nowego typu elementu służy polecenie:

<code>\newtheorem{nazwa}[licznik]{tekst}[sekcja]</code>
---------------------------------------------------------

Argument *nazwa* jest krótkim słowem wykorzystywanym do rozpoznawania „twierdzenia”. W argumentie *tekst* zdefiniowana jest faktyczna nazwa, która będzie ostatecznie wydrukowana.

Argumenty w nawiasach kwadratowych są opcjonalne. Oba są wykorzystywane do określania w jaki sposób „twierdzenia” powinny być numerowane. Użyj argumentu *licznik* aby określić *nazwę* poprzednio zdefiniowanego „twierdzenia”. Nowe „twierdzenie” będzie wtedy numerowane kolejno z nim. Argument *sekcja* pozwala na określenie w której części hierarchii dokumentu twierdzenia powinny otrzymywać swój numer.

Po dodaniu polecenia `\newtheorem` do preambuły, możesz korzystać z następujących poleceń wewnątrz dokumentu.

```
\begin{name}[text]
Oto moje interesujące twierdzenie
\end{name}
```

W pakiecie `amsthm` (część `AMS-LATEX`) dostępne jest również polecenie `\theoremstyle{styl}` które pozwala określać jeden z trzech dostępnych stylów dla twierdzenia: `definition` (definicje, pogrubiony tytuł, zwykła treść), `plain` (pogrubiony tytuł, treść zapisana kursywą) lub `remark` (tytuł kursywą, zwykła treść).

Wystarczy teorii. Następujące przykłady powinny rozwiązać wszystkie wątpliwości i zasygnalizować, że otoczenie `\newtheorem` jest możliwe do zrozumienia.

Najpierw zdefiniujmy twierdzenia:

```
\theoremstyle{definition} \newtheorem{prawo}{Prawo}
\theoremstyle{plain}      \newtheorem{sąd}{Sąd}
\theoremstyle{remark}     \newtheorem*{małg}{Małgorzata}
```

```
\begin{prawo} \label{prawo:chow}
Świadkowie nie mogą się chować
\end{prawo}
\begin{sąd}[Dwunastu]
To mogłaś być ty! Więc uważaj
i przestrzegaj
prawa~\ref{prawo:chow}.\end{sąd}
\begin{sąd}
Należy zignorować poprzednie
zdanie.\end{sąd}
\begin{małg}Nie, Nie, Nie\end{małg}
\begin{małg}Denis!\end{małg}
```

**Prawo 1.** Świadkowie nie mogą się chować

**Sąd 2** (Dwunastu). *To mogłaś być ty! Więc uważaj i przestrzegaj prawa 1.*

**Sąd 3.** *Należy zignorować poprzednie zdanie.*

*Małgorzata.* Nie, Nie, Nie

*Małgorzata.* Denis!

Twierdzenie „Sąd” korzysta z tego samego licznika co „Prawo”, więc dostaje numer kolejnego „Prawa”. Argument w nawiasach kwadratowych może zawierać tytuł lub coś podobnego dla danej teorii.

```
\newtheorem{mur}{Murphy}[section]
```

```
\begin{mur} Jeżeli coś można
wykonać na dwa lub więcej
sposobów, przy czym jeden
z nich prowadzi do katastrofy,
to sposób ten zostanie
przez kogoś wybrany.\end{mur}
```

*Murphy 3.9.1.* Jeżeli coś można wykonać na dwa lub więcej sposobów, przy czym jeden z nich prowadzi do katastrofy, to sposób ten zostanie przez kogoś wybrany.

Numeracja twierdzenia „Murphy’ego” jest tu powiązana z numeracją kolejnych punktów. Można też do numerowania twierdzeń stosować inne jednostki podziału dokumentu, jak rozdziały czy podsekcje.

Jeśli chcesz więcej kontroli nad wyglądem „twierdzeń”, możesz przyrzeć się pakietowi `ntheorem`, który oferuje wiele opcji.

### 3.9.1 Dowody i symbol końca dowodu

W pakiecie `amsthm` znajduje się też otoczenie `proof` do zapisywania dowodów

```
\begin{proof}
  Trywialne, użyj
  \begin{equation*}
    E=mc^2.
  \end{equation*}
\end{proof}
```

*Dowód.* Trywialne, użyj

$$E = mc^2.$$

□

Polecenie `\qedhere` pozwala wstawić symbol „końca dowodu” w określonym miejscu zamiast domyślnego umieszczenia go w oddzielnym wierszu:

```
\begin{proof}
  Trywialne, użyj
  \begin{equation*}
    E=mc^2. \qedhere
  \end{equation*}
\end{proof}
```

*Dowód.* Trywialne, użyj

$$E = mc^2.$$

□

Niestety korekcja ta nie działa w `IEEEeqnarray`:

```
\begin{proof}
  Oto dowód, który kończy
  się tablicą równań:
  \begin{IEEEeqnarray*}{rCl}
    a & = & b + c \\
    & & d + e. \qedhere
  \end{IEEEeqnarray*}
\end{proof}
```

*Dowód.* Oto dowód, który kończy się tablicą równań:

$$a = b + c$$

$$= d + e. \quad \square$$

Powodem takiego zachowania jest wewnętrzna struktura `IEEEeqnarray`: zawsze wstawia ono dwie niewidzialne kolumny po obu stronach tablicy, które zawierają rozciągliwy odstęp. W ten sposób `IEEEeqnarray` gwarantuje, że

równania są wyśrodkowane. Polecenie `\qedhere` powinno być wstawione *poza* tymi rozciągliwymi odstępami, jednak tak się nie dzieje ponieważ kolumny te są niewidzialne dla użytkownika.

Istnieje bardzo proste rozwiązanie. Zdefiniować rozciągliwe odstęp samodzielnie!

```
\begin{proof}
  Oto dowód, który kończy
  się tablicą równań:
  \begin{IEEEeqnarray*}{+rCl+x*}
    a & = & b + c \\
    & = & d + e. & \qedhere
  \end{IEEEeqnarray*}
\end{proof}
```

*Dowód.* Oto dowód, który kończy się tablicą równań:

$$\begin{aligned} a &= b + c \\ &= d + e. \quad \square \end{aligned}$$

Zwróć uwagę, że `+` w `{+rCl+x*}` oznacza rozciągliwe odstęp, jeden po lewej stronie równań (który w przeciwnym przypadku byłby wstawiony automatycznie przez `IEEEeqnarray`!) oraz jeden z prawej strony równań. Jednak teraz po prawej stronie, *po* rozciągliwej kolumnie, dodajemy pustą kolumnę `x`. Ta kolumna będzie potrzebna tylko w ostatnim wierszu jeśli polecenie `\qedhere` zostanie tam wstawione. Wreszcie wstawiamy `*`. Oznacza ono pusty odstęp, który zapobiega przed automatycznym wstawieniem przez `IEEEeqnarray` kolejnej niechcianej kolumny z odstępem!

W przypadku numerowanych równań występuje podobny problem. Porównując

```
\begin{proof}
  Oto dowód, który kończy
  się numerowanym równaniem:
  \begin{equation}
    a = b + c.
  \end{equation}
\end{proof}
```

*Dowód.* Oto dowód, który kończy się numerowanym równaniem:

$$a = b + c. \quad (3.41)$$

□

z

```
\begin{proof}
  Oto dowód, który kończy
  się numerowanym równaniem:
  \begin{equation}
    a = b + c. \quad \qedhere
  \end{equation}
\end{proof}
```

*Dowód.* Oto dowód, który kończy się numerowanym równaniem:

$$a = b + c. \quad (3.42)$$

□

zauważysz, że w drugiej (poprawnej) wersji  $\square$  jest dużo bliżej równania niż w pierwszej.

Podobnie poprawne miejsce wstawiania symbolu końca dowodu na końcu tablicy równań uzyskiwane jest następująco:

```
\begin{proof}
Oto dowód, który kończy
się tablicą równań:
\begin{IEEEeqnarray}{+rCl+x*}
a & = & b + c \\
& & d + e. \\
& & \qedhere\nonumber
\end{IEEEeqnarray}
\end{proof}
```

*Dowód.* Oto dowód, który kończy się tablicą równań:

$$a = b + c \quad (3.43)$$

$$= d + e. \quad (3.44)$$

$\square$

co wygląda lepiej niż:

```
\begin{proof}
Oto dowód, który kończy
się tablicą równań:
\begin{IEEEeqnarray}{rCl}
a & = & b + c \\
& & d + e.
\end{IEEEeqnarray}
\end{proof}
```

*Dowód.* Oto dowód, który kończy się tablicą równań:

$$a = b + c \quad (3.45)$$

$$= d + e. \quad (3.46)$$

$\square$

### 3.10 Zestawienie symboli matematycznych

W poniższych tabelach zestawiono wszystkie symbole standardowo dostępne w trybie *matematycznym*.

Zwróć uwagę, że niektóre tabele są dostępne tylko po załadowaniu pakietu `amssymb` w preambule dokumentu<sup>12</sup>. W razie braku fontów lub pakietu można je odnaleźć w `CTAN:pkg/amslatex`. Ponadto dużo bardziej kompletne zestawienie symboli matematycznych można znaleźć w `CTAN:info/symbols/comprehensive`.

Tabela 3.1: Akcenty w trybie matematycznym.

$\hat{a}$	<code>\hat{a}</code>	$\check{a}$	<code>\check{a}</code>	$\tilde{a}$	<code>\tilde{a}</code>
$\grave{a}$	<code>\grave{a}</code>	$\dot{a}$	<code>\dot{a}</code>	$\ddot{a}$	<code>\ddot{a}</code>
$\bar{a}$	<code>\bar{a}</code>	$\vec{a}$	<code>\vec{a}</code>	$\widehat{AAA}$	<code>\widehat{AAA}</code>
$\acute{a}$	<code>\acute{a}</code>	$\breve{a}$	<code>\breve{a}</code>	$\widetilde{AAA}$	<code>\widetilde{AAA}</code>
$\mathring{a}$	<code>\mathring{a}</code>				

Tabela 3.2: Litery greckie.

Nie znajdziesz tu odpowiedników wielkich liter takich jak `\Alpha`, `\Beta` itp., ponieważ wyglądają one identycznie do łacińskich liter A, B...

$\alpha$	<code>\alpha</code>	$\theta$	<code>\theta</code>	$o$	<code>o</code>	$v$	<code>\upsilon</code>
$\beta$	<code>\beta</code>	$\vartheta$	<code>\vartheta</code>	$\pi$	<code>\pi</code>	$\phi$	<code>\phi</code>
$\gamma$	<code>\gamma</code>	$\iota$	<code>\iota</code>	$\varpi$	<code>\varpi</code>	$\varphi$	<code>\varphi</code>
$\delta$	<code>\delta</code>	$\kappa$	<code>\kappa</code>	$\rho$	<code>\rho</code>	$\chi$	<code>\chi</code>
$\epsilon$	<code>\epsilon</code>	$\lambda$	<code>\lambda</code>	$\varrho$	<code>\varrho</code>	$\psi$	<code>\psi</code>
$\varepsilon$	<code>\varepsilon</code>	$\mu$	<code>\mu</code>	$\sigma$	<code>\sigma</code>	$\omega$	<code>\omega</code>
$\zeta$	<code>\zeta</code>	$\nu$	<code>\nu</code>	$\varsigma$	<code>\varsigma</code>		
$\eta$	<code>\eta</code>	$\xi$	<code>\xi</code>	$\tau$	<code>\tau</code>		
$\Gamma$	<code>\Gamma</code>	$\Lambda$	<code>\Lambda</code>	$\Sigma$	<code>\Sigma</code>	$\Psi$	<code>\Psi</code>
$\Delta$	<code>\Delta</code>	$\Xi$	<code>\Xi</code>	$\Upsilon$	<code>\Upsilon</code>	$\Omega$	<code>\Omega</code>
$\Theta$	<code>\Theta</code>	$\Pi$	<code>\Pi</code>	$\Phi$	<code>\Phi</code>		

<sup>12</sup>Tabele przygotowano na podstawie pliku `symbols.tex` autorstwa David Carlisle, gruntownie zmodyfikowanego zgodnie z sugestiami Josefa Tkadleca.

Tabela 3.3: Relacje binarne.

Aby zanegować dany symbol relacji należy wstawić przed nimi polecenie `\not`.

$<$	<code>&lt;</code>	$>$	<code>&gt;</code>	$=$	<code>=</code>
$\leq$	<code>\leq</code> or <code>\le</code>	$\geq$	<code>\geq</code> or <code>\ge</code>	$\equiv$	<code>\equiv</code>
$\ll$	<code>\ll</code>	$\gg$	<code>\gg</code>	$\doteq$	<code>\doteq</code>
$\prec$	<code>\prec</code>	$\succ$	<code>\succ</code>	$\sim$	<code>\sim</code>
$\preceq$	<code>\preceq</code>	$\succeq$	<code>\succeq</code>	$\simeq$	<code>\simeq</code>
$\subset$	<code>\subset</code>	$\supset$	<code>\supset</code>	$\approx$	<code>\approx</code>
$\subseteq$	<code>\subseteq</code>	$\supseteq$	<code>\supseteq</code>	$\cong$	<code>\cong</code>
$\sqsubset^a$	<code>\sqsubset^a</code>	$\sqsupset^a$	<code>\sqsupset^a</code>	$\bowtie^a$	<code>\bowtie^a</code>
$\sqsubseteq$	<code>\sqsubseteq</code>	$\sqsupseteq$	<code>\sqsupseteq</code>	$\bowtie$	<code>\bowtie</code>
$\in$	<code>\in</code>	$\ni$	<code>\ni</code> , <code>\owns</code>	$\propto$	<code>\propto</code>
$\vdash$	<code>\vdash</code>	$\dashv$	<code>\dashv</code>	$\models$	<code>\models</code>
$\mid$	<code>\mid</code>	$\parallel$	<code>\parallel</code>	$\perp$	<code>\perp</code>
$\smile$	<code>\smile</code>	$\frown$	<code>\frown</code>	$\asymp$	<code>\asymp</code>
$:$	<code>:</code>	$\notin$	<code>\notin</code>	$\neq$	<code>\neq</code> or <code>\ne</code>

<sup>a</sup>Użyj pakietu `latexsym` by mieć dostęp do tego symbolu

Tabela 3.4: Operatory binarne.

$+$	<code>+</code>	$-$	<code>-</code>	$\triangleleft$	<code>\triangleleft</code>
$\pm$	<code>\pm</code>	$\mp$	<code>\mp</code>	$\triangleangleright$	<code>\triangleangleright</code>
$\cdot$	<code>\cdot</code>	$\div$	<code>\div</code>	$\star$	<code>\star</code>
$\times$	<code>\times</code>	$\setminus$	<code>\setminus</code>	$\ast$	<code>\ast</code>
$\cup$	<code>\cup</code>	$\cap$	<code>\cap</code>	$\circ$	<code>\circ</code>
$\sqcup$	<code>\sqcup</code>	$\sqcap$	<code>\sqcap</code>	$\bullet$	<code>\bullet</code>
$\vee$	<code>\vee</code> , <code>\lor</code>	$\wedge$	<code>\wedge</code> , <code>\land</code>	$\diamond$	<code>\diamond</code>
$\oplus$	<code>\oplus</code>	$\ominus$	<code>\ominus</code>	$\uplus$	<code>\uplus</code>
$\odot$	<code>\odot</code>	$\oslash$	<code>\oslash</code>	$\amalg$	<code>\amalg</code>
$\otimes$	<code>\otimes</code>	$\bigcirc$	<code>\bigcirc</code>	$\dagger$	<code>\dagger</code>
$\triangle$	<code>\bigtriangleup</code>	$\nabla$	<code>\bigtriangledown</code>	$\ddagger$	<code>\ddagger</code>
$\triangleleft$	<code>\lhd^a</code>	$\triangleright$	<code>\rhd^a</code>	$\wr$	<code>\wr</code>
$\triangleleft$	<code>\unlhd^a</code>	$\triangleright$	<code>\unrhd^a</code>		



Tabela 3.5: DUŻE operatory.

$\Sigma$	<code>\sum</code>	$\cup$	<code>\bigcup</code>	$\vee$	<code>\bigvee</code>
$\prod$	<code>\prod</code>	$\cap$	<code>\bigcap</code>	$\wedge$	<code>\bigwedge</code>
$\coprod$	<code>\coprod</code>	$\sqcup$	<code>\bigsqcup</code>	$\oplus$	<code>\bigoplus</code>
$\int$	<code>\int</code>	$\oint$	<code>\oint</code>	$\odot$	<code>\bigodot</code>
$\oplus$	<code>\bigoplus</code>	$\otimes$	<code>\bigotimes</code>		

Tabela 3.6: Strzałki.

$\leftarrow$	<code>\leftarrow</code> or <code>\gets</code>	$\longleftarrow$	<code>\longleftarrow</code>
$\rightarrow$	<code>\rightarrow</code> or <code>\to</code>	$\longrightarrow$	<code>\longrightarrow</code>
$\leftrightarrow$	<code>\leftrightarrow</code>	$\longleftrightarrow$	<code>\longleftrightarrow</code>
$\Leftarrow$	<code>\Leftarrow</code>	$\Lleftarrow$	<code>\Lleftarrow</code>
$\Rightarrow$	<code>\Rightarrow</code>	$\Rrightarrow$	<code>\Rrightarrow</code>
$\Leftrightarrow$	<code>\Leftrightarrow</code>	$\Llongleftrightarrow$	<code>\Llongleftrightarrow</code>
$\mapsto$	<code>\mapsto</code>	$\longmapsto$	<code>\longmapsto</code>
$\hookrightarrow$	<code>\hookrightarrow</code>	$\hookleftarrow$	<code>\hookleftarrow</code>
$\lleftarrow$	<code>\lleftarrow</code>	$\rightharpoonup$	<code>\rightharpoonup</code>
$\lrightarrow$	<code>\lrightarrow</code>	$\rightharpoondown$	<code>\rightharpoondown</code>
$\rightleftharpoons$	<code>\rightleftharpoons</code>	$\iff$	<code>\iff</code> (bigger spaces)
$\uparrow$	<code>\uparrow</code>	$\downarrow$	<code>\downarrow</code>
$\updownarrow$	<code>\updownarrow</code>	$\Uparrow$	<code>\Uparrow</code>
$\Downarrow$	<code>\Downarrow</code>	$\Updownarrow$	<code>\Updownarrow</code>
$\nearrow$	<code>\nearrow</code>	$\searrow$	<code>\searrow</code>
$\swarrow$	<code>\swarrow</code>	$\nwarrow$	<code>\nwarrow</code>
$\rightsquigarrow$	<code>\rightsquigarrow</code> <sup>a</sup>		

<sup>a</sup>Użyj pakietu `latexsym` by mieć dostęp do tego symbolu

Tabela 3.7: Strzałki i akcenty.

$\overrightarrow{AB}$	<code>\overrightarrow{AB}</code>	$\underline{\overrightarrow{AB}}$	<code>\underrightarrow{AB}</code>
$\overleftarrow{AB}$	<code>\overleftarrow{AB}</code>	$\underline{\overleftarrow{AB}}$	<code>\underleftarrow{AB}</code>
$\overleftrightarrow{AB}$	<code>\overleftrightarrow{AB}</code>	$\underline{\overleftrightarrow{AB}}$	<code>\underleftrightarrow{AB}</code>

Tabela 3.8: Ograniczniki.

(	(	)	)	↑	\uparrow
[	[ or \lbrack	]	] or \rbrack	↓	\downarrow
{	\{ or \lbrace	}	\} or \rbrace	↕	\updownarrow
⟨	\langle	⟩	\rangle	⇑	\Uparrow
	or \vert		\  or \Vert	⇓	\Downarrow
/	/	\	\backslash	↕	\Updownarrow
⌊	\lfloor	⌋	\rfloor		
⌈	\rceil	⌉	\lceil		

Tabela 3.9: Duże ograniczniki.

{	\lgroup	}	\rgroup	⎵	\lmoustache
	\arrowvert		\Arrowvert		\bracevert
}	\rmoustache				

Tabela 3.10: Różne symbole.

...	\dots	...	\cdots	:	\vdots	⋯	\ddots
$\hbar$	\hbar	$\imath$	\imath	$\jmath$	\jmath	$\ell$	\ell
$\Re$	\Re	$\Im$	\Im	$\aleph$	\aleph	$\wp$	\wp
$\forall$	\forall	$\exists$	\exists	$\mho$	\mho <sup>a</sup>	$\partial$	\partial
'	'	'	\prime	$\emptyset$	\emptyset	$\infty$	\infty
$\nabla$	\nabla	$\triangle$	\triangle	$\square$	\Box <sup>a</sup>	$\diamond$	\Diamond <sup>a</sup>
$\perp$	\bot	$\top$	\top	$\angle$	\angle	$\surd$	\surd
$\diamond$	\diamondsuit	$\heartsuit$	\heartsuit	$\clubsuit$	\clubsuit	$\spadesuit$	\spadesuit
$\neg$	\neg or \lnot	$\flat$	\flat	$\natural$	\natural	$\sharp$	\sharp

<sup>a</sup>Użyj pakietu latexsym by mieć dostęp do tego symbolu

Tabela 3.11: Symbole niematematyczne.

Polecenia te są dostępne również w trybie tekstowym.

†	<code>\dag</code>	§	<code>\S</code>	©	<code>\copyright</code>	®	<code>\textregistered</code>
‡	<code>\ddag</code>	¶	<code>\P</code>	£	<code>\pounds</code>	%	<code>\%</code>

Tabela 3.12: Ograniczniki w  $\mathcal{AMS}$ .

$\ulcorner$	<code>\ulcorner</code>	$\urcorner$	<code>\urcorner</code>	$\llcorner$	<code>\llcorner</code>	$\lrcorner$	<code>\lrcorner</code>
$\lvert$	<code>\lvert</code>	$\rvert$	<code>\rvert</code>	$\lVert$	<code>\lVert</code>	$\rVert$	<code>\rVert</code>

Tabela 3.13: Litery greckie i hebrajskie w  $\mathcal{AMS}$ .

$\digamma$	<code>\digamma</code>	$\varkappa$	<code>\varkappa</code>	$\beth$	<code>\beth</code>	$\gimel$	<code>\gimel</code>	$\daleth$	<code>\daleth</code>
------------	-----------------------	-------------	------------------------	---------	--------------------	----------	---------------------	-----------	----------------------

Tabela 3.14: Alfabet matematyczne.

W tabeli 6.4 na stronie 130 pokazane są inne fonty matematyczne.

Przykład	Polecenie	Wymagany pakiet
$\mathrm{ABCDEabcde1234}$	<code>\mathrm{ABCDE abcde 1234}</code>	
$\mathit{ABCDEabcde1234}$	<code>\mathit{ABCDE abcde 1234}</code>	
$\mathnormal{ABCDEabcde1234}$	<code>\mathnormal{ABCDE abcde 1234}</code>	
$\mathcal{ABCDE}$	<code>\mathcal{ABCDE abcde 1234}</code>	
$\mathscr{ABCDE}$	<code>\mathscr{ABCDE abcde 1234}</code>	<code>mathrsfs</code>
$\mathfrak{ABCDEabcde1234}$	<code>\mathfrak{ABCDE abcde 1234}</code>	<code>amsfonts</code> lub <code>amssymb</code>
$\mathbb{ABCDE\Omega\#\#\#}$	<code>\mathbb{ABCDE abcde 1234}</code>	<code>amsfonts</code> lub <code>amssymb</code>

Tabela 3.15: Operatory binarne w  $\mathcal{AMS}$ .

$\dot{+}$	<code>\dotplus</code>	$\cdot$	<code>\centerdot</code>		
$\ltimes$	<code>\ltimes</code>	$\rtimes$	<code>\rtimes</code>	$\div$	<code>\divideontimes</code>
$\cup$	<code>\doublecup</code>	$\cap$	<code>\doublecap</code>	$\smallsetminus$	<code>\smallsetminus</code>
$\veebar$	<code>\veebar</code>	$\bar{\wedge}$	<code>\barwedge</code>	$\overline{\wedge}$	<code>\doublebarwedge</code>
$\boxplus$	<code>\boxplus</code>	$\boxminus$	<code>\boxminus</code>	$\ominus$	<code>\circleddash</code>
$\boxtimes$	<code>\boxtimes</code>	$\boxdot$	<code>\boxdot</code>	$\odot$	<code>\circledcirc</code>
$\intercal$	<code>\intercal</code>	$\circledast$	<code>\circledast</code>	$\times$	<code>\rightthreetimes</code>
$\curlyvee$	<code>\curlyvee</code>	$\curlywedge$	<code>\curlywedge</code>	$\times$	<code>\leftthreetimes</code>

Tabela 3.16: Relacje binarne w  $\mathcal{AMS}$ .

$\lessdot$	<code>\lessdot</code>	$\gtrdot$	<code>\gtrdot</code>	$\doteqdot$	<code>\doteqdot</code>
$\leqslant$	<code>\leqslant</code>	$\geqslant$	<code>\geqslant</code>	$\risingdotseq$	<code>\risingdotseq</code>
$\eqslantless$	<code>\eqslantless</code>	$\eqslantgtr$	<code>\eqslantgtr</code>	$\fallingdotseq$	<code>\fallingdotseq</code>
$\leqq$	<code>\leqq</code>	$\geqq$	<code>\geqq</code>	$\eqcirc$	<code>\eqcirc</code>
$\lll$ or $\llless$	<code>\lll</code> or <code>\llless</code>	$\ggg$	<code>\ggg</code>	$\circeq$	<code>\circeq</code>
$\lesssim$	<code>\lesssim</code>	$\gtrsim$	<code>\gtrsim</code>	$\triangleq$	<code>\triangleq</code>
$\lessapprox$	<code>\lessapprox</code>	$\gtrapprox$	<code>\gtrapprox</code>	$\bumpeq$	<code>\bumpeq</code>
$\lessgtr$	<code>\lessgtr</code>	$\gtrless$	<code>\gtrless</code>	$\Bumpeq$	<code>\Bumpeq</code>
$\lesseqgtr$	<code>\lesseqgtr</code>	$\gtreqless$	<code>\gtreqless</code>	$\thicksim$	<code>\thicksim</code>
$\lesseqqgtr$	<code>\lesseqqgtr</code>	$\gtreqqless$	<code>\gtreqqless</code>	$\thickapprox$	<code>\thickapprox</code>
$\preccurlyeq$	<code>\preccurlyeq</code>	$\succcurlyeq$	<code>\succcurlyeq</code>	$\approxeq$	<code>\approxeq</code>
$\curlyeqprec$	<code>\curlyeqprec</code>	$\curlyeqsucc$	<code>\curlyeqsucc</code>	$\backsim$	<code>\backsim</code>
$\precsim$	<code>\precsim</code>	$\succsim$	<code>\succsim</code>	$\backsimeq$	<code>\backsimeq</code>
$\precapprox$	<code>\precapprox</code>	$\succapprox$	<code>\succapprox</code>	$\vDash$	<code>\vDash</code>
$\subseteqq$	<code>\subseteqq</code>	$\supseteqq$	<code>\supseteqq</code>	$\Vdash$	<code>\Vdash</code>
$\shortparallel$	<code>\shortparallel</code>	$\Supset$	<code>\Supset</code>	$\Vvdash$	<code>\Vvdash</code>
$\blacktriangleleft$	<code>\blacktriangleleft</code>	$\sqsupset$	<code>\sqsupset</code>	$\backepsilon$	<code>\backepsilon</code>
$\vartriangleright$	<code>\vartriangleright</code>	$\because$	<code>\because</code>	$\varpropto$	<code>\varpropto</code>
$\blacktriangleright$	<code>\blacktriangleright</code>	$\Subset$	<code>\Subset</code>	$\between$	<code>\between</code>
$\trianglerighteq$	<code>\trianglerighteq</code>	$\smallfrown$	<code>\smallfrown</code>	$\pitchfork$	<code>\pitchfork</code>
$\vartriangleleft$	<code>\vartriangleleft</code>	$\shortmid$	<code>\shortmid</code>	$\smallsmile$	<code>\smallsmile</code>
$\trianglelefteq$	<code>\trianglelefteq</code>	$\therefore$	<code>\therefore</code>	$\sqsubset$	<code>\sqsubset</code>

Tabela 3.17: Strzałki w  $\mathcal{AMS}$ .

$\dashleftarrow$	<code>\dashleftarrow</code>	$\dashrightarrow$	<code>\dashrightarrow</code>
$\Lleftarrow$	<code>\Lleftarrow</code>	$\Rrightarrow$	<code>\Rrightarrow</code>
$\twoheadleftarrow$	<code>\twoheadleftarrow</code>	$\twoheadrightarrow$	<code>\twoheadrightarrow</code>
$\leftarrowtail$	<code>\leftarrowtail</code>	$\rightarrowtail$	<code>\rightarrowtail</code>
$\leftrightharpoons$	<code>\leftrightharpoons</code>	$\rightleftharpoons$	<code>\rightleftharpoons</code>
$\Lsh$	<code>\Lsh</code>	$\Rsh$	<code>\Rsh</code>
$\looparrowleft$	<code>\looparrowleft</code>	$\looparrowright$	<code>\looparrowright</code>
$\curvearrowleft$	<code>\curvearrowleft</code>	$\curvearrowright$	<code>\curvearrowright</code>
$\circlearrowleft$	<code>\circlearrowleft</code>	$\circlearrowright$	<code>\circlearrowright</code>
$\multimap$	<code>\multimap</code>	$\Uparrow$	<code>\Uparrow</code>
$\Downdownarrows$	<code>\Downdownarrows</code>	$\Uparpoonleft$	<code>\Uparpoonleft</code>
$\Uparpoonright$	<code>\Uparpoonright</code>	$\Downharpoonright$	<code>\Downharpoonright</code>
$\rightsquigarrow$	<code>\rightsquigarrow</code>	$\leftrightsquigarrow$	<code>\leftrightsquigarrow</code>

Tabela 3.18: Zanegowane relacje binarne i strzałki w  $\mathcal{AMS}$ .

$\nless$	$\ngtr$	$\varsubsetneqq$
$\lneq$	$\gneq$	$\varsupsetneqq$
$\nleq$	$\ngeq$	$\nsubseteqeq$
$\nleqslant$	$\ngeqslant$	$\nsupseteqeq$
$\lneqq$	$\gneqq$	$\nmid$
$\lvertneqq$	$\gvertneqq$	$\nparallel$
$\nleqq$	$\ngeqq$	$\nshortmid$
$\lnsim$	$\gnsim$	$\nshortparallel$
$\lnapprox$	$\gnapprox$	$\nsim$
$\nprec$	$\nsucc$	$\ncong$
$\npreceq$	$\nsucceq$	$\nvdash$
$\precneqq$	$\succneqq$	$\nvDash$
$\precnsim$	$\succnsim$	$\nVdash$
$\precnapprox$	$\succnapprox$	$\nVDash$
$\subsetneq$	$\supsetneq$	$\ntriangleleft$
$\varsubsetneq$	$\varsupsetneq$	$\ntriangleright$
$\nsubseteq$	$\nsupseteq$	$\ntrianglelefteq$
$\subsetneqq$	$\supsetneqq$	$\ntrianglerighteq$
$\nleftarrow$	$\rightarrow$	$\leftrightharpoonup$
$\nLeftarrow$	$\Rightarrow$	$\Leftrightarrow$

Tabela 3.19: Różne symbole w  $\mathcal{AMS}$ .

$\hbar$	$\hslash$	$\mathbb{k}$
$\square$	$\blacksquare$	$\textcircled{S}$
$\triangle$	$\blacktriangle$	$\complement$
$\nabla$	$\blacktriangledown$	$\Game$
$\diamond$	$\blacklozenge$	$\bigstar$
$\sphericalangle$	$\sphericalangle$	
$\diagup$	$\diagdown$	$\backprime$
$\nexists$	$\Finv$	$\varnothing$
$\eth$	$\sphericalangle$	$\mho$





## Rozdział 4

# Specjalności

Przy składaniu większych dokumentów,  $\LaTeX$  jest w stanie pomóc z wieloma specjalnymi dodatkami, takimi jak generowanie indeksu, zarządzanie bibliografią i wiele innych. Bardziej szczegółowy opis tych i pokrewnych zagadnień można znaleźć w *LaTeX Manual* [1] oraz *The LaTeX Companion* [3].

### 4.1 Bibliografia

Stwórz bibliografię przy użyciu otoczenia `thebibliography`. Każdy wpis zaczyna się od

```
\bibitem[etykieta]{zaczynnik}
```

*Zaczynnik* można następnie użyć do zacytowania w dokumencie danej książki, artykułu bądź pracy konferencyjnej.

```
\cite{zaczynnik}
```

Jeśli nie użyjesz opcjonalnej *etykiety* to wpisy będą numerowane automatycznie. Otoczenie `thebibliography` ma jeden parametr, który określa ile miejsca należy zarezerwować na liczbę etykiet. W poniższym przykładzie zapis `99` oznacza, że numery pozycji w spisie nie będą szersze niż liczba `99`.

```
Partl~\cite{pa} zaproponował,
żeby\ldots
\begin{thebibliography}{99}
\bibitem{pa} H.~Partl:
\emph{German \TeX},
TUGboat Vol.~9, No.~1 (1988)
\end{thebibliography}
```

Partl [pa] zaproponował, żeby...

## Bibliografia

[1] H. Partl: *German T<sub>E</sub>X*, TUGboat  
Vol. 9, No. 1 (1988)

Do większych projektów przydaje się program o nazwie BibT<sub>E</sub>X. Program ten znajduje się w każdej współczesnej dystrybucji T<sub>E</sub>Xa. BibT<sub>E</sub>X korzysta z bazy bibliograficznej (biblioteki), z której wybiera tylko te pozycje literaturowe, które były cytowane w dokumencie. Sposób formatowania spisów literatury jest sterowany za pomocą specjalnych szablonów, których modyfikacja umożliwia zmianę układu graficznego spisu.

## 4.2 Indeksy

Niezwykle użytecznym elementem wielu książek jest indeks (skorowidz). Można go utworzyć stosunkowo łatwo za pomocą L<sup>A</sup>T<sub>E</sub>Xa oraz programu narzędziowego o nazwie `makeindex`<sup>1</sup>. W tym wprowadzeniu omówimy jedynie podstawowe polecenia dotyczące indeksów. Jak zawsze, więcej informacji znajdziemy w *The L<sup>A</sup>T<sub>E</sub>X Companion* [3].

Generować hasła do indeksu można dopiero po załadowaniu w preambule

<sup>1</sup>Albo `makeidx`, jeśli nasz system operacyjny nie pozwala używać nazw dłuższych niż 8 znaków.

Tabela 4.1: Przykłady użycia wpisu do indeksu.

Przykład	Wpis indeksu	Komentarz
<code>\index{witaj}</code>	witaj, 1	Zwykły wpis
<code>\index{witaj!Peter}</code>	Peter, 3	Podwpis pod ‘witaj’
<code>\index{Sam@\textsl{Sam}}</code>	<i>Sam</i> , 2	Sformatowany wpis
<code>\index{Lin@\textbf{Lin}}</code>	<b>Lin</b> , 7	Sformatowany wpis
<code>\index{Kaese@\textbf{K"ase}}</code>	<b>Käse</b> , 33	Sformatowany wpis
<code>\index{ecole@\'ecole}</code>	école, 4	Sformatowany wpis
<code>\index{Jenny textbf}</code>	Jenny, <b>3</b>	Sformatowany numer strony
<code>\index{Joe textit}</code>	Joe, 5	Sformatowany numer strony

dokumentu pakietu o nazwie `makeidx`:

```
\usepackage{makeidx}
```

oraz wstawieniu (także w obrębie preambuły) instrukcji

```
\makeindex
```

Hasło wstawiamy do indeksu poleceniem

```
\index{hasło@sformatowane hasło}
```

gdzie *sformatowane hasło* pojawi się w indeksie, a *hasło* zostanie użyte do sortowania. Część *sformatowane hasło* jest opcjonalna. Jeśli ją pominiemy zostanie użyte *hasło*. Hasła do indeksu wstawiamy w miejscu gdzie chcemy, by wpisy z indeksu wskazywały. Tabela 4.1 przedstawia składnie oraz kilka przykładów.

Gdy plik źródłowy zostanie przetworzony przez  $\text{\LaTeX}$ , wszystkie polecenia `\index` zapisują odpowiedni wpis do indeksu wraz z aktualnym numerem strony do specjalnego pliku. Jego nazwa jest taka sama jak nazwa pliku źródłowego  $\text{\LaTeX}$ , ale z innym rozszerzeniem (`.idx`). Ten plik `.idx` może

być następnie przetworzony programem `makeindex`:

```
makeindex nazwa pliku
```

W rezultacie program `makeindex` tworzy posortowany skorowidz i zapisuje go do pliku o nazwie identycznej z nazwą głównego pliku źródłowego i o rozszerzeniu `.ind`. Jeżeli jeszcze raz przetworzymy plik źródłowy, to tym razem skorowidz zostanie włączony w miejscu wystąpienia polecenia

```
\printindex
```

Pakiet `showidx`, wchodzący w skład standardowej dystrybucji  $\text{\LaTeX} 2_{\epsilon}$ , drukuje pozycje skorowidza na lewym marginesie. Jest on dość przydatny do sprawdzania i korygowania skorowidza.

Zwróć uwagę, że polecenie `\index` może zmienić układ twojego dokumentu jeśli nie będzie używane ostrożnie.

Moje Słowo `\index{Słowo}`.  
W przeciwieństwie do  
Słowa`\index{Słowo}`. Zwróć  
uwagę na pozycję kropki.

```
Moje Słowo . W przeciwieństwie do Słowa.  
Zwróć uwagę na pozycję kropki.
```

`makeindex` nie zna znaków spoza. Aby poprawnie je posortować należy użyć `@` jak pokazano w przykładach Käse oraz école powyżej.

### 4.3 Nagłówki i stopki

Pakiet `fancyhdr`<sup>2</sup>, autorstwa Piet van Oostrum, definiuje kilka prostych poleceń, które pozwalają dostosować nagłówki i stopkę dokumentu. Spójrz na górną część tej strony by zobaczyć przykład użycia tego pakietu.

Delikatnym problemem z dostosowywaniem nagłówki i stopki jest uzyskanie prawidłowych nazw sekcji i rozdziałów wewnątrz.  $\text{\LaTeX}$  uzyskuje to za pomocą dwóch kroków. W definicji nagłówki i stopki wykorzystujesz polecenia `\rightmark` oraz `\leftmark` aby uzyskać odpowiednio nagłówek aktualnej sekcji oraz rozdziału. Wartości w tych poleceniach są nadpisywane za każdym razem gdy napotkane jest polecenie rozdziału lub sekcji.

W rzeczywistości, `\chapter` oraz inne polecenia podziału logicznego dokumentu nie zmieniają znaczenia poleceń `\rightmark` i `\leftmark`. Odwołują się one natomiast do poleceń `\chaptermark`, `\sectionmark` oraz

<sup>2</sup>Dostępny w [CTAN://macros/latex/contrib/supported/fancyhdr](http://CTAN://macros/latex/contrib/supported/fancyhdr).

---

```
\documentclass{book}
\usepackage{fancyhdr}
\pagestyle{fancy}
% dzięki temu upewniamy się, że nagłówki rozdziału
% i sekcji są napisane małymi literami
\renewcommand{\chaptermark}[1]{%
    \markboth{#1}{}}
\renewcommand{\sectionmark}[1]{%
    \markright{\thesection\ #1}}
\fancyhf{} % usuń aktualny nagłówek i stopkę
\fancyhead[LE,R0]{\bfseries\thepage}
\fancyhead[LO]{\bfseries\rightmark}
\fancyhead[RE]{\bfseries\leftmark}
\renewcommand{\headrulewidth}{0.5pt}
\renewcommand{\footrulewidth}{0pt}
\addtolength{\headheight}{0.5pt} % miejsce na linię
\fancypagestyle{plain}{%
    \fancyhead{} % pozbaź się nagłówku na pustych stronach
    \renewcommand{\headrulewidth}{0pt} % tak samo z linią
}
```

---

Rysunek 4.1: Przykładowe użycie fancyhdr.

`\subsectionmark`. Dopiero użycie tych instrukcji powoduje zmianę definicji poleceń `\rightmark` i `\leftmark`.

Do zmiany postaci tytułu rozdziału w nagłówkach wystarczy modyfikacja polecenia `\chaptermark`.

Rysunek 4.1 przedstawia takie wykorzystanie pakietu, że nagłówki będą wyglądać mniej więcej tak jak w tym podręczniku. Kompletny opis pakietu znajduje się w jego dokumentacji.

## 4.4 Pakiet `verbatim`

Wcześniej w tej książce pokazaliśmy *otoczenie* `verbatim`. W tej sekcji dowiesz się o *pakięcie* `verbatim`. Pakiet `verbatim` jest reimplementacją otoczenia `verbatim`, które pozbywa się części jego ograniczeń. Nie jest to samo w sobie zachwycające jednak pakiet `verbatim` dodał również nową funkcjonalność, co jest powodem dla którego tutaj o nim wspominamy. Pakiet udostępnia polecenie

```
\verbatiminput{nazwa pliku}
```

które pozwala na włączenie pliku ASCII do dokumentu tak jakby znajdował się wewnątrz otoczenia `verbatim`.

Ponieważ pakiet `verbatim` jest częścią kolekcji „tools”, powinien on znajdować się w domyślnej dystrybucji w większości systemów. Jeśli chcesz dowiedzieć się więcej o tej paczce, możesz poczytać więcej w [13].

## 4.5 Instalacja dodatkowych pakietów

Większość dystrybucji  $\text{\LaTeX}$ a dostarcza liczny zbiór zainstalowanych pakietów, ale wiele więcej dostępnych jest w internecie. Głównym miejscem gdzie można znaleźć pakiety jest CTAN (<http://www.ctan.org/>).

Pakiety takie jak `geometry` albo `hyphenat`, a także wiele innych, składają się na ogół z dwóch plików: jednego o rozszerzeniu `.ins` i drugiego o rozszerzeniu `.dtx`. Często jest też dołączany plik `readme.txt` z krótkim opisem pakietu. Ma się rozumieć, zawsze warto zacząć od przeczytania tego właśnie pliku.

Po skopiowaniu plików pakietu na nasz komputer trzeba je w ten czy inny sposób przetworzyć, aby po pierwsze wprowadzić do dystrybucji  $\text{\TeX}$ a informację o nowym pakiecie, a po drugie uzyskać dokumentację. Oto, jak osiąga się pierwszy z tych celów:

1. Uruchamiamy  $\LaTeX$ a na pliku `.ins`. To powoduje wygenerowanie pliku `.sty`.
2. Plik `.sty` kopiujemy w miejsce, w którym nasza dystrybucja  $\TeX$ a potrafi go odnaleźć. Zazwyczaj jest to katalog o nazwie podobnej do `.../localtexmf/tex/latex` (użytkownicy systemów Windows oraz OS/2 wiedzą, że w ich systemie trzeba używać znaku ukośnika pochylonego przeciwnie).
3. Odświeżamy zawartą w dystrybucji bazę danych nazw plików. Odpowiednie polecenie zależy od dystrybucji  $\TeX$ a:  $\TeX$ live – `texhash`; `web2c` – `maktexlsr`;  $\text{MiK}\TeX$  – `initexmf --update-fndb` lub przy użyciu interfejsu graficznego

Można teraz wygenerować dokumentację z pliku `.dtx`:

1. Uruchamiamy  $X\TeX$  na pliku `.dtx`. Wygeneruje to plik `.pdf` file. Niewykluczone, że zanim w dokumentacji uporządkują się numery odsyłaczy, będziemy musieli uruchomić  $\LaTeX$ a kilka razy.
2. Sprawdzamy czy  $\LaTeX$  wygenerował plik `.idx` wśród wielu wygenerowanych plików. Jeśli go nie widać, to oznacza, że dokumentacja nie ma indeksu. Możemy przejść do kroku 5.
3. Aby wygenerować indeks, wpisz następujące polecenie:  

```
makeindex -s gind.ist nazwa
```

  
(gdzie *nazwa* oznacza nazwę głównego spośród przetwarzanych plików, bez rozszerzenia).
4. Uruchom  $\LaTeX$  na pliku `.dtx` raz jeszcze.
5. koniec coś równie ważnego: aby uprzyjemnić sobie czytanie, tworzymy plik `.ps` albo `.pdf`.

Czasami wśród wygenerowanych plików znajdziemy jeszcze plik `.glo`. W takiej sytuacji między krokami 4 i 5 powinniśmy wykonać:

```
makeindex -s gglo.ist -o nazwa.gls nazwa.glo
```

Należy również uruchomić  $\LaTeX$ a raz jeszcze na pliku `.dtx` przed przejściem do kroku 5.

## 4.6 L<sup>A</sup>T<sub>E</sub>X i PDF

Daniel Flipo <Daniel.Flipo@univ-lille1.fr>

PDF to przenośny format hipertekstowych dokumentów. Podobnie jak na stronie internetowej, niektóre słowa są w dokumencie zaznaczone jako hiperłącza. Łączą one miejsca w dokumencie z innymi jego miejscami bądź nawet z innymi dokumentami; gdy klikamy w takie hiperłącze, jesteśmy przeniesieni do miejsca docelowego. W kontekście L<sup>A</sup>T<sub>E</sub>Xa oznacza to, że wszystkie wystąpienia poleceń `\ref` i `\pageref` stają się hiperłączami. Na dodatek spis treści, skorowidz i wszystkie podobne struktury stają się zestawami łączy.

Większość spotykanych dziś w Internecie stron jest napisana w języku HTML (*HyperText Markup Language*). Kiedy przychodzi do pisania pracy naukowej, format ten ma dwie wady:

1. Nie ma dobrej metody włączania wzorów matematycznych. Chociaż został zdefiniowany odpowiedni standard, to jednak większość przeglądarek go nie obsługuje, lub nie dostarcza odpowiednich fontów.
2. Dokumenty HTML można co prawda drukować, ale wydruki wyglądają różnie w różnych przeglądarkach, a także na różnych platformach. Od jakości, do której przyzwyczajają nas L<sup>A</sup>T<sub>E</sub>X, wydruki te dzieli przepaść.

Próbowano już na wiele sposobów napisać program tłumaczący język L<sup>A</sup>T<sub>E</sub>Xa na HTML. Niektóre z tych programów dają nawet niezłe wyniki – w tym sensie, że są w stanie ze źródła w standardowym L<sup>A</sup>T<sub>E</sub>Xu wygenerować czytelny dokument HTML. Wszystkim jednak zawsze czegoś brakuje. Gdy tylko zachodzi konieczność użycia bardziej wyrafinowanych możliwości L<sup>A</sup>T<sub>E</sub>Xa bądź jego pakietów zewnętrznych, jakość spada. Autorom, którym zależy na publikowaniu w sieci WWW, a jednocześnie chcą utrzymać wysoką jakość typograficzną, pozostaje w zasadzie wyłącznie PDF. Większość współczesnych przeglądarek sieciowych potrafi wyświetlać dokumenty w tym formacie.

Wszystkie nowoczesne silniki T<sub>E</sub>Xa są w stanie generować dokumenty PDF od razu po zainstalowaniu. Jeśli wykonujesz przykłady opisane w tej książce to prawdopodobnie proces ten jest Ci znany.

### 4.6.1 Łącza hipertekstowe

Pakiet `hyperref` dodaje dwie klawie funkcje do Twoich plików PDF z L<sup>A</sup>T<sub>E</sub>Xa:



1. Rozmiar papieru jest ustawiony zgodnie ze specyfikacją w opcjach klasy dokumentu.
2. Wszystkie odniesienia w dokumencie staną się łączami.

Aby to osiągnąć wystarczy dodać `\usepackage{hyperref}` jako *ostatnie* polecenie w preambule Twojego dokumentu.

Dostępne jest wiele opcji modyfikujących zachowanie pakietu `hyperref`, które można podać:

- jako listę oddzieloną przecinkami po opcji `pdftex`  
`\usepackage{hyperref}`
- jako indywidualne linie przy użyciu polecenia `\hypersetup{opcje}`.

W poniższym wykazie wartości domyślne są podane pismem prostym.

**bookmarks** (`=true, false`) pokaż lub ukryj pasek zakładek podczas wyświetlania dokumentu

**unicode** (`=false, true`) pozwala na wykorzystanie znaków nielacińskich alfabetów w zakładkach Acrobat

**pdftoolbar** (`=true, false`) pokaż bądź ukryj pasek narzędziowy Acrobat

**pdfmenubar** (`=true, false`) pokaż bądź ukryj menu Acrobat

**pdffitwindow** (`=false, true`) dostosuj wielkość wyświetlanego PDFa do wielkości okna

**pdftitle** (`={tekst}`) tytuł dokumentu wyświetlany w informacji o dokumencie

**pdfauthor** (`={tekst}`) nazwisko autora

**pdfnewwindow** (`=false, true`) określa, czy w wypadku gdy łącze prowadzi poza dokument, ma być otwierane nowe okna

**colorlinks** (`=false, true`) określa, czy otoczyć hiperłącza kolorowymi ramkami (`false`) czy kolorować same hiperłącza (`true`). Kolory można konfigurować za pomocą następujących opcji (w nawiasach kolory domyślne):

`linkcolor (=red)` kolor łączy wewnętrznych (rozdziałów, punktów, stron itp.),  
`citecolor (=green)` kolor cytowań, czyli odsyłaczy do bibliografii,  
`filecolor (=magenta)` kolor odsyłaczy do plików,  
`urlcolor (=cyan)` kolor łączy typu URL (adresy poczty elektronicznej bądź sieciowe).

Jeśli zadowolają nas ustawienia domyślne, to wpisujemy:

```
\usepackage{hyperref}
```

Jeśli przeglądarka ma pokazać listę zakładek, a łączy mają być kolorowane (wartości domyślnych `=true` nie trzeba podawać), to:

```
\usepackage[bookmarks,colorlinks]{hyperref}
```

Gdy tworzymy dokumenty PDF przeznaczone do druku czarno-białego, to kolorowanie łączy nie jest najlepszym pomysłem, bo w wydrukach wychodzą one szare, co utrudnia czytanie. Zamiast tego możemy użyć kolorowych ramek, które nie są drukowane:

```
\usepackage{hyperref}
\hypersetup{colorlinks=false}
```

albo dla łączy używać koloru czarnego:

```
\usepackage{hyperref}
\hypersetup{colorlinks,%
             citecolor=black,%
             filecolor=black,%
             linkcolor=black,%
             urlcolor=black,%
             pdftex}
```

dostarczyć informacji

Jeśli chcesz tylko uzupełnić sekcję Informacje o dokumencie w pliku PDF:

```
\usepackage[pdftitle={Des femmes qui tombent},%
             pdftext]{hyperref}
```

Oprócz hiperłączy tworzonych automatycznie możemy również umieszczać własne przy użyciu

```
\href{url}{tekst}
```

Wynikiem kodu

```
Strona \href{http://www.ctan.org}{CTAN}.
```

jest „[CTAN](http://www.ctan.org)”; kliknięcie na napis „[CTAN](http://www.ctan.org)” zabierze cię na stronę CTAN.

Jeśli docelowe miejsce łączy nie jest URL-em, lecz plikiem lokalnym, to możemy użyć polecenia `\href` w następującej postaci:

```
Pełny dokument jest \href{manual.pdf}{tutaj}
```

Czego efektem jest „Pełny dokument jest [tutaj](#)”. Kliknięcie na napis „[tutaj](#)” otworzy plik `manual.pdf`. (Nazwa pliku jest interpretowana względem położenia dokumentu PDF).

Autorka dokumentu może zachęcić czytelników do korespondencji elektronicznej, wpisując instrukcję `\href` wewnątrz polecenia `\author` na stronie tytułowej dokumentu:

```
\author{Mary Oetiker <\href{mailto:mary@oetiker.ch}%  
{mary@oetiker.ch}>}
```

Zauważmy, że łączy zapisano tu tak, iż adres elektroniczny pojawia się zarówno w łączy, jak i na stronie. Gdyby podać go tak:

```
\href{mailto:mary@oetiker.ch}{Maria Oetiker}
```

to łączy działałoby w programie Acrobat, lecz adres nie byłby widoczny w wydruku.

### 4.6.2 Problemy z łączy

Ukazaniu się komunikatu w rodzaju:

```
! pdfTeX warning (ext4): destination with the same  
  identifier (name{page.1}) has been already used,  
  duplicate ignored
```

towarzyszy reinicjalizacja licznika, na przykład w wyniku użycia polecenia `\mainmatter`, dostępnego w klasie dokumentów `book`. Ustala ono wartość licznika stron na 1 tuż przed pierwszym rozdziałem książki. Ale ponieważ również i wstęp do książki zawiera stronę 1, to odsyłacz „strona 1” staje się niejednoznaczny, stąd notka „duplicate ignored”.

Możemy temu przeciwdziałać, dodając opcję `plainpages=false`. Pomaga to niestety tylko w odniesieniu do licznika stron. Radykalniejszym rozwiązaniem jest opcja `hypertextnames=false`, która jednak powoduje, że przestają działać odsyłacze do stron w indeksie.

### 4.6.3 Problemy z zakładkami

Tekst na zakładkach nie zawsze wygląda zgodnie z naszymi oczekiwaniami. Ponieważ zakładki są traktowane jako „czysto tekstowe”, może w nich wystąpić mniejszy zakres znaków niż w normalnym tekście  $\text{\LaTeX}$ owym. Pakiet `hyperref` zauważy tego typu problem i zasygnalizuje go komunikatem:

```
Package hyperref Warning:
Token not allowed in a PDFDocEncoded string:
```

Możemy ten problem obejść, podając napis przeznaczony na zakładkę, który ma zastąpić wywołujący trudność tekst:

```
\texorpdfstring{Tekst dla  $\text{\TeX}$ }{Tekst dla zakładki}
```

Podstawowym kandydatem do zastosowania takiego postępowania są wzory matematyczne:

```
\section{\texorpdfstring{\$E=mc^2\$}%
{E = mc ** 2}}
```

Powyższy kod zamieni `\section{\$E=mc^2\$}` na „E = mc \*\* 2” polu zakładki.

Jeśli piszesz swój dokument w Unicode i korzystasz z opcji `unicode` pakietu `hyperref` aby pozwolić na wykorzystanie znaków unicode w zakładkach, to możesz wykorzystać znacznie więcej znaków do wyboru przy używaniu `\texorpdfstring`.

## 4.7 Praca z $\text{\XeLaTeX}$ i PDF

Axel Kielhorn <[A.Kielhorn@web.de](mailto:A.Kielhorn@web.de)>

Większość punktów opisanych w poprzedniej sekcji odnosi się również do  $\text{\XeLaTeX}$ .

Istnieje Wiki na <http://wiki.xelatex.org/doku.php>, która gromadzi informacje istotne dla  $\text{\XeLaTeX}$  i  $\text{\XeLaTeX}$ .

### 4.7.1 Fonty

Oprócz zwyczajnych fontów opartych na `tfm`, X<sub>Y</sub>LaTeX jest w stanie użyć dowolnego fontu znanego w systemie operacyjnym. Jeśli masz zainstalowany font Linux Libertine to możesz po prostu napisać

```
\usepackage{fontspec}
\setmainfont[Ligatures=TeX]{Linux Libertine}
```

w preambule. Wersja pogrubiona i kursywa zostanie wykryta automatycznie, więc `\textit` i `\textbf` będą działać zgodnie z oczekiwaniami. Gdy font korzysta z technologii OpenType to zyskujesz również dostęp do wielu funkcji, które wymagały dotąd zmiany na inne fonty bądź korzystania z wirtualnych fontów w przeszłości. Główną zaletą jest rozszerzony zbiór znaków; font może zawierać zarówno znaki łacińskie, greckie, cyrylice oraz odpowiednie ligatury dla nich.

Wiele fontów zawiera również dwa rodzaje cyfr, zwyczajnie ustawiane cyfry oraz tak zwane stare cyfry, które wystają nieco poniżej linii tekstu. Mogą też oprócz proporcjonalnych cyfr (gdzie „1” zajmuje mniej miejsca niż „0”) zawierać również cyfry o jednakowej szerokości, które lepiej sprawdzają się w tabelach.

```
\newfontfamily\LLln[Numbers=Lining]{(font)}
\newfontfamily\LLos[Numbers=OldStyle]{(font)}
\newfontfamily\LLlnm[Numbers=Lining,Numbers=Monospaced]{(font)}
\newfontfamily\LLosm[Numbers=OldStyle,Numbers=Monospaced]{(font)}
```

Niemal wszystkie fonty OpenType zawierają standardowe ligatury (fi fi ffi), ale istnieją również rzadsze bądź historyczne ligatury takie jak st, ct czy tz. Prawdopodobnie nie chcesz ich używać w pismach naukowych jednak mogą być przydatne w nowelach. Aby włączyć te ligatury wykorzystaj jedną z poniższych linii:

```
\setmainfont[Ligatures=Rare]{(font)}
\setmainfont[Ligatures=Historic]{(font)}
\setmainfont[Ligatures=Historic,Ligatures=Rare]{(font)}
```

Nie każdy font zawiera oba zbiory ligatury, sprawdź dokumentację lub go wypróbuj by się dowiedzieć czy działa. Czasami ligatury te są zależne od języka; przykładowo w języku Polskim (fk) nie jest wykorzystywane w angielskim. Musisz dodać

```
\setmainfont [Language=Polish] {(font)}
```

aby włączyć polskie ligatury.

Niektóre fonty (jak na przykład komercyjne Adobe Garamond Premier Pro) zawierają alternatywne glify, które są aktywne domyślnie w  $X_{\text{L}}\text{A}_{\text{T}}\text{E}_{\text{X}}$ u w dystrybucji  $\text{T}_{\text{E}}\text{X}$ Live 2010<sup>3</sup>. Rezultatem jest stylowe „Q”, którego kreska sięga poniżej „u”. Aby wyłączyć tę funkcję musisz zdefiniować font z wyłączonymi „contextualami”:

```
\setmainfont [Contextuals=NoAlternate] {(font)}
```

Aby dowiedzieć się więcej o fontach w  $X_{\text{L}}\text{A}_{\text{T}}\text{E}_{\text{X}}$  przeczytaj dokumentację pakietu `fontspec`.

### Skąd wziąć fonty OpenType?

Jeśli masz zainstalowanego  $\text{T}_{\text{E}}\text{X}$ Live to już posiadasz ich nieco w katalogu `.../texmf-dist/fonts/opentype`, wystarczy, że zainstalujesz je w swoim systemie. Kolekcja ta nie zawiera DejaVu, które możesz znaleźć na <http://dejavu-fonts.org/>.

Upewnij się, że każdy font zainstalowany jest dokładnie *raz*, w przeciwnym razie mogą się wydarzyć ciekawe rzeczy.

Możesz korzystać z dowolnego fontu zainstalowanego na Twoim komputerze, jednak pamiętaj, że inni użytkownicy mogą nie mieć danego fontu u siebie. Font Zapfino użyty w dokumentacji `fontspec` jest dostępny w Mac OSX, ale nie jest dostępny na komputerach z systemem Windows<sup>4</sup>.

### Wprowadzanie znaków Unicode

Liczba znaków w fontach znacząco wzrosła, jednak liczba klawiszy na klawiaturze już nie. Więc jak możemy wprowadzać znaki spoza ASCII?

Jeśli piszesz sporą część tekstu w obcym języku, możesz zainstalować układ klawiatury i wydrukować pozycje znaków (większość systemów operacyjnych zawiera jakąś klawiaturę wirtualną, więc wystarczy wykonać zrzut ekranu).

Jeśli tylko czasami potrzebujesz wprowadzić inny znak, możesz zwykle wybrać z palety znaków.

<sup>3</sup>W poprzednich wersjach było to domyślnie wyłączone.

<sup>4</sup>Dostępna jest komercyjna wersja fontu o nazwie Zapfino Extra.

Niektóre otoczenia (np. X Window System) oferuje wiele opcji na wprowadzanie znaków spoza ASCII. Niektóre edytory (np. Vim i Emacs) oferują sposoby wprowadzania tych znaków. Przeczytaj dokumentację narzędzi z których korzystasz.

### 4.7.2 Kompatybilność pomiędzy X<sub>q</sub>LaTeX, a pdfLaTeX

Istnieją pewne różnice pomiędzy X<sub>q</sub>LaTeXem i pdfLaTeXem.

- Dokument X<sub>q</sub>LaTeX musi być napisany w Unicode (UTF-8) podczas gdy pdfLaTeX może korzystać z różnych kodowań plików źródłowego.
- Pakiet `microtype` nie działa jeszcze z X<sub>q</sub>LaTeX, jednak wsparcie dla występów liter jest w trakcie tworzenia<sup>1</sup>.
- Wszystko co związane z fontami musi zostać sprawdzone (chyba, że korzystasz z Latin Modern).

## 4.8 Tworzenie prezentacji

Daniel Flipo <Daniel.Flipo@univ-lille1.fr>

Wyniki naszej pracy naukowej możemy przedstawiać kredą na tablicy, za pomocą rzutnika i przeźroczy (slajdów) bądź – posługując się odpowiednim oprogramowaniem – bezpośrednio z laptopa.

Program pdfLaTeX w połączeniu z klasą `beamer` służy do tworzenia prezentacji w formacie PDF. Wyglądają one tak, jak gdybyśmy je wygenerowali – mając dobry dzień i sporo szczęścia – za pomocą PowerPointa, ale są bardziej przenośne, ponieważ programy obsługujące PDF są dostępne w większej liczbie systemów.

Klasa `beamer` używa pakietów `graphicx`, `color` oraz `hyperref` z opcjami zaadaptowanymi do prezentacji ekranowych.

Kompilując kod sprezentowany w rysunku 4.2 przy pomocy pdfLaTeX otrzymamy plik PDF ze stroną tytułową oraz jeszcze jedną stronę, zawierającą kilka punktów, które mają się odsłaniać w miarę, jak podczas prezentacji będziemy przechodzili do kolejnych jej kroków.

Jedną z zalet klasy `beamer` jest to, że generuje ona gotowy do użycia plik PDF, bez konieczności przechodzenia przez fazę postscriptową, jak to jest w wypadku pakietu `prospere` albo wymagającego dodatkowego przetworzenia pakietu `ppower4`.

Korzystając z klasy `beamer`, możemy z tego samego pliku źródłowego generować kilka wersji prezentacji, tak zwanych trybów. Plik źródłowy może

```
\documentclass[10pt]{beamer}
\mode<beamer>{%
  \usetheme[hideothersubsections,
            right,width=22mm]{Goettingen}
}

\title{Prosta prezentacja}
\author[D. Flipo]{Daniel Flipo}
\institute{U.S.T.L. \& GUTenberg}
\titlegraphic{\includegraphics[width=20mm]{USTL}}
\date{2005}

\begin{document}

\begin{frame}<handout:0>
  \titlepage
\end{frame}

\section{Przykład}

\begin{frame}
  \frametitle{Co robić w niedzielne popołudnie?}
  \begin{block}{Można\ldots}
    \begin{itemize}
      \item pójść na spacer z psem\ldots \pause
      \item przeczytać książkę\pause
      \item pobawić się z kotem\pause
    \end{itemize}
  \end{block}
  i wiele innych rzeczy.
\end{frame}
\end{document}
```

Rysunek 4.2: Prosty kod dla klasy beamer



w nawiasach kątowych zawierać instrukcje przeznaczone do różnych trybów. Dostępne są następujące tryby:

**beamer** dla omówionych wyżej prezentacji PDF,

**trans** do slajdów.

**handout** do wydruku.

Domyślnym jest **beamer**, a inny tryb możemy zadać jako opcję globalną, wpisując na przykład `\documentclass[10pt,handout]{beamer}` w celu wydrukowania materiałów do rozdania.

Wygląd ekranu prezentacji zależy od wybranego motywu. Możemy wskazać jeden z motywów dostarczanych wraz z klasą albo stworzyć nasz własny. Więcej informacji na ten temat można znaleźć w opisie klasy zawartym w pliku `beameruserguide.pdf`.

Przyjrzyjmy się bliżej kodowi z rysunku 4.2.

Do wersji ekranowej prezentacji wybrano dla trybu `\mode<beamer>` motyw *Goettingen*, w którym spisowi treści towarzyszy panel do nawigacji. Opcje motywu pozwalają określić rozmiar panelu (w tym wypadku 22 mm) oraz jego pozycję (z prawej strony głównego tekstu). Opcja *hideothersubsections* nakazuje pokazywać tytuły punktów, jednak tylko te, które pochodzą z bieżącego rozdziału. Dla trybów `\mode<trans>` i `\mode<handout>` nie zadano żadnej opcji; mają się ukazywać w swoim układzie domyślnym.

Polecenia `\title`, `\author`, `\institute`, oraz `\titlegraphic` określają zawartość strony tytułowej. Opcjonalne argumenty poleceń `\title` oraz `\author` pozwalają podać specjalną wersję tytułu oraz autora do wyświetlenia w panelu motywu *Goettingen*.

Tytuły oraz podtytuły na panelu są tworzone przez zwykłe polecenia `\section` i `\subsection`, umieszczone *poza* otoczeniem `frame`.

Małe ikony u dołu ekranu służą do nawigowania po dokumencie. Ich obecność nie zależy od wybranego motywu.

Zawartość każdego slajdu oraz ekranu należy umieścić wewnątrz otoczenia `frame`. W nawiasach kątowych `< i >` można podać opcjonalny argument, który pozwala ukryć slajd w jednym z trybów prezentacyjnych. W powyższym przykładzie pierwsza strona nie ukaże się w trybie materiałów do rozdania, gdyż w argumentcie otoczenia `frame` podano argument `<handout:0>`.

Warto zatytułować każdy ze slajdów z wyjątkiem slajdu tytułowego. Służy do tego polecenie `\frametitle{}`. Jeśli potrzebny jest również podtytuł użyj otoczenia `block` tak jak pokazano w przykładzie. Zauważmy, że polecenia `\section` i `\subsection` nie składają niczego na faktycznych slajdach.

Użycie polecenia `\pause` w otoczeniu `itemize` pozwala rozwijać punkty jeden po drugim, w miarę postępów prezentacji. Dodatkowe efekty prezentacyjne można osiągnąć za pomocą instrukcji: `\only`, `\uncover`, `\alt` oraz `\temporal`. W wielu miejscach możliwe jest użycie nawiasów kątowych do dalszego sterowania prezentacją.

Cokolwiek mówić, aby uzyskać pełny obraz wszystkich dostępnych parametrów, trzeba przeczytać dokumentację `beameruserguide.pdf` klasy `beamer`. Pakiet ten jest ciągle rozwijany, dlatego warto po nowości zajrzeć na ich stronę (<http://latex-beamer.sourceforge.net/>).

## Rozdział 5

# Tworzenie grafiki matematycznej

Większość ludzi używa  $\LaTeX$ a do składania tekstów. Ponieważ jednak podejście strukturalno-logiczne do tworzenia dokumentów jest tak wygodne,  $\LaTeX$  oferuje pewną – fakt, że obarczoną ograniczeniami – możliwość generowania grafiki z opisów tekstowych w pliku źródłowym. Co więcej, powstało sporo rozszerzeń  $\LaTeX$ a przełamujących wspomniane ograniczenia. W tym rozdziale dowiesz się o kilku z tych rozszerzeń.

### 5.1 Przegląd

Tworzenie grafik w  $\LaTeX$ u ma długą tradycję. Rozpoczęło się ono od otoczenia `picture`, które pozwala na tworzenie grafik przez umieszczanie przygotowanych elementów na płótno. Jego pełny opis można znaleźć w  *$\LaTeX$  Manual* [1]. Otoczenie `picture` w  $\LaTeX$  2<sub>ε</sub> dodaje polecenie `\qbezier`, gdzie „q” oznacza „kwadratowa” (ang. „quadratic”). Wiele często używanych krzywych takich jak okręgi, elipsy lub krzywa łańcuchowa może być wystarczająco dobrze przybliżona kwadratowymi krzywymi Béziera jednak może to wymagać nieco przeliczeń. Jeśli dodatkowo wykorzystamy język programowania do generowania bloków `\qbezier` w plikach źródłowych to otoczenie `picture` staje się bardzo potężne.

Chociaż programowanie rysunków w  $\LaTeX$ u wiąże się z ograniczeniami i jest niekiedy nużące, to jednak są powody, by z tej możliwości korzystać. Dokumenty wytworzone w ten sposób są „małe” – w sensie liczby bajtów zajmowanych przez pliki – a na dodatek nie trzeba do nich wczytywać dodatkowych plików graficznych.

Taki był stan rzeczy dopóki kilka lat temu Till Tantau z klasy `beamer` przygotował Portable Graphics Format `pgf` oraz pakiet akompaniujący `TikZ` (`tikz`). System ten pozwala na tworzenie grafiki wektorowej wysokiej jakości przy użyciu wszystkich aktualnych systemów `TeX`a włączając w to pełne wsparcie PDF.

Korzystając z tych podstaw, wiele pakietów zostało utworzonych do rozwiązywania konkretnych problemów. Wiele z nich zostało dokładnie opisanych w *The L<sup>A</sup>T<sub>E</sub>X Graphics Companion* [14].

Spośród narzędzi graficznych związanych z `LATEX`em największe chyba możliwości posiada `METAPOST`. Jest to samodzielna aplikacja oparta na programie `METAFONT` Donald E. Knutha. Język `METAPOST` posiada bardzo potężny i matematycznie zaawansowany język programowania `METAFONT`, ale w przeciwieństwie do niego generuje pliki `POSTSCRIPT`, które można zaimportować do `LATEX`a, a nawet do `pdfLATEX`a. Jako wprowadzenie możesz przeczytać *A User's Manual for METAPOST* [15], lub tutorial w [16].

Bardzo dokładną dyskusję strategii dla `LATEX`a i `TeX`a dotyczących grafik (i fontów) można znaleźć w *TeX Unbound* [17].

## 5.2 Otoczenie `picture`

Urs Oswald <osurs@bluewin.ch>

Jak wspomniano wyżej otoczenie `picture` jest częścią standardowego `LATEX`a i jest świetne do prostych zadań oraz w przypadku gdy chcesz dokładnej kontroli nad ułożeniem elementów na stronie. Jednak jeśli zamierzasz robić poważne grafiki matematyczne lepiej od razu użyć `TikZ`a jak pokazano w sekcji 5.3 na stronie 5.3.

### 5.2.1 Podstawowe polecenia

Otoczenie `picture`<sup>1</sup> można tworzyć poleceniem

```
\begin{picture}(x,y)...\end{picture}
```

lub

```
\begin{picture}(x,y)(x_0,y_0)...\end{picture}
```

Liczby  $x$ ,  $y$ ,  $x_0$ ,  $y_0$  odnoszą się do wielkości `\unitlength`, którą można

<sup>1</sup>Otoczenie `picture` działa w standardowym `LATEX 2 $\epsilon$` ; nie trzeba ładować żadnych dodatkowych pakietów.

zmieniać w dowolnym momencie (jednak nie wewnątrz otoczenia picture) poleceniem takim jak

```
\setlength{\unitlength}{1.2cm}
```

Wartością domyślną `\unitlength` jest 1pt. Pierwsza para,  $(x, y)$ , to wymiary rezerwowanego wewnątrz dokumentu prostokątnego obszaru na rysunek. Opcjonalna druga para,  $(x_0, y_0)$ , to współrzędne przypisane dolnemu lewemu narożnikowi zarezerwowanego prostokąta.

Większość poleceń rysujących ma jedną z dwóch postaci:

```
\put(x, y){obiekt}
```

lub

```
\multiput(x, y)(\Delta x, \Delta y){n}{object}
```

Wyjątkiem są krzywe Béziera, gdyż rysuje się je poleceniem

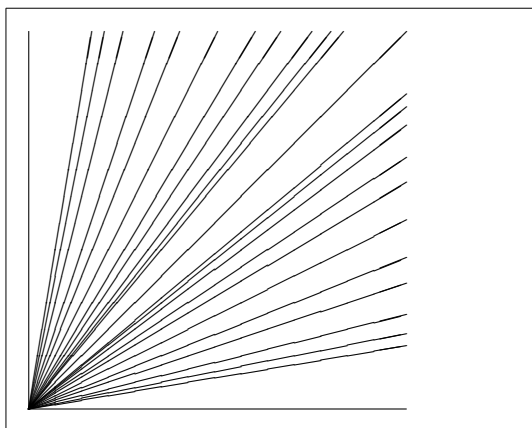
```
\qBezier(x_1, y_1)(x_2, y_2)(x_3, y_3)
```

### 5.2.2 Odcinki

```

\setlength{\unitlength}{5cm}
\begin{picture}(1,1)
  \put(0,0){\line(0,1){1}}
  \put(0,0){\line(1,0){1}}
  \put(0,0){\line(1,1){1}}
  \put(0,0){\line(1,2){.5}}
  \put(0,0){\line(1,3){.3333}}
  \put(0,0){\line(1,4){.25}}
  \put(0,0){\line(1,5){.2}}
  \put(0,0){\line(1,6){.1667}}
  \put(0,0){\line(2,1){1}}
  \put(0,0){\line(2,3){.6667}}
  \put(0,0){\line(2,5){.4}}
  \put(0,0){\line(3,1){1}}
  \put(0,0){\line(3,2){1}}
  \put(0,0){\line(3,4){.75}}
  \put(0,0){\line(3,5){.6}}
  \put(0,0){\line(4,1){1}}
  \put(0,0){\line(4,3){1}}
  \put(0,0){\line(4,5){.8}}
  \put(0,0){\line(5,1){1}}
  \put(0,0){\line(5,2){1}}
  \put(0,0){\line(5,3){1}}
  \put(0,0){\line(5,4){1}}
  \put(0,0){\line(5,6){.8333}}
  \put(0,0){\line(6,1){1}}
  \put(0,0){\line(6,5){1}}
\end{picture}

```



Odcinki mogą być narysowane przy użyciu

$\text{\put}(x,y)\{\text{\line}(x_1,y_1)\{length\}$

Polecenie `\linema` dwa argumenty:

1. wektor kierunkowy,
2. długość.

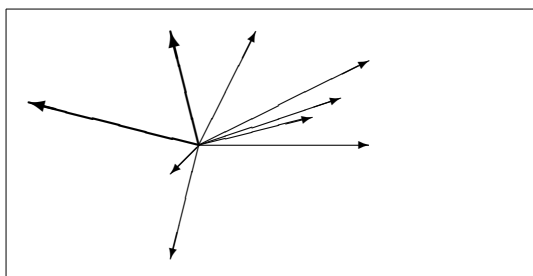
Jako składników wektora kierunku można użyć jedynie liczb całkowitych:

-6, -5, ..., 5, 6,

Muszą one na dodatek być liczbami względnie pierwszymi (jedynym ich wspólnym dzielnikiem może być 1). Powyższy rysunek ilustruje wszystkie możliwe wartości nachylenia w pierwszej ćwiartce płaszczyzny. Długość jest podawana w jednostkach `\unitlength`. Argument długości oznacza współrzędną pionową w wypadku odcinka pionowego, zaś współrzędną poziomą we wszystkich pozostałych przypadkach.

### 5.2.3 Strzałki

```
\setlength{\unitlength}{0.75mm}
\begin{picture}(60,40)
  \put(30,20){\vector(1,0){30}}
  \put(30,20){\vector(4,1){20}}
  \put(30,20){\vector(3,1){25}}
  \put(30,20){\vector(2,1){30}}
  \put(30,20){\vector(1,2){10}}
  \thicklines
  \put(30,20){\vector(-4,1){30}}
  \put(30,20){\vector(-1,4){5}}
  \thinlines
  \put(30,20){\vector(-1,-1){5}}
  \put(30,20){\vector(-1,-4){5}}
\end{picture}
```



Strzałki rysuje się poleceniem

```
\put(x,y){\vector(x1,y1){length}}
```

W wypadku strzałek na składniki wektora kierunku nałożone są jeszcze większe ograniczenia niż dla odcinków, bo jedynymi dopuszczalnymi liczbami całkowitymi są

$$-4, -3, \dots, 3, 4.$$

Wartości składników i tu muszą być liczbami względnie pierwszymi (jedynym wspólnym dzielnikiem może być 1). Zauważ efekt działania polecenia `\thicklines` na dwie strzałki skierowane w stronę lewego górnego narożnika.

### 5.2.4 Okręgi

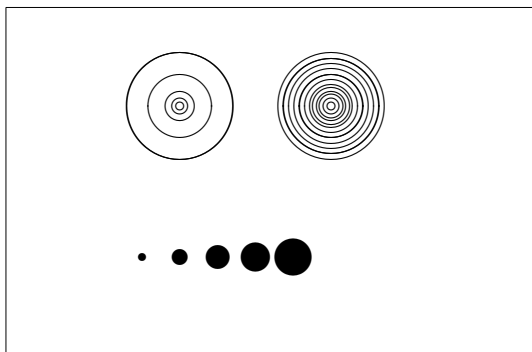
```

\setlength{\unitlength}{1mm}
\begin{picture}(60, 40)
  \put(20,30){\circle{1}}
  \put(20,30){\circle{2}}
  \put(20,30){\circle{4}}
  \put(20,30){\circle{8}}
  \put(20,30){\circle{16}}
  \put(20,30){\circle{32}}

  \put(40,30){\circle{1}}
  \put(40,30){\circle{2}}
  \put(40,30){\circle{3}}
  \put(40,30){\circle{4}}
  \put(40,30){\circle{5}}
  \put(40,30){\circle{6}}
  \put(40,30){\circle{7}}
  \put(40,30){\circle{8}}
  \put(40,30){\circle{9}}
  \put(40,30){\circle{10}}
  \put(40,30){\circle{11}}
  \put(40,30){\circle{12}}
  \put(40,30){\circle{13}}
  \put(40,30){\circle{14}}

  \put(15,10){\circle*{1}}
  \put(20,10){\circle*{2}}
  \put(25,10){\circle*{3}}
  \put(30,10){\circle*{4}}
  \put(35,10){\circle*{5}}
\end{picture}

```



Polecenie

```
\put( $x$ , $y$ ){\circle{średnica}}
```

rysuje okrąg o środku  $(x, y)$  i średnicy (nie promieniu) równym *średnica*. Otoczenie `picture` dopuszcza średnice co najwyżej rzędu 14 mm, na dodatek nie wszystkie długości średnic są dozwolone. Polecenie `\circle*` generuje koła (wypełnione okręgi).

Podobnie jak w wypadku odcinków, można się uciec do dodatkowych pakietów, takich jak `eepic` bądź `pstricks`. Obszerny opis tych pakietów można

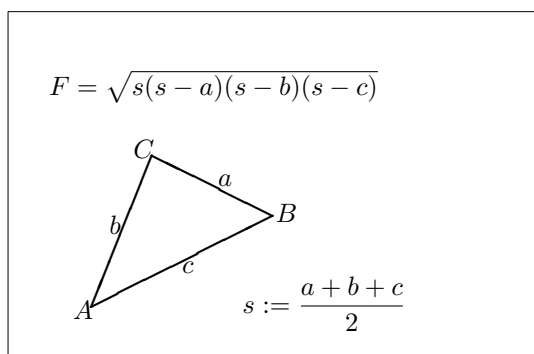


znaleźć w *The L<sup>A</sup>T<sub>E</sub>X Graphics Companion* [14].

Otoczenie picture oferuje jeszcze jedną opcję. Jeśli nie boisz się wykonania niezbędnych obliczeń (być może nawet za pomocą programu), to okręgi oraz elipsy możesz połączyć z krzywymi Béziera drugiego stopnia. Przykłady oraz źródłowe pliki w Javie można znaleźć w *Graphics in L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub>* [16].

### 5.2.5 Tekst i wzory

```
\setlength{\unitlength}{0.8cm}
\begin{picture}(6,5)
  \thicklines
  \put(1,0.5){\line(2,1){3}}
  \put(4,2){\line(-2,1){2}}
  \put(2,3){\line(-2,-5){1}}
  \put(0.7,0.3){$A$}
  \put(4.05,1.9){$B$}
  \put(1.7,2.95){$C$}
  \put(3.1,2.5){$a$}
  \put(1.3,1.7){$b$}
  \put(2.5,1.05){$c$}
  \put(0.3,4){$F=$
    \sqrt{s(s-a)(s-b)(s-c)}$}
  \put(3.5,0.4){$\displaystyle
    s:=\frac{a+b+c}{2}$}
\end{picture}
```



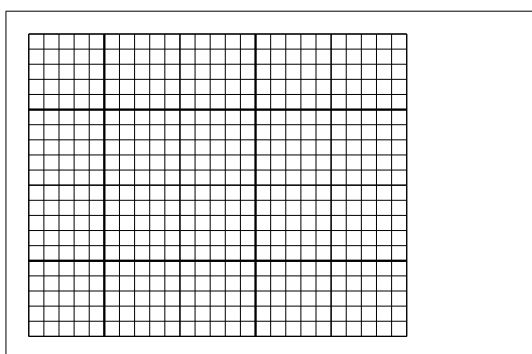
Jak widać w powyższym przykładzie, tekst oraz wzory można łatwo wprowadzać do otoczenia picture poleceniem `\put`.

5.2.6 `\multiput` i `\linethickness`

```

\setlength{\unitlength}{2mm}
\begin{picture}(30,20)
  \linethickness{0.075mm}
  \multiput(0,0)(1,0){26}%
    {\line(0,1){20}}
  \multiput(0,0)(0,1){21}%
    {\line(1,0){25}}
  \linethickness{0.15mm}
  \multiput(0,0)(5,0){6}%
    {\line(0,1){20}}
  \multiput(0,0)(0,5){5}%
    {\line(1,0){25}}
  \linethickness{0.3mm}
  \multiput(5,0)(10,0){2}%
    {\line(0,1){20}}
  \multiput(0,5)(0,10){2}%
    {\line(1,0){25}}
\end{picture}

```



Polecenie

```
\multiput(x,y)(\Delta x,\Delta y){n}{object}
```

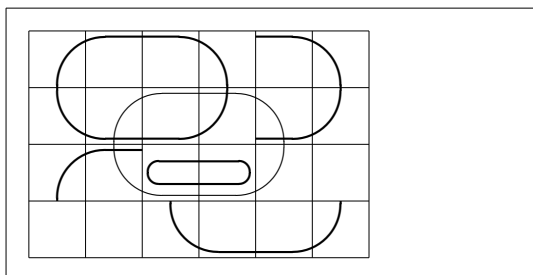
ma cztery argumenty: punkt początkowy, wektor przesunięcia z jednego obiektu do kolejnego, liczbę obiektów oraz obiekt do narysowania. Instrukcja `\linethickness` odnosi się do odcinków poziomych oraz pionowych, jednakże nie do odcinków ukośnych ani nie do okręgów. Stosuje się ona jednak do krzywych Béziera drugiego stopnia!

## 5.2.7 Owale

```

\setlength{\unitlength}{0.75cm}
\begin{picture}(6,4)
  \linethickness{0.075mm}
  \multiput(0,0)(1,0){7}%
    {\line(0,1){4}}
  \multiput(0,0)(0,1){5}%
    {\line(1,0){6}}
  \thicklines
  \put(2,3){\oval(3,1.8)}
  \thinlines
  \put(3,2){\oval(3,1.8)}
  \thicklines
  \put(2,1){\oval(3,1.8)[t1]}
  \put(4,1){\oval(3,1.8)[b]}
  \put(4,3){\oval(3,1.8)[r]}
  \put(3,1.5){\oval(1.8,0.4)}
\end{picture}

```



Polecenie

```
\put(x,y){\oval(w,h)}
```

lub

```
\put(x,y){\oval(w,h)[pozycja]}
```

generuje owal o środku  $(x, y)$ , szerokości  $w$  i wysokości  $h$ . Opcjonalne argumenty *pozycji*: b, t, l i r oznaczają odpowiednio: „bottom” (dół), „top” (górze), „left” (lewo) i „right” (prawo). Jak pokazuje przykład, można także używać ich kombinacji.

Grubość linii można sterować na dwa sposoby: z jednej strony przy użyciu

```
\linethickness{długość}
```

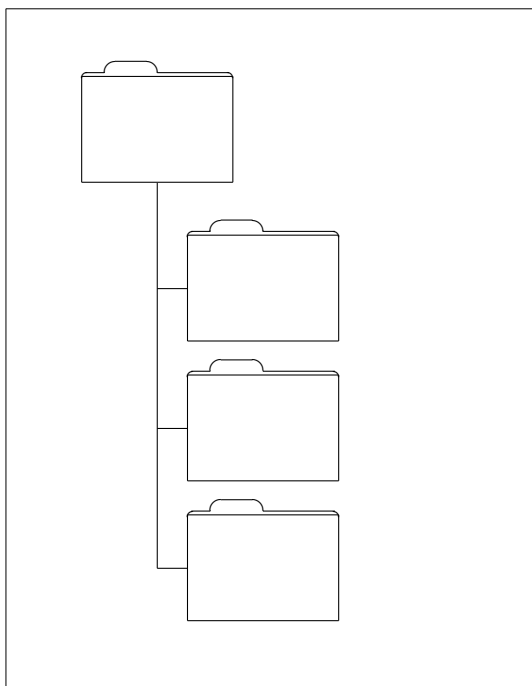
z drugiej za pomocą `\thinlines` i `\thicklines`. O ile pierwszy ze sposobów odnosi się jedynie do linii poziomych oraz pionowych, a także do krzywych Béziera drugiego stopnia, o tyle `\thinlines` i `\thicklines` stosują się do odcinków ukośnych, jak też do okręgów i owali.

## 5.2.8 Wielokrotne użycie pudełek z rysunkami

```

\setlength{\unitlength}{0.5mm}
\begin{picture}(120,168)
\newsavebox{\foldera}
\savebox{\foldera}
(40,32)[b1]{% definition
\multiput(0,0)(0,28){2}
{\line(1,0){40}}
\multiput(0,0)(40,0){2}
{\line(0,1){28}}
\put(1,28){\oval(2,2)[t1]}
\put(1,29){\line(1,0){5}}
\put(9,29){\oval(6,6)[t1]}
\put(9,32){\line(1,0){8}}
\put(17,29){\oval(6,6)[tr]}
\put(20,29){\line(1,0){19}}
\put(39,28){\oval(2,2)[tr]}
}
\newsavebox{\folderb}
\savebox{\folderb}
(40,32)[l]{% definition
\put(0,14){\line(1,0){8}}
\put(8,0){\usebox{\foldera}}
}
\put(34,26){\line(0,1){102}}
\put(14,128){\usebox{\foldera}}
\multiput(34,86)(0,-37){3}
{\usebox{\folderb}}
\end{picture}

```



Pudełko rysunku można *zadeklarować* instrukcją

```
\newsavebox{nazwa}
```

następnie *zdefiniować* poleceniem

```
\savebox{nazwa}(szerokość,wysokość)[pozycja]{zawartość}
```

i ostatecznie dowolnie często *rysować*, wywołując

```
\put(x,y){\usebox{nazwa}}
```

Opcjonalny parametr *pozycja* definiuje „punkt zaczepienia” zachowywanego pudełka (savebox). W przykładzie nadano mu wartość `bl`, co oznacza umieszczenie punktu zaczepienia w dolnym lewym narożniku pudełka. Pozycje można też oznaczać literami `t` (góra) i `r` (prawo).

Argument *nazwa* odnosi się do L<sup>A</sup>T<sub>E</sub>Xowego magazynu poleceń i dlatego ma naturę instrukcji (co w powyższym przykładzie objawia się choćby użyciem znaków `'\'`). Rysunki przechowywane w pudełkach można zagnieżdżać. W powyższym przykładzie wewnątrz definicji `\folderb` użyto `\foldera`.

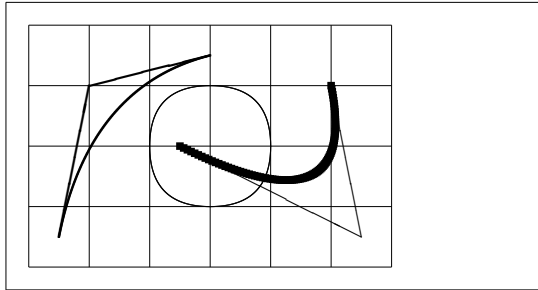
Trzeba było użyć polecenia `\oval`, ponieważ instrukcja `\line` nie działa, gdy długość odcinka wynosi mniej niż około 3 mm.

### 5.2.9 Krzywe Béziera drugiego stopnia

```

\setlength{\unitlength}{0.8cm}
\begin{picture}(6,4)
  \linethickness{0.075mm}
  \multiput(0,0)(1,0){7}
    {\line(0,1){4}}
  \multiput(0,0)(0,1){5}
    {\line(1,0){6}}
  \thicklines
  \put(0.5,0.5){\line(1,5){0.5}}
  \put(1,3){\line(4,1){2}}
  \qBezier(0.5,0.5)(1,3)(3,3.5)
  \thinlines
  \put(2.5,2){\line(2,-1){3}}
  \put(5.5,0.5){\line(-1,5){0.5}}
  \linethickness{1mm}
  \qBezier(2.5,2)(5.5,0.5)(5,3)
  \thinlines
  \qBezier(4,2)(4,3)(3,3)
  \qBezier(3,3)(2,3)(2,2)
  \qBezier(2,2)(2,1)(3,1)
  \qBezier(3,1)(4,1)(4,2)
\end{picture}

```



Jak widać w przykładzie, podzielenie okręgu na cztery krzywe Béziera drugiego stopnia nie daje zadowalającego efektu; lepsze przybliżenie dałoby osiem. Przykład ponownie ilustruje wpływ instrukcji `\linethickness` na linie poziome i pionowe oraz poleceń `\thinlines` i `\thicklines` na odcinki pochyłe. Pokazuje on również, że oba te rodzaje poleceń oddziałują na krzywe Béziera i że kolejne użycie któregośkolwiek z nich przesłania poprzednie.

Niech symbole  $P_1 = (x_1, y_1)$ ,  $P_2 = (x_2, y_2)$  oznaczają punkty końcowe, zaś  $m_1$ ,  $m_2$  – odpowiednie nachylenia krzywej Béziera drugiego stopnia. Pośredni punkt kontrolny  $S = (x, y)$  jest zatem opisany równaniami

$$\begin{cases} rclx = \frac{m_2x_2 - m_1x_1 - (y_2 - y_1)}{m_2 - m_1}, \\ y = y_i + m_i(x - x_i) \quad (i = 1, 2). \end{cases} \quad (5.1)$$

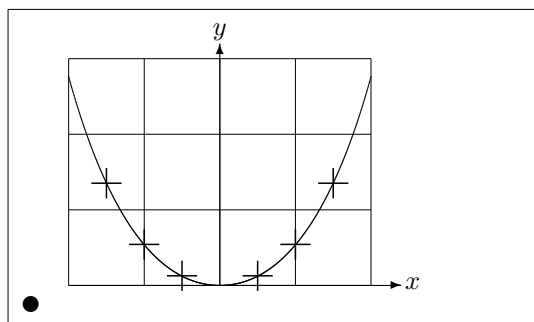
W *Graphics in L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub>* [16] można znaleźć program w Javie, który generuje odpowiednią linię polecenia `\qBezier`.

## 5.2.10 Krzywe łańcuchowe

```

\setlength{\unitlength}{1cm}
\begin{picture}(4.3,3.6)(-2.5,-0.25)
\put(-2,0){\vector(1,0){4.4}}
\put(2.45,-.05){\mathit{x}}
\put(0,0){\vector(0,1){3.2}}
\put(0,3.35){\makebox(0,0){\mathit{y}}}
\qBezier(0.0,0.0)(1.2384,0.0)
(2.0,2.7622)
\qBezier(0.0,0.0)(-1.2384,0.0)
(-2.0,2.7622)
\linethickness{.075mm}
\multiput(-2,0)(1,0){5}
{\line(0,1){3}}
\multiput(-2,0)(0,1){4}
{\line(1,0){4}}
\linethickness{.2mm}
\put(.3,.12763){\line(1,0){.4}}
\put(.5,-.07237){\line(0,1){.4}}
\put(-.7,.12763){\line(1,0){.4}}
\put(-.5,-.07237){\line(0,1){.4}}
\put(.8,.54308){\line(1,0){.4}}
\put(1,.34308){\line(0,1){.4}}
\put(-1.2,.54308){\line(1,0){.4}}
\put(-1,.34308){\line(0,1){.4}}
\put(1.3,1.35241){\line(1,0){.4}}
\put(1.5,1.15241){\line(0,1){.4}}
\put(-1.7,1.35241){\line(1,0){.4}}
\put(-1.5,1.15241){\line(0,1){.4}}
\put(-2.5,-0.25){\circle*{0.2}}
\end{picture}

```



Każdą z symetrycznych połówek wykresu cosinusa hiperbolicznego  $y = \cosh x - 1$  przybliżono na rysunku krzywą Béziera. Prawa połowka krzywej kończy się w punkcie  $(2, 2.7622)$ , w którym nachylenie ma wartość  $m = 3.6269$ . Używając ponownie równania (5.1), możemy wyliczyć pośrednie punkty kontrolne. Okazuje się, że są to:  $(1.2384, 0)$  i  $(-1.2384, 0)$ . Krzyżkami została zaznaczona „prawdziwa” krzywa. Błąd jest ledwie zauważalny, bo wynosi mniej niż jeden procent.

Ten przykład ilustruje również użycie opcjonalnego argumentu otoczenia `\begin{picture}`. Rysunek zdefiniowano w terminach wygodnych współ-

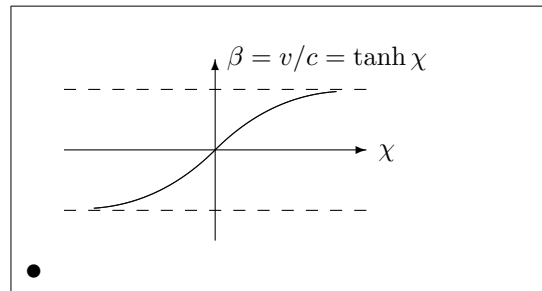
rzędnych „matematycznych”, podczas gdy poleceniem:

```
\begin{picture}(4.3,3.6)(-2.5,-0.25)
```

jego lewemu dolnemu narożnikowi (oznaczonemu czarnym kółeczkiem) przypisano współrzędne  $(-2.5, -0.25)$ .

### 5.2.11 Prędkość w Szczególnej Teorii Względności

```
\setlength{\unitlength}{0.8cm}
\begin{picture}(6,4)(-3,-2)
\put(-2.5,0){\vector(1,0){5}}
\put(2.7,-0.1){$\chi$}
\put(0,-1.5){\vector(0,1){3}}
\multiput(-2.5,1)(0.4,0){13}
{\line(1,0){0.2}}
\multiput(-2.5,-1)(0.4,0){13}
{\line(1,0){0.2}}
\put(0.2,1.4)
{${\beta=v/c=\tanh\chi}$}
\qbezier(0,0)(0.8853,0.8853)
(2,0.9640)
\qbezier(0,0)(-0.8853,-0.8853)
(-2,-0.9640)
\put(-3,-2){\circle*{0.2}}
\end{picture}
```



Punkty kontrolne dwóch krzywych Béziera wyliczono ze wzorów (5.1). Gałąź dodatnia jest określona wartościami  $P_1 = (0, 0)$ ,  $m_1 = 1$  oraz  $P_2 = (2, \tanh 2)$ ,  $m_2 = 1/\cosh^2 2$ . I znowu rysunek wyrażono w wygodnych matematycznie współrzędnych, a lewemu dolnemu narożnikowi przypisano współrzędne  $(-3, -2)$  (czarne kółeczko).

## 5.3 Pakiety PGF i TikZ

W dzisiejszych czasach systemy wyjściowe L<sup>A</sup>T<sub>E</sub>Xa są w stanie stworzyć przyjemną grafikę wektorową, jednak interfejsy do tego są bardzo zróżnicowane. Pakiet `pgf` dostarcza abstrakcyjną warstwę ponad tymi interfejsami. Pakiet `pgf` ma swój własny sporej wielkości podręcznik [18]. Poruszmy tutaj jedynie powierzchowne polecenia z tego pakietu w tej sekcji.

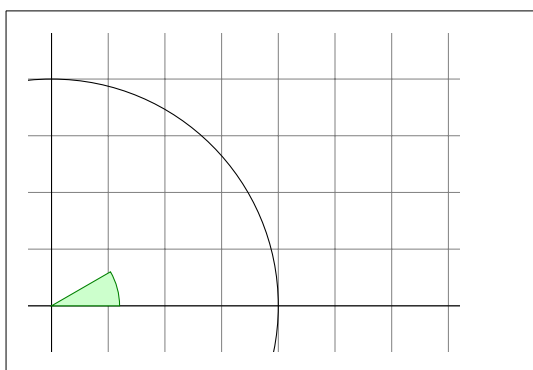


Pakiet `pgf` posiada swój własny język wysokiego poziomu dostarczane przez pakiet `tikz`. TikZ udostępnia bardzo wydajne polecenia do rysowania grafik prosto z twojego dokumentu. Użyj otoczenia `tikzpicture` aby wstawiać polecenia TikZ.

Jak wspomniano powyżej istnieje świetny podręcznik dla pakietu `pgf` i powiązanych pakietów. Zamiast dokładnie opisywać jak on działa pokażemy tylko kilka przykładów by zaprezentować możliwości tego narzędzia.

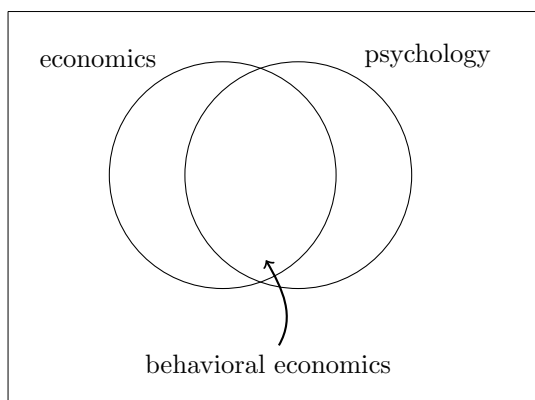
Na początek bezsensowny diagram.

```
\begin{tikzpicture}[scale=3]
  \clip (-0.1,-0.2)
    rectangle (1.8,1.2);
  \draw[step=.25cm,gray,very thin]
    (-1.4,-1.4) grid (3.4,3.4);
  \draw (-1.5,0) -- (2.5,0);
  \draw (0,-1.5) -- (0,1.5);
  \draw (0,0) circle (1cm);
  \filldraw[fill=green!20!white,
    draw=green!50!black]
    (0,0) -- (3mm,0mm)
    arc (0:30:3mm) -- cycle;
\end{tikzpicture}
```



Zwróć uwagę na średnik (;). Oddziela on od siebie polecenia.  
Prosty diagram Venna.

```
\begin{tikzpicture}
  \node[circle,draw,
    minimum size=3cm,
    label=120:{economics}]
    at (0,0) {};
  \node[circle,draw,
    minimum size=3cm,
    label=60:{psychology}]
    at (1,0) {};
  \node (i) at (0.5,-1) {};
  \node at (0.6,-2.5)
    {behavioral economics}
    edge[->,thick,
    out=60,in=-60] (i);
\end{tikzpicture}
```

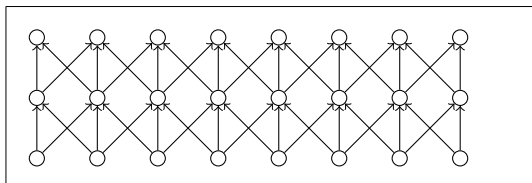


Zwróć uwagę na pętle `foreach` w następnym przykładzie.

```

\begin{tikzpicture}[scale=0.8]
  \tikzstyle{v}=[circle, minimum size=2mm,inner sep=0pt,draw]
  \foreach \i in {1,...,8}
    \foreach \j in {1,...,3}
      \node[v]
        (G-\i-\j) at (\i,\j) {};
  \foreach \i in {1,...,8}
    \foreach \j/\o in {1/2,2/3}
      \draw[->]
        (G-\i-\j) -- (G-\i-\o);
  \foreach \i/\n in
    {1/2,2/3,3/4,4/5,5/6,6/7,7/8}
    \foreach \j/\o in {1/2,2/3} {
      \draw[->] (G-\i-\j) -- (G-\n-\o);
      \draw[->] (G-\n-\j) -- (G-\i-\o);
    }
}
\end{tikzpicture}

```

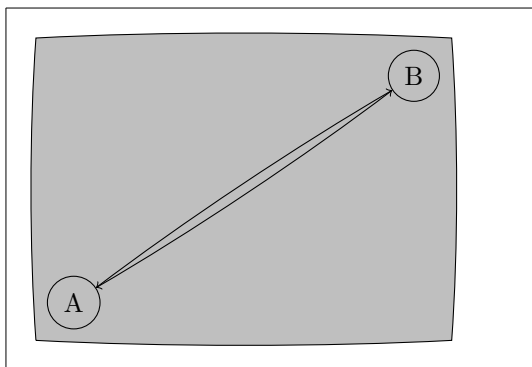


Przy użyciu polecenia `\usetikzlibrary` w preambule możemy włączyć dodatkowe funkcje do rysowania specjalnych kształtów, jak na przykład poniższe pudełko, które jest lekko wygięte.

```

\usetikzlibrary{%
  decorations.pathmorphing}
\begin{tikzpicture}[
  decoration={bent,aspect=.3}]
\draw [decorate,fill=lightgray]
  (0,0) rectangle (5.5,4);
\node[circle,draw]
  (A) at (.5,.5) {A};
\node[circle,draw]
  (B) at (5,3.5) {B};
\draw[->,decorate] (A) -- (B);
\draw[->,decorate] (B) -- (A);
\end{tikzpicture}

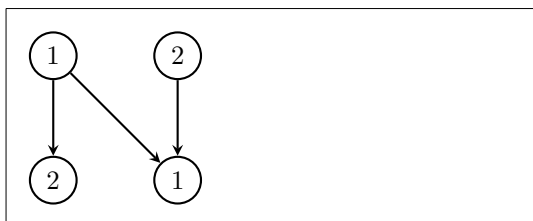
```



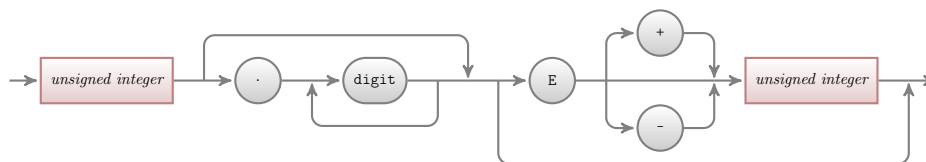
```

\usetikzlibrary{positioning}
\begin{tikzpicture}[xscale=6,
  yscale=8,>=stealth]
  \tikzstyle{v}=[circle,
    minimum size=1mm,draw,thick]
  \node[v] (a) {$1$};
  \node[v] (b) [right=of a] {$2$};
  \node[v] (c) [below=of a] {$2$};
  \node[v] (d) [below=of b] {$1$};
  \draw[thick,->]
    (a) to node {} (c);
  \draw[thick,->]
    (a) to node {} (d);
  \draw[thick,->]
    (b) to node {} (d);
\end{tikzpicture}

```



Możesz nawet rysować diagramy składni, które wyglądają jakby zostały wyrwane prosto z książki o programowaniu w Pascalu. Kod jest nieco trudniejszy niż przykłady powyżej, więc zaprezentujemy tylko wynik końcowy. Jeśli zerkniesz to dokumentacji `pgf` znajdziesz tam szczegółowy poradnik rysowania tego dokładnie diagramu.



Jest tego o wiele więcej. Jeśli musisz narysować wykres danych numerycznych lub funkcji, to przyjrzyj się pakietowi `pgfplot`. Zapewnia on wszystko czego potrzeba do rysowania wykresów. Potrafi on nawet uruchomić zewnętrzny program `gnuplot` by wyliczyć faktyczne funkcje, które wpisano w wykres.

Więcej informacji możesz znaleźć na znakomitej stronie <http://www.texample.net/tikz/> autorstwa Kjell Magne Fauske. Zawiera ona wciąż rozszerzającą się kolekcję pięknych wykresów i innego kodu  $\LaTeX$ a. Na `TExample.net` znajdziesz również [listę narzędzi do pracy z PGF/TikZ](#) dzięki którym unikniesz pisania całego tego kodu ręcznie.



## Rozdział 6

# Adaptowanie L<sup>A</sup>T<sub>E</sub>Xa

Dokumenty składane za pomocą poznanych do tej pory poleceń będą się zapewne podobały zdecydowanej większości czytelników. Chociaż ich wygląd nie będzie może wyrafinowany, z pewnością jednak spełnią one podstawowe zasady składu, dzięki czemu będzie się je czytało łatwo i przyjemnie.

W niektórych wypadkach może się jednak okazać, że brakuje polecenia czy otoczenia, za pomocą którego moglibyśmy złożyć dany fragment tekstu w sposób odpowiadający potrzebom, albo też że sposób działania dostępnej w L<sup>A</sup>T<sub>E</sub>Xu instrukcji nie spełnia naszych wymagań.

W tym rozdziale przedstawimy, jak nauczyć L<sup>A</sup>T<sub>E</sub>Xa formatować dokumenty tak, aby wyglądały inaczej niż w wypadku korzystania jedynie ze standardowych klas i pakietów.

### 6.1 Definiowane instrukcji i otoczeń

Czytelnicy zauważyli zapewne, że nowo wprowadzane w tej książce polecenia ukazują się w ramkach oraz że znajdują się one w skorowidzu. Aby to osiągnąć, nie korzystaliśmy za każdym razem z wbudowanych w L<sup>A</sup>T<sub>E</sub>Xa instrukcji, lecz utworzyliśmy własny pakiet, w którym zawarliśmy nowe, potrzebne nam polecenia i otoczenia. Dysponując takim pakietem, wystarczy po prostu napisać:

```
\begin{lscommand}  
\ci{dum}  
\end{lscommand}
```



```
\dum
```

W tym przykładzie użyliśmy zarówno zdefiniowanego otoczenia o nazwie `lscmmand`, odpowiedzialnego za rysowanie ramek dookoła instrukcji, jak i nowego polecenia `\ci`, służącego do składu nazw poleceń i wprowadzania ich do skorowidza. Proponujemy Czytelnikom odszukanie hasła `\dum` w skorowidzu; przy hasle powinny być podane numery stron, na których ta instrukcja występuje w książce.

Jeśli kiedyś zdecydujemy się zaprzestać otaczać polecenia ramkami, to do zmiany wyglądu wystarczy, że zmienimy definicję otoczenia `command`. Jest to znacznie łatwiejsze od przebiegnięcia przez cały dokument w celu wyłapania w nim wszystkich standardowych poleceń L<sup>A</sup>T<sub>E</sub>Xa, które służą do rysowania ramek wokół słów.

### 6.1.1 Nowe polecenia

Do definiowania potrzebnych nam nowych poleceń możemy użyć instrukcji

```
\newcommand{nazwa} [liczba] {definicja}
```

Wymaga ona podania dwóch argumentów. Pierwszy z nich, *nazwa*, oznacza nazwę nowej instrukcji, natomiast *definicja* to jej znaczenie, czyli tekst, który ma zostać wstawiony do składu w momencie wykonania instrukcji. Podawany w nawiasach kwadratowych argument *liczba* powinien być cyfrą od 1 do 9, określającą liczbę (obowiązkowych) argumentów instrukcji. Argument *num* jest opcjonalny, a jego pominięcie oznacza, że definiowana instrukcja jest bezargumentowa.

Następujące przykłady pomogą lepiej zrozumieć zagadnienie. W pierwszym z nich definiujemy instrukcję o nazwie `\kwle`, mającą być skrótem dla słów „Nie za krótkie wprowadzenie do systemu L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub>”. Takie polecenie mogłoby się przydać, gdyby tytuł książki miał w niej występować wielokrotnie.

```
\newcommand{\kwle}{Nie za
krótkie wprowadzenie
do systemu \LaTeXe}
Oto ,, \kwle''\ldots{}
,, \kwle''
```

```
Oto „Nie za krótkie wprowadzenie do sys-
temu LATEX 2ε”... „Nie za krótkie wprowad-
zenie do systemu LATEX 2ε”
```

Następny przykład ilustruje sposób wykorzystania opcjonalnego argumentu *liczba*. Znacznik `#1` oznacza pierwszy parametr formalny, `#2` oznacza drugi (`#3` oznaczałby trzeci itd.).

```
\newcommand{\txsit}[2]
{Oto \emph{#1}
 #2 wprowadzenie
 do systemu \LaTeXe}
% w ciele dokumentu:
\begin{itemize}
\item \txsit{Nie za}{krótkie}
\item \txsit{Bardzo}{długie}
\end{itemize}
```

- Oto *Nie za* krótkie wprowadzenie do systemu  $\text{\LaTeX} 2_{\epsilon}$
- Oto *Bardzo* długie wprowadzenie do systemu  $\text{\LaTeX} 2_{\epsilon}$

$\text{\LaTeX}$  nie pozwala zdefiniować instrukcji nazwanej tak samo jak wcześniej zdefiniowane polecenie. W wypadku gdy chcemy zmienić znaczenie już istniejącej instrukcji, powinniśmy użyć polecenia `\renewcommand`. Za wyjątkiem nazwy ma ono składnię identyczną jak `\newcommand`.

Czasami może się też przydać polecenie `\providecommand`. Działa ono jak `\newcommand`, z tym że jeśli istnieje już komenda o takiej samej nazwie, to nie zastępuje ono starej, zwyczajnie ignorując nową.

Nieco uwagi należy poświęcić temu, jaki skutek ma umieszczenie odstępów po komendzie  $\text{\LaTeX}$ a. Informacje na ten temat można znaleźć na stronie ??.

### 6.1.2 Nowe otoczenia

Odpowiednikiem definiującej nowe polecenie instrukcji `\newcommand` jest dla otoczeń instrukcja `\newenvironment`. Ma ona następującą składnię:

```
\newenvironment{nazwa}[liczba]{przed}{po}
```

Podobnie jak w przypadku `\newcommand`, `\newenvironment` można użyć z argumentem opcjonalnym albo bez niego.  $\text{\LaTeX}$  wstawia tekst *przed*, gdy w dokumencie napotyka napis `\begin{nazwa}`, a zawartość argumentu *po* – po napotkaniu napisu `\end{nazwa}`.

Poniższy przykład ilustruje sposób użycia instrukcji `\newenvironment`.

```
\newenvironment{król}
{\rule{1ex}{1ex}%
 \hspace{\stretch{1}}}
{\hspace{\stretch{1}}%
 \rule{1ex}{1ex}}
```

```
■                   Moi wierni poddani...                   ■
```

```
\begin{król}
Moi wierni poddani\ldots
\end{król}
```

Znaczenie argumentu *num* jest takie samo jak w instrukcji `\newcommand`. L<sup>A</sup>T<sub>E</sub>X nie pozwala zdefiniować otoczenia o już istniejącej nazwie. W razie potrzeby zastąpienia już istniejącego otoczenia powinniśmy użyć polecenia `\renewenvironment`, o składni takiej samej jak `\newenvironment`.

Polecenia użyte w powyższym przykładzie są objaśniane dalej. O poleceniu `\rule` można przeczytać na stronie 141, o `\stretch` – na stronie 134, a informacje na temat `\hspace` można znaleźć na stronie 133.

### 6.1.3 Nadmiarowe odstępy

Gdy tworzymy nowe otoczenie, problemem mogą być niechciane przez nas odstępy, które L<sup>A</sup>T<sub>E</sub>X wstawia do składu. Rozważmy przykład otoczenia, które ma się rozpoczynać od akapitu bez wcięcia, ponadto pierwszy akapit po otoczeniu także nie ma mieć wcięcia. Polecenie `\ignorespaces`, umieszczone jako ostatnie polecenie bloku *begin* otoczenia, spowoduje zignorowanie wszystkich odstępów występujących przed pierwszym akapitem otoczenia. Usunięcie drugiego wcięcia przez umieszczenie w bloku końcowym `\ignorespaces` jest niemożliwe, gdyż zawsze ostatnim poleceniem będzie `\end{otoczenie}`, które anuluje działanie `\ignorespaces`. W takiej sytuacji trzeba skorzystać z polecenia `\ignorespacesafterend`. Napotkawszy je, L<sup>A</sup>T<sub>E</sub>X wstawi `\ignorespaces` dopiero po wykonaniu zamykającego `\end{otoczenie}`.

```
\newenvironment{proste}%
  {\noindent}%
  {\par\noindent}

\begin{proste}
Zobacz odstęp\\z lewej strony.
\end{proste}
Tak samo\\tutaj.
```

Zobacz odstęp  
z lewej strony.

Tak samo  
tutaj.

```
\newenvironment{poprawne}%
  {\noindent\ignorespaces}%
  {\par\noindent%
   \ignorespacesafterend}

\begin{poprawne}
Bez odstepu\\z lewej strony.
\end{poprawne}
Tak samo\\tutaj.
```

Bez odstepu  
z lewej strony.

Tak samo  
tutaj.



### 6.1.4 Wywoływanie L<sup>A</sup>T<sub>E</sub>Xa z linii poleceń

Użytkownicy systemów operacyjnych typu Unix do obsługi L<sup>A</sup>T<sub>E</sub>Xowych projektów stosują zapewne pliki Makefile. W tym kontekście ciekawa byłaby możliwość tworzenia różnych wersji tego samego dokumentu metodą podawaniu różnych parametrów w linii poleceń wołającej L<sup>A</sup>T<sub>E</sub>Xa. Możemy to ułatwić, umieszczając w dokumencie coś takiego:

```
\usepackage{ifthen}
\ifthenelse{\equal{\blackandwhite}{true}}{
  % coś robimy w trybie "czarno--białym"
}{
  % robimy coś innego w trybie "kolorowym"
}
```

Możemy teraz wywołać L<sup>A</sup>T<sub>E</sub>Xa w następujący sposób:

```
xelatex '\newcommand{\blackandwhite}{true}\input{test.tex}'
```

Polecenie `\blackandwhite` zostaje zdefiniowane jeszcze przed odczytem pliku źródłowego. Zmieniając `\blackandwhite` na `\false` spowodujemy generowanie kolorowej wersji dokumentu.

### 6.1.5 Własne pakiety

W wypadku definiowania wielu nowych poleceń i otoczeń preambuła dokumentu może się znacznie wydłużyć. Dobrze w takiej sytuacji stworzyć pakiet zawierający definicje tych instrukcji i otoczeń. Taki pakiet można później dołączyć do dokumentu poleceniem `\usepackage`.

---

```
% Przykładowy pakiet ***
\ProvidesPackage{demopack}
\newcommand{\kwle}{Nie za krótkie wprowadzenie do systemu \LaTeXe}
\newcommand{\wle}[1]{\emph{#1} wprowadzenie
do systemu \LaTeXe}
\newenvironment{król}{\begin{quote}}{\end{quote}}
```

---

Rysunek 6.1: Przykładowy pakiet.

Tworzenie pakietu polega na skopiowaniu poleceń z preambuły do oddzielnego pliku o rozszerzeniu `.sty`. Jest tylko jedno specjalne polecenie,

które należy wpisać na początku pakietu

```
\ProvidesPackage{nazwa pakietu}
```

Dzięki instrukcji `\ProvidesPackage` L<sup>A</sup>T<sub>E</sub>X poznaje nazwę pakietu, a to pozwala mu na przykład ostrzec użytkownika w wypadku powtórnego dołączenia pakietu do dokumentu. Rysunek 6.1 przedstawia niewielki pakiet z instrukcjami z powyższych przykładów.

## 6.2 Fonty i rozmiary

### 6.2.1 Polecenia do zmiany fontu

L<sup>A</sup>T<sub>E</sub>X automatycznie wybiera odpowiedni font i jego rozmiar

na podstawie logicznej struktury dokumentu (tytułów rozdziałów, sekcji, przypisów itp.). Czasami zachodzi jednak chęć „ręcznie” przełączyć krój bądź stopień pisma. Można do tego użyć poleceń zestawionych w tabelach 6.1 i 6.2. Rozmiar fontu jest kwestią układu graficznego dokumentu i zależy od wybranej klasy dokumentu oraz ustawienia odpowiednich opcji. W tabeli ?? zestawiono stopnie pisma w jednostkach absolutnych dla poleceń zmieniających wielkość kroju w standardowych klasach dokumentów.

```
{\small Nieliczni lecz  
\textbf{odważni}  
Rzymianie rządząli}  
{\Large wielką \textit{Italia}}.}
```

Nieliczni lecz **odważni** Rzymianie rządząli  
wielką *Italia*.

Jedną z istotnych funkcji L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub> jest fakt, że własności fontów są niezależne. Oznacza to, że zmiana rozmiaru bądź fontu przez polecenia nie zmienia ustawionych wcześniej własności takich jak pochylenie czy pogrubienie.

W *trybie matematycznym* korzystanie z *poleceń* zmiany fontu chwilowo opuszcza *tryb matematyczny* i wstawia normalny tekst. Aby zmienić font do składania matematyki należy użyć specjalnych poleceń zestawionych w tabeli 6.4.

Przy okazji omawiania poleceń dotyczących fontów trzeba wspomnieć o koncepcji *grupowania*. Grupa zaczyna się od znaku {, a kończy znakiem }. Grupy służą do ograniczania zasięgu działania poleceń L<sup>A</sup>T<sub>E</sub>Xa.

```
Lubię {\LARGE duże oraz  
\small małe} litery} i cyfry.
```

Lubię **duże** oraz małe **litery**  
i cyfry.

Tabela 6.1: Fonty.

<code>\textrm{...}</code>	szeryfowe	<code>\textsf{...}</code>	bezseryfowe
<code>\texttt{...}</code>	maszynowe		
<code>\textmd{...}</code>	jasne	<code>\textbf{...}</code>	<b>pogrubione</b>
<code>\textup{...}</code>	proste	<code>\textit{...}</code>	<i>kursywa</i>
<code>\textsl{...}</code>	<i>pochylone</i>	<code>\textsc{...}</code>	KAPITALIKI
<code>\textnormal{...}</code>	font dokumentu		

Tabela 6.2: Rozmiary fontów.

<code>\tiny</code>	mikroskopijny	<code>\Large</code>	wiekszy
<code>\scriptsize</code>	bardzo mały	<code>\LARGE</code>	bardzo duży
<code>\footnotesize</code>	mniejszy	<code>\huge</code>	ogromny
<code>\small</code>	mały	<code>\Huge</code>	największy
<code>\normalsize</code>	normalny		
<code>\large</code>	duży		

Tabela 6.3: Absolutny rozmiar fontów w klasach standardowych.

Rozmiar	10pt (domyślny)	opcja 11pt	opcja 12pt
<code>\tiny</code>	5pt	6pt	6pt
<code>\scriptsize</code>	7pt	8pt	8pt
<code>\footnotesize</code>	8pt	9pt	10pt
<code>\small</code>	9pt	10pt	11pt
<code>\normalsize</code>	10pt	11pt	12pt
<code>\large</code>	12pt	12pt	14pt
<code>\Large</code>	14pt	14pt	17pt
<code>\LARGE</code>	17pt	17pt	20pt
<code>\huge</code>	20pt	20pt	25pt
<code>\Huge</code>	25pt	25pt	25pt

Polecenia zmieniające rozmiar fontu zmieniają także interlinię. Dzieje się tak jednak tylko wtedy, gdy przed zamknięciem odpowiedniej grupy kończony jest akapit. Zwróćmy uwagę na miejsce, w którym umieszczono instrukcję `\par` w poniższych dwóch przykładach<sup>1</sup>.

```
{\Large Nie czytaj tego!  
To nieprawda.  
Możesz mi zaufać!\par}
```

Nie czytaj tego! To nieprawda. Możesz mi zaufać!

```
{\Large To też jest nieprawda.  
Ale pamiętaj, że jestem kłamcą.}  
\par
```

To też jest nieprawda. Ale pamiętaj, że jestem kłamcą.

Jeśli zachodzi konieczność zmiany stopnia pisma dla całego akapitu lub jeszcze dłuższego tekstu, to możemy skorzystać ze składni przyjętej dla otoczeń.

```
\begin{Large}  
To nie jest prawdziwe.  
Ale cóż jest w dzisiejszych  
czasach\ldots  
\end{Large}
```

To nie jest prawdziwe. Ale cóż jest w dzisiejszych czasach...

Zapis taki pozwala unikać liczenia nawiasów klamrowych.

<sup>1</sup>`\par` jest równoważny wstawieniu pustej linii

Tabela 6.4: Fonty matematyczne.

<code>\mathrm{...}</code>	Font Prosty
<code>\mathbf{...}</code>	<b>Font Pogrubiony</b>
<code>\mathsf{...}</code>	Font bezszeryfowy
<code>\mathtt{...}</code>	Font maszynowy
<code>\mathit{...}</code>	<i>Kursywa</i>
<code>\mathcal{...}</code>	FONT KALIGRAFICZNY
<code>\mathnormal{...}</code>	<i>Normalny Font</i>

### 6.2.2 Uważaj! Tu mogą być promile

Jak zaznaczyliśmy na początku tego rozdziału, nie należy instrukcji zmiany fontu wstawiać bezpośrednio do pliku źródłowego. Byłoby to niezgodne z podstawową ideą L<sup>A</sup>T<sub>E</sub>Xa, jaką jest oddzielenie formy od treści dokumentu i posługiwanie się formatowaniem logicznym, a nie wizualnym. Jeżeli fragment tekstu ma zostać wyróżniony przez złożenie go innym krojem lub stopniem pisma, to należy zdefiniować odpowiednie polecenie i potem właśnie jego używać w treści dokumentu.

```
\newcommand{\ups}[1]{%
  \textbf{#1}}
Nie \ups{wchodzić} do pokoju,
jest zajęty przez \ups{maszyny}
nieznanego pochodzenia.
```

Nie **wchodzić** do pokoju, jest zajęty przez **maszyny** nieznanego pochodzenia.

Niewątpliwą zaletą tego podejścia jest to, że kiedy później będziemy chcieli wyróżnić wszystkie elementy, na które czytelnik powinien zwrócić *szczególną uwagę*, w sposób inny niż składając je pismem pogrubionym, to nie musimy przeglądać całego pliku w celu sprawdzenia, czy dane wystąpienie `\textbf` dotyczy tekstu, na który ma zostać zwrócona *szczególna uwaga*, czy też wstawione zostało w zupełnie innym celu.

Zwróć uwagę, że poinstruowanie L<sup>A</sup>T<sub>E</sub>Xa do *wyróżnienia* czegoś jest różne od poinstruowania użycia innego *fontu*. Poleceni `\emph` działa inaczej w zależności od kontekstu, podczas gdy polecenia zmiany fontu są absolutne.

```
\textit{Możesz również
  \emph{wyróżnić} tekst
  napisany kursywą,}
\textsf{w foncie
  \emph{bezseryfowym},}
\texttt{jak i również
  \emph{maszynowym}.}
```

*Możesz również wyróżnić tekst napisany kursywą, w foncie bezseryfowym, jak i również maszynowym.*

### 6.2.3 Rada

Na zakończenie naszej podróży w świat fontów i ich rozmiarów, zostawiamy Was z drobną radą:

**Pamiętaj!** *Im* **Wi**ęcej fontów **użyjesz** w dokumencie, *tym* bardziej **CZYTELNY** i *piękny się staje*.

## 6.3 Odstępy

### 6.3.1 Interlinia

Jeśli chcesz używać większej interlinii w dokumencie, to możesz zmienić jej wartość przy użyciu polecenia

```
\linespread{factor}
```

w preambule dokumentu. Wstaw `\linespread{1.3}` aby otrzymać interlinię „jeden i pół”, a `\linespread{1.6}` dla „podwójnej” interlinii. Zwykła interlinia ma wartość 1.

Polecenie `\linespread` wpływa na odstępy międzywierszowe w całym dokumencie. Jeśli są wyraźne powody do zmiany odstępu tylko w pewnym fragmencie dokumentu, to lepsza może się okazać instrukcja:

```
\setlength{\baselineskip}{1.5\baselineskip}
```

```
{\setlength{\baselineskip}%
  {1.5\baselineskip}
Ten akapit jest składany
z \ci{baselineskip} ustawionym
na 1,5 dotychczasowej wartości.
Zwróćmy uwagę na wystąpienie
\ci{par} na końcu akapitu.\par}
```

Przeznaczenie tego akapitu jest jasne. Ilustruje on, że po zamykającym nawiasie klamrowym następuje powrót do normalnego składu.

Ten akapit jest składany z `\baselineskip` ustawionym na 1,5 dotychczasowej wartości. Zwróćmy uwagę na wystąpienie `\par` na końcu akapitu.

Przeznaczenie tego akapitu jest jasne. Ilustruje on, że po zamykającym nawiasie klamrowym następuje powrót do normalnego składu.

### 6.3.2 Formatowanie akapitów

Dwa dodatkowe parametry określają w L<sup>A</sup>T<sub>E</sub>Xu wielkość, odpowiednio, wcięcie akapitowego oraz odstępu między akapitami. Wpisując na przykład do preambuły dokumentu

```
\setlength{\parindent}{0pt}
\setlength{\parskip}{1ex plus 0.5ex minus 0.2ex}
```

możemy zmienić układ akapitów. Te dwa polecenia zmieniają odstęp pomiędzy dwoma akapitami, a cięcie ustawiają na zero.

`plus` i `minus` w długości powyżej instruuja  $\TeX$ a, że dozwolone jest zwiększenie i zmniejszenie odstępów jeśli jest to konieczne do idealnego składu akapitów na stronie.

W niektórych krajach Europejskich, akapity są często oddzielane przy użyciu odstepu i nie mają wcięć. Zwróć jednak uwagę, że takie ustawienie wpływa również na spis treści przez co on również jest nieco obszerniejszy. Aby tego uniknąć możesz przenieść oba polecenia z preambuły do miejsca w dokumencie pod poleceniem `\tableofcontents` lub po prostu korzystać z domyślnego oddzielania akapitów, tak jak w większości profesjonalnych książek.

Jeśli chcesz wciąć akapit który nie jest wcięty, użyj

```
\indent
```

na początku akapitu<sup>2</sup>. Oczywiście, przyniesie to tylko pożądaną efekt pod warunkiem, że `\parindent` nie jest ustawione na zero.

Aby uzyskać akapit bez wcięcia, należy przed nim umieścić polecenie

```
\noindent
```

Może to być użyteczne gdy zaczynasz dokument od tekstu, a nie od polecenia wstawiającego sekcje.

### 6.3.3 Odstępy poziome

Wielkość odstępów między słowami oraz między zdaniem  $\LaTeX$  ustala automatycznie. Dodatkowy odstęp poziomy możemy wstawić przy pomocy

```
\hspace{odległość}
```

Jeżeli taki odstęp, w wyniku złamania akapitu na wiersze, wypadnie na początku lub na końcu wiersza, to zostanie on usunięty – aby zapobiec justowaniu akapitu „w chorągiewkę”. Jeżeli  $\LaTeX$  ma wstawić odstęp także na początku lub na końcu wiersza, to zamiast `\hspace` należy użyć „gwiazdkowej” wersji `\hspace*`. Argument *odległość* oznacza wymiar  $\LaTeX$ owy. W najprostszej postaci jest to liczba wraz z jednostką odległości. Wykaz

<sup>2</sup>Dla uzyskania efektu wcięcia w pierwszym akapicie po tytule rozdziału, sekcji itd. należy dołączyć do dokumentu pakiet `indentfirst` z zestawu pakietów „tools”.

ważniejszych spośród dostępnych w L<sup>A</sup>T<sub>E</sub>Xu jednostek odległości znajduje się w tabeli 6.5.

To jest `\hspace{1.5cm}` odstęp  
równy 1,5~cm.

To jest	odstęp równy 1,5 cm.
---------	----------------------

Polecenie

<code>\stretch{n}</code>
--------------------------

wstawia specjalny rozciągliwy odstęp, który potrafi wypełnić całą wolną przestrzeń w pionie lub w poziomie. Jeżeli na przykład wstawimy w wierszu dwa lub więcej poleceń `\hspace{\stretch{n}}`, to odstępy dzięki nim uzyskane będą miały wielkość według proporcji zadanych przez argument  $n$ .

`x\hspace{\stretch{1}}`  
`x\hspace{\stretch{3}}x`

x	x	x
---	---	---

Wielkość odstępów towarzyszących tekstowi warto dostosować do aktualnego rozmiaru czcionki. Można do tego użyć względnych jednostek miary `em` oraz `ex`:

`{\Large}duż\hspace{1em}e\`  
`{\tiny}mał\hspace{1em}e`

duż	e
mał	e

### 6.3.4 Odstępy pionowe

Odstępy pionowe między akapitami, rozdziałami, sekcjami itp. L<sup>A</sup>T<sub>E</sub>X wstawia automatycznie. Jeśli zachodzi potrzeba wstawienia dodatkowego odstępu pionowego *pomiędzy dwoma akapitami*, to należy zastosować polecenie:

<code>\vspace{odległość}</code>
---------------------------------

Polecenie to należy oddzielić pustymi liniami od otaczającego je tekstu. Jeżeli w wyniku złamania strony odstęp taki znajdzie się na początku lub na końcu strony (będzie zaczynał lub też kończył kolumnę tekstu), to zostanie on usunięty. Jeżeli ma zostać wstawiony także na początku lub końcu strony,



Tabela 6.5: Jednostki  $\TeX$ a.

---

mm	milimetr $\approx 1/25$ cala	⊐
cm	centymetr = 10 mm	┌───┐
in	cal = 25,4 mm	┌──────────┐
pt	punkt $\approx 1/72$ cala $\approx \frac{1}{3}$ mm	⊐
em	$\approx$ szerokość litery „M” w bieżącym foncie	┌┐
ex	$\approx$ wysokość litery „x” w bieżącym foncie	┌┐

---

to należy użyć wersji „gwiazdkowej” `\vspace*`. Argument *odległość* oznacza  $\LaTeX$ owy wymiar.

Do rozmieszczania tekstu w pionie można używać polecenia `\stretch`, łącznie z `\pagebreak`. Dzięki temu można uzyskać tekst w ostatnim wierszu strony lub wyśrodkowany pionowo.

Jakiś tekst\ldots

`\vspace{\stretch{1}}`

To będzie w ostatniej linii strony.`\pagebreak`

Dodatkowy odstęp między dwoma wierszami *tego samego* akapitu lub między wierszami tabeli możemy uzyskać poleceniem

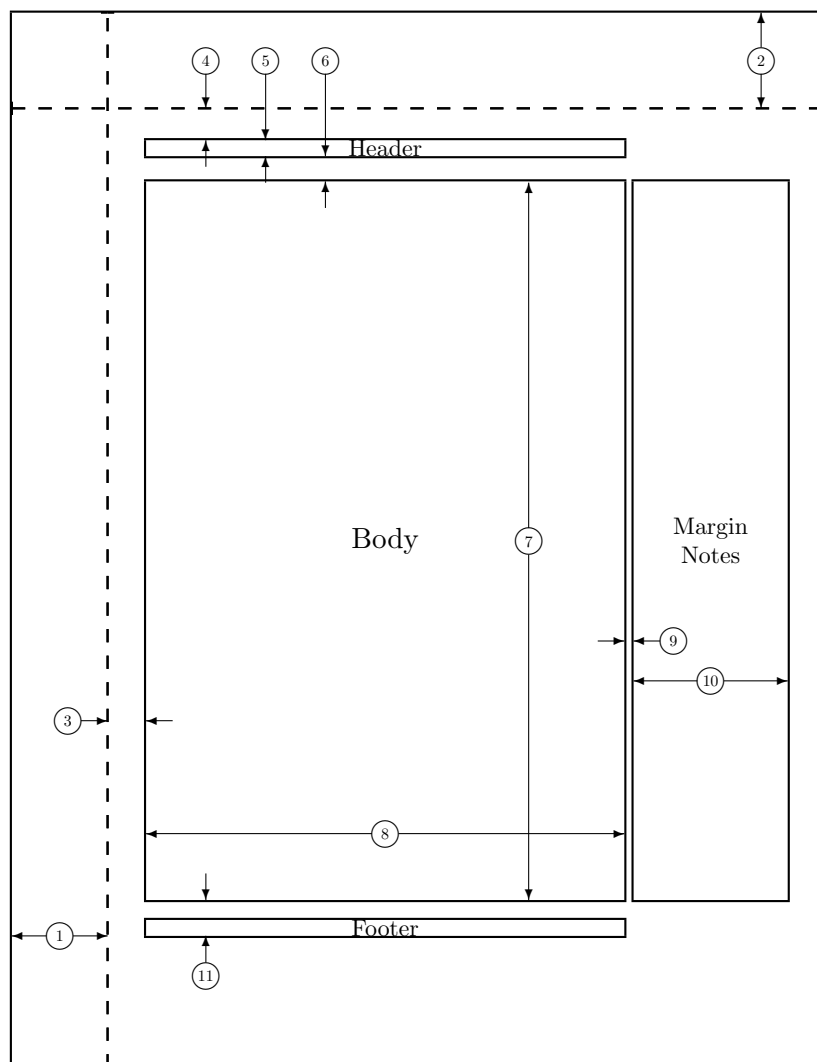
`\[odległość]`

Polecenia `\bigskip`, `\medskip` i `\smallskip` wstawiają odpowiednio odstępy „elastyczne” o następujących wielkościach:  $12\text{pt} \pm 4\text{pt}$ ,  $6\text{pt} \pm 2\text{pt}$  oraz  $3\text{pt} \pm 1\text{pt}$ .

## 6.4 Układ graficzny strony

Wymiary papieru można podać jako argumenty instrukcji `\documentclass`. Na podstawie zadeklarowanych wymiarów  $\LaTeX$  oblicza szerokość i wysokość kolumny, marginesy i inne parametry. Na rysunku 6.2 przedstawiono dostępne parametry graficznego układu strony. Do przygotowania rysunku użyliśmy pakietu `layout` z zestawu „tools”<sup>3</sup>. Jeżeli obliczone przez  $\LaTeX$ a wartości są z pewnych względów nieodpowiednie, to można je zmienić.

<sup>3</sup>[CTAN://macros/latex/packages/tools](http://CTAN://macros/latex/packages/tools).



1	one inch + \hoffset	2	one inch + \voffset
3	\oddsidemargin = 29pt lub \evensidemargin	4	\topmargin = 24pt
5	\headheight = 12pt	6	\headsep = 19pt
7	\textheight = 541pt	8	\textwidth = 360pt
9	\marginparsep = 7pt	10	\marginparwidth = 116pt
11	\footskip = 27pt		\marginparpush = 5pt (not shown)
	\hoffset = 0pt		\voffset = 0pt
	\paperwidth = 614pt		\paperheight = 794pt

Rysunek 6.2: Argumenty układu dla tej książki. Wypróbuj pakiet layouts aby wyświetlić układ swojego własnego dokumentu.

**UWAGA!**...Zanim zacniemy eksperymentować, zmniejszając na przykład szerokość marginesu, chwilę pomyślmy. Za większością wartości domyślnych ustawień w  $\text{\LaTeX}$ u stoją ważne powody. Nie inaczej jest z szerokością marginesu.

Z pewnością w porównaniu z wydrukiem przygotowanym za pomocą świeżo kupionego programu MS Word strona  $\text{\LaTeX}$ owa ma denerwująco szeroki margines. Ale spójrz na Twoją ulubioną książkę<sup>4</sup> i policzmy na niej liczbę znaków w przeciętnym wierszu. Okazuje się, że wynosi ona około 66. Jeśli porównamy ją z wydrukiem złożonym przez  $\text{\LaTeX}$ a, to zapewne i tym razem będzie ona zbliżona do 66. Z doświadczeń wynika bowiem, że w miarę wzrostu liczby znaków w wierszu czytanie staje się męczące. Dzieje się tak, gdyż przy długich wierszach naszym oczom trudniej jest przenosić wzrok z końca jednego wiersza na początek następnego. Jest to jedna z przyczyn stosowania składu wielolamowego w gazetach i czasopismach.

Tak więc, jeśli zwiększamy szerokość kolumny, to pamiętajmy, że może to utrudnić odbiorcom czytanie naszej pracy. No, ale dość już kazań. Obiecaliśmy przecież wyjaśnić, jak można te rzeczy robić...

W  $\text{\LaTeX}$ u mamy dwie instrukcje do zmiany wielkości wymiarów, używane zazwyczaj w obrębie preambuły dokumentu.

Pierwsza z nich nadaje parametrowi określoną wielkość:

```
\setlength{parametr}{wielkość}
```

Drugie polecenie zwiększa wartość parametru o określoną wielkość:

```
\addtolength{parametr}{wielkość}
```

Z tej pary częściej stosowana jest druga instrukcja, ponieważ pozwala zmieniać wymiary. Przykładowo, aby zwiększyć szerokość tekstu o jeden centymetr, umieszczamy w preambule dokumentu następujące polecenia:

```
\addtolength{\hoffset}{-0.5cm}
\addtolength{\textwidth}{1cm}
```

Warto używać tutaj pakietu `calc`. Pozwala on wykonywać operacje arytmetyczne w argumentach do `\setlength` oraz innych miejsc gdzie wprowadzane są wartości numeryczne.

<sup>4</sup>Mamy tu na myśli prawdziwą wydrukowaną książkę z renomowanego wydawnictwa.

## 6.5 Więcej o odległościach

Kiedy to tylko możliwe, unikajmy stosowania wymiarów zdefiniowanych w jednostkach absolutnych, takich jak punkty czy milimetry. Starajmy się raczej odnosić wymiary do już istniejących, takich jak wysokość czy szerokość kolumny. Dla rysunków może to być szerokość linii tekstu by mogły wypełnić określoną część strony.

Następujące trzy polecenia pozwalają określić szerokość, wysokość i głębokość napisu.

```
\settoheight{nazwa}{napis}
\settodepth{nazwa}{napis}
\settowidth{nazwa}{napis}
```

Przykład poniżej ilustruje zastosowanie tych poleceń.

```
\newenvironment{vardesc}[1]{%
  \settowidth{\parindent}{#1:\ }
  \makebox[0pt][r]{#1:\ }}{}
```

```
\begin{displaymath}
a^2+b^2=c^2
\end{displaymath}
```

```
\begin{vardesc}{gdzie}$a$,
$b$ -- są bokami stykające
się z kątem prostym.
```

```
$c$ -- jest bokiem naprzeciwko
kąta prostego i jest samotny.
```

```
$d$ -- który nie pokazuje się
tutaj w ogóle.
```

```
Zagadkowe, czyż nie?
```

```
\end{vardesc}
```

$$a^2 + b^2 = c^2$$

gdzie:  $a$ ,  $b$  – są bokami stykające się z kątem prostym.

$c$  – jest bokiem naprzeciwko kąta prostego i jest samotny.

$d$  – który nie pokazuje się tutaj w ogóle. Zagadkowe, czyż nie?

## 6.6 Pudełka

Każdą stronę L<sup>A</sup>T<sub>E</sub>X tworzy z pudełek (box), które odpowiednio skleja. Elementarnymi pudełkami są litery, z których sklejane są słowa. Słowa są następnie łączone w wiersze, a wiersze – w akapity. Do łączenia używany jest

specjalny klej, który dzięki elastyczności pozwala wyrazy ścisnąć lub rozciągnąć tak, by dokładnie wypełniały wiersze na stronie.

Trzeba przyznać, że takie ujęcie jest mocno uproszczoną wersją tego, co się naprawdę dzieje, chociaż zasadniczo biorąc, działanie T<sub>E</sub>Xa można jednak wyjaśnić właśnie w terminach pudełek oraz kleju (odstępu wstawianego między pudełkami). Pudełkami są nie tylko litery. Do pudełka można włożyć praktycznie wszystko, także inne pudełka. Każde pudełko L<sup>A</sup>T<sub>E</sub>X traktuje jak pojedynczą literę.

Chociaż nie mówiliśmy o tym wprost, pudełka pojawiały się już w poprzednich rozdziałach. Na przykład polecenie `\includegraphics` albo otoczenie `tabular` tworzą pudełka. Dzięki temu dwa rysunki albo tabele można łatwo zestawić obok siebie. Trzeba jedynie zadbać o to, by łączna szerokość połączonych obiektów nie przekraczała szerokości tekstu.

To samo odnosi się do akapitów, które – jeśli tego potrzebujemy – możemy składać w pudełka o zadanej szerokości przy użyciu polecenia

```
\parbox[pozycja]{szerokość}{tekst}
```

lub otoczenia

```
\begin{minipage}[pozycja]{szerokość} tekst \end{minipage}
```

Argument *pozycja* przyjmuje jedną z liter *c*, *t* lub *b* aby sterować wyrównaniem pionowym tekstu wokół. Argument *szerokość* to wymiar określający szerokość pudełka. Główna różnica między `\parbox` a `minipage` polega na tym, że wewnątrz `\parbox` nie wszystkie polecenia i otoczenia są dozwolone, w przeciwieństwie do otoczenia `minipage`.

Polecenie `\parbox` składa tekst w pudełku, w razie potrzeby dzieląc tekst na linijki. Istnieją również pudełka, które operują tylko na materiale wyrównanym poziomo. Znamy już jedno z nich: nazywa się `\mbox`. Pakuje ono wiele pudełek w jedno i może być użyte by zapobiec łamaniu słów przez L<sup>A</sup>T<sub>E</sub>Xa. Ponieważ pudełka można wkładać do pudełek, to pudełka te dają nam pełną dowolność jeśli chodzi o układ poziomy.

```
\makebox[szerokość][pozycja]{tekst}
```

W opcjonalnym argumencie *szerokość* możemy zadać szerokość pudełka widoczną z zewnątrz<sup>5</sup>. W obrębie argumentu *szerokość* możemy się też posłu-

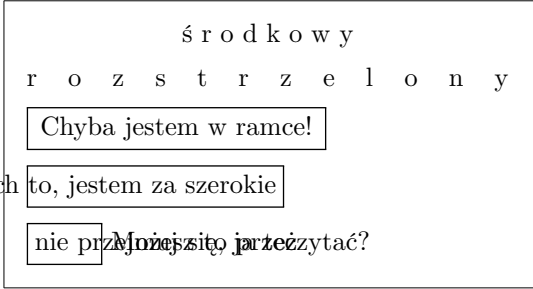
<sup>5</sup>Może się ona różnić od faktycznej szerokości tekstu w pudełku; może wynosić zero,

giwać wielkościami `\width` (szerokość), `\height` (wysokość), `\depth` (głębokość) oraz `\totalheight` (suma wysokości i głębokości). Ponadto argument *pozycja* określa sposób umieszczenia tekstu. Litera *c* oznacza wyśrodkowanie, *l* – dosunięcie do lewej, *r* – dosunięcie do prawej, a *s* – rozstrzelenie zawartości.

Polecenie `\framebox` działa dokładnie tak samo jak `\makebox`, ale rysuje ramkę wokół tekstu.

Poniższy przykład obrazuje w jaki sposób można wykorzystać polecenia `\makebox` oraz `\framebox`.

```
\makebox[\textwidth]{%
  ś r o d k o w y}\par
\makebox[\textwidth][s]{%
  r o z s t r z e l o n y}\par
\framebox[1.1\width]{Chyba
  jestem w ramce!}\par
\framebox[0.8\width][r]{Niech
  to, jestem za szerokie}\par
\framebox[1cm][l]{nie przejmuj
  się, ja też}
Możesz to przeczytać?
```



Ponieważ kontrolujemy już poziom, oczywistym następnym krokiem jest przejście kontroli nad pionem<sup>6</sup>. Żaden problem dla L<sup>A</sup>T<sub>E</sub>Xa. Polecenie

```
\raisebox{przesunięcie}[zasięg-powyżej-tekstu][zasięg-poniżej-tekstu]{tekst}
```

pozwala definiować wartości pionowe pudełka. Możesz użyć `\width`, `\height`, `\depth` i `\totalheight` w pierwszych trzech argumentach by opierać się na wewnętrznym *tekście* pudełka.

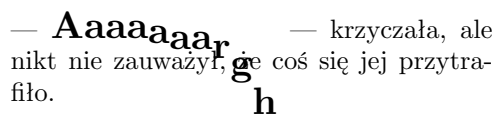
a nawet być wielkością ujemną! Ustawienie jej na wartość zero oznacza, że nie będzie ona wpływać na pozostałe pudełka w dokumencie.

<sup>6</sup>Pełną kontrolę można osiągnąć jedynie poprzez opanowanie zarówno poziomu jak i pionu...

```

--- \raisebox{0pt}[0pt][0pt]{%
\Large\textbf{%
Aaaa\raisebox{-0.3ex}{a}%
\raisebox{-0.7ex}{aa}%
\raisebox{-1.2ex}{r}%
\raisebox{-2.2ex}{g}%
\raisebox{-4.5ex}{h}}
--- krzyczała, ale nikt nie
zauważył, że coś się
jej przytrafiło.

```



## 6.7 Kreski

Kilka stron temu napotkaliśmy polecenie

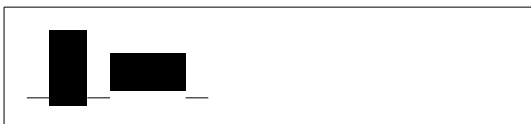
```
\rule[przesunięcie]{szerokość}{wysokość}
```

W normalnym użyciu tworzy ono zwyczajne czarne pudełko.

```

\rule{3mm}{.1pt}%
\rule[-1mm]{5mm}{1cm}%
\rule{3mm}{.1pt}%
\rule[1mm]{1cm}{5mm}%
\rule{3mm}{.1pt}

```



Polecenie `\rule` służy zwykle do rysowania kresek pionowych i poziomych. Na przykład gruba czarna krecha na stronie tytułowej niniejszego *Wprowadzenia* to wynik zadziałania tej właśnie instrukcji.

Koniec.





## Dodatek A

# Instalacja L<sup>A</sup>T<sub>E</sub>Xa

Knuth opublikował kod źródłowy T<sub>E</sub>Xa w czasach gdy jeszcze Wolne Oprogramowanie nie było przyjętym terminem. Licencja na której opublikowany jest T<sub>E</sub>X pozwala robić cokolwiek z kodem źródłowym, jednak nazwanie swojego rezultatu T<sub>E</sub>X wymaga by program przechodził zbiór testów dostarczonych przez Knutha. Doprowadziło to do sytuacji w której wolne implementacje T<sub>E</sub>Xa są dostępne na niemal wszystkich istniejących systemach operacyjnych. Ten rozdział dostarczy kilku wskazówek co należy zainstalować na systemach Linux, macOS oraz Windows by móc rozpocząć pracę z T<sub>E</sub>Xem.

### A.1 Co zainstalować

Aby używać L<sup>A</sup>T<sub>E</sub>Xa na komputerze potrzebne jest kilka programów.

1. Program T<sub>E</sub>X/L<sup>A</sup>T<sub>E</sub>X do przetwarzania plików źródłowych L<sup>A</sup>T<sub>E</sub>Xa oraz składania ich do plików PDF.
2. Edytor tekstu do edytowania twoich plików źródłowych L<sup>A</sup>T<sub>E</sub>Xa. Niektóre z nich pozwalają nawet uruchomić L<sup>A</sup>T<sub>E</sub>Xa bezpośrednio z programu.
3. Przeglądarka plików PDF do przeglądania oraz drukowania dokumentów.
4. Program do obsługi plików POSTSCRIPT oraz obrazów do włączenia w dokumencie.

Każda platforma zawiera kilka programów spełniających powyższe kryteria. Tutaj opowiemy o kilku, które znamy i z którymi mamy doświadczenie.

## A.2 Edytor międzyplatformowy

O ile T<sub>E</sub>X jest dostępny na wielu platformach, to edytory L<sup>A</sup>T<sub>E</sub>Xa przez długi czas były związane z konkretnymi platformami.

W ciągu kilku ostatnich lat polubiłem bardzo program Texmaker. Oprócz bycia bardzo użytecznym edytorem z wbudowanym podglądem PDF i podświetleniem składni ma on tę zaletę, że działa równie dobrze na systemach Windows Mac oraz Unix/Linux. Więcej informacji w [19]. Istnieje również fork o nazwie TeXstudio [20]. Również wygląda na aktualny i jest dostępny na wszystkich trzech platformach.

W sekcjach poniżej znajdziesz również sugestie edytorów przeznaczonych pod konkretny system,

## A.3 T<sub>E</sub>X na macOS

### A.3.1 Dystrybucje T<sub>E</sub>Xa

Wystarczy pobrać MacTeX [21]. Jest to skompilowana dystrybucja L<sup>A</sup>T<sub>E</sub>Xa dla systemu macOS. MacTeX dostarcza pełną instalację L<sup>A</sup>T<sub>E</sub>Xa oraz kilka dodatkowych narzędzi.

### A.3.2 Edytor T<sub>E</sub>X dla macOS

Jeśli nie podoba ci się nasza sugestia wieloplatformowego Texmaker (sekcja A.2).

Najbardziej popularnym otwartoźródłowym edytorem L<sup>A</sup>T<sub>E</sub>Xa na maca zdaje się być T<sub>E</sub>XShop [22]. Jest on również zawarty w dystrybucji MacTeX.

Nowsze wersje dystrybucji T<sub>E</sub>XLive zawierają edytor T<sub>E</sub>Xworks [23], który również jest wieloplatformowym edytorem podobnym z wyglądu do T<sub>E</sub>XShop. Ponieważ T<sub>E</sub>Xworks korzysta z Qt jest on dostępny na wszystkich platformach wspierając Qt (macOS, Windows, Linux).

### A.3.3 Poczęstuj się PDFView

Aby podglądać pliki PDF wygenerowane przez L<sup>A</sup>T<sub>E</sub>Xa używaj PDFView [24]; integruje się on dobrze z edytorami tekstu L<sup>A</sup>T<sub>E</sub>Xa. Po instalacji otwórz panel ustawień aplikacji i upewnij się, że opcja *automatycznie przeladuj dokumenty* jest włączona oraz, że wsparcie dla PDFSync jest poprawnie ustawione.

## A.4 T<sub>E</sub>X na Windows

### A.4.1 Dystrybucje T<sub>E</sub>Xa

Na początku możesz skorzystać z doskonałej dystrybucji MiK<sub>T</sub>E<sub>X</sub>T<sub>E</sub>X [25]. Zawiera ona wszystkie podstawowe programy i pliki wymagane do kompilowania L<sup>A</sup>T<sub>E</sub>Xowych dokumentów. Dla mnie najlepszą jej funkcją jest fakt, że MiK<sub>T</sub>E<sub>X</sub> automatycznie pobierze brakujące pakiety i zainstaluje je magicznie podczas kompilacji dokumentu. Alternatywnie możesz też skorzystać z dystrybucji TeXlive [26], która jest dostępna dla systemów Windows, Unix oraz Mac OS aby rozpocząć pracę.

### A.4.2 Edytor L<sup>A</sup>T<sub>E</sub>Xa

Jeśli nie podoba ci się nasza sugestia wieloplatformowego Texmaker (sekcja A.2).

TeXnicCenter [27] korzysta z wielu konceptów znanych ze świata programowania by dostarczyć przyjemne i efektywne środowisko pracy w L<sup>A</sup>T<sub>E</sub>Xu na Windowsie. TeXnicCenter współpracuje dobrze z dystrybucją MiK<sub>T</sub>E<sub>X</sub>.

Nowsze wersje dystrybucji T<sub>E</sub>XLive zawierają edytor T<sub>E</sub>Xworks [23]. Wspiera on Unicode i wymaga co najmniej Windows XP.

### A.4.3 Podgląd dokumentu

Jeśli chodzi o PDF to warto przyjrzeć się Sumatra PDF [28]. Wspominam o tym programie ponieważ pozwala on skoczyć z dowolnego miejsca w dokumencie do odpowiadającego mu miejsca w pliku źródłowym dokumentu.

### A.4.4 Praca z grafiką

Aby uzyskać wysoką jakość grafiki w L<sup>A</sup>T<sub>E</sub>Xu oznacza, że musisz pracować z formatami Encapsulated POSTSCRIPT (eps) lub PDF przy twoich grafikach. Program który z tym pomaga nazywa się GhostScript [29]. Posiada on własny interfejs GhostView.

Jeśli korzystasz z grafiki rastrowej (fotografie i skany) możesz również przyjrzeć się na otwartoźródłową alternatywę dla Photoshopa – Gimp [30].

## A.5 T<sub>E</sub>X na Linuxie

Jeśli pracujesz z Linuxem, to istnieje spora szansa, że L<sup>A</sup>T<sub>E</sub>X jest już zainstalowany na twoim systemie, lub przynajmniej dostępny w źródle instalacji,

która została użyta. Skorzystaj z menadżera aplikacji aby zainstalować następujące paczki:

- texlive – podstawowy program T<sub>E</sub>X/L<sup>A</sup>T<sub>E</sub>X.
- emacs (z AU<sub>C</sub>TeX) – edytor, który integruje się mocno z L<sup>A</sup>T<sub>E</sub>Xem przy użyciu plugin AU<sub>C</sub>TeX.
- ghostscript – program do podglądu POSTSCRIPT.
- xpdf oraz acrobat – przeglądarka plików PDF.
- imagemagick – darmowy program do konwertowania grafiki rastrowej.
- gimp – wolny bliźniak Photoshopa.
- inkscape – wolny ilustrator/bliźniak corel draw.

Jeśli oczekujesz bardziej graficznego środowiska do edycyjnego przyjrzyj się programowi Texmaker. Zobacz w sekcji [A.2](#).

Większość dystrybucji Linuxa upiera się na podzieleniu swoich środowisk T<sub>E</sub>Xa na sporą liczbę opcjonalnych paczek, więc jeśli czegoś brakuje po instalacji sprawdź to ponownie.

**Dodatek B**

# **GNU GENERAL PUBLIC LICENSE**

**Version 2, June 1991**

Copyright © 1989, 1991 Free Software Foundation, Inc.

51 Franklin Street, Fifth Floor, Boston, MA 02110-1301, USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

## **Preamble**

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software—to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

## TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program

(independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:
  - (a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.
  - (b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.
  - (c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when

you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:
  - (a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,
  - (b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,
  - (c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special



exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.
5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.
6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.
7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and

any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.
9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and “any later version”, you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version

number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

## NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.
12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

## END OF TERMS AND CONDITIONS

## Appendix: How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the “copyright” line and a pointer to where the full notice is found.

one line to give the program’s name and a brief idea of what it does.

Copyright (C) yyyy name of author

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301, USA.

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) yyyy name of author  
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type ‘show w’.

This is free software, and you are welcome to redistribute it under certain conditions; type ‘show c’ for details.

The hypothetical commands `show w` and `show c` should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w` and `show c`; they could even be mouse-clicks or menu items—whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a “copyright disclaimer” for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program  
‘Gnomovision’ (which makes passes at compilers) written by James Hacker.

signature of Ty Coon, 1 April 1989  
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.



# Bibliografia

- [1] Leslie Lamport. *L<sup>A</sup>T<sub>E</sub>X: A Document Preparation System*. 2 wyd. Reading, Massachusetts: Addison-Wesley, 1994. ISBN: 0-201-52983-1.
- [2] Donald E. Knuth. *The T<sub>E</sub>Xbook*. 2 wyd. T. A. Computers and Wpisz-setting. Reading, Massachusetts: Addison-Wesley, 1984. ISBN: 0-201-13448-9.
- [3] Frank Mittelbach i in. *The L<sup>A</sup>T<sub>E</sub>X Companion*. 2 wyd. Reading, Massachusetts: Addison-Wesley, 2004. ISBN: 0-201-36299-6.
- [4] Każda instalacja L<sup>A</sup>T<sub>E</sub>Xa powinna zawierać *L<sup>A</sup>T<sub>E</sub>X Local Guide*, w którym są opisane rzeczy specyficzne dla danej lokalnej instalacji. Dokument ten powinien być zawarty w pliku `local.tex`. W wielu wypadkach administratorzy nie udostępniają jednak użytkownikom takiego dokumentu. Pozostaje wtedy zwrócić się o pomoc do lokalnego L<sup>A</sup>T<sub>E</sub>X-owego guru.
- [5] François Charette. *Polyglossia: A Babel Replacement for X<sub>Y</sub>L<sup>A</sup>T<sub>E</sub>X*. Dostępne wraz z dystrybucją T<sub>E</sub>XLive w pliku `polyglossia.pdf`. (Wpisz `texdoc polyglossia` w linii poleceń.)
- [6] Will Robertson i Khaled Hosny. *The fontspec package*. Dostępne wraz z dystrybucją T<sub>E</sub>XLive w pliku `fontspec.pdf`. (Wpisz `texdoc fontspec` w linii poleceń.)
- [7] Apostolos Syropoulos. *The xgreek package*. Dostępne wraz z dystrybucją T<sub>E</sub>XLive w pliku `xgreek.pdf`. (Wpisz `texdoc xgreek` w linii poleceń.)
- [8] Vafa Khalighi. *The bidi package*. Dostępne wraz z dystrybucją T<sub>E</sub>XLive w pliku `bidi.pdf`. (Wpisz `texdoc bidi` w linii poleceń.)
- [9] Vafa Khalighi. *The XePersian package*. Dostępne wraz z dystrybucją T<sub>E</sub>XLive w pliku `xepersian-doc.pdf`. (Wpisz `texdoc xepersian` w linii poleceń.)

- [10] François Charette. *An Arab $\TeX$ -like interface for typesetting languages in Arabic script with  $X_{\text{F}}\text{L}\text{A}\text{T}\text{E}\text{X}$* . Dostępne wraz z dystrybucją  $\text{T}\text{E}\text{X}$ Live w pliku arabxetex.pdf. (Wpisz texdoc arabxetex w linii poleceń.)
- [11] Wenchang Sun. *The xeCJK package*. Dostępne wraz z dystrybucją  $\text{T}\text{E}\text{X}$ Live w pliku xeCJK.pdf. (Wpisz texdoc xecjk w linii poleceń.)
- [12] D. P. Carlisle. *Packages in the ‘graphics’ bundle*. Dokument dostępny w zestawie pakietów „graphics” w pliku grfguide.tex.
- [13] Rainer Schöpf, Raichle Bernd i Chris Rowley. *A New Implementation of  $\text{L}\text{A}\text{T}\text{E}\text{X}$ ’s verbatim Environments*. Dokument dostępny w zestawie pakietów „tools” w pliku verbatim.dtx.
- [14] Michel Goossens, Sebastian Rahtz i Frank Mittelbach. *The  $\text{L}\text{A}\text{T}\text{E}\text{X}$  Graphics Companion*. Reading, Massachusetts: Addison-Wesley, 1997. ISBN: 0-201-85469-4.
- [15] John D. Hobby. *A User’s Manual for METAPOST*. Dostępny w <http://cm.bell-labs.com/who/hobby/>.
- [16] Urs Oswald. *Graphics in  $\text{L}\text{A}\text{T}\text{E}\text{X} 2_{\epsilon}$* . Oba dostępne w <http://www.ursoswald.ch>. zawiera plik źródłowy Javy do generowania dowolnych okręgów i elips przy użyciu otoczenia picture, i *METAPOST - A Tutorial*.
- [17] Alan Hoenig.  *$\text{T}\text{E}\text{X}$  Unbound*. Oxford University Press, 1998. ISBN: 0-19-509685-1; 0-19-509686-X (pbk.)
- [18] Till Tantau. *TikZ&PGF Manual*. Dostępne w <CTAN://graphics/pgf/base/doc/generic/pgf/pgfmanual.pdf>.
- [19] Pascal Brachet. *Texmaker*. Wer. 5.1.3. 2022. URL: <https://www.xm1math.net/texmaker/>.
- [20] Benito van der Zander. *TeXstudio*. Wer. 4.2.3. 2022. URL: <https://www.texstudio.org/>.
- [21] MacTeX TeXnical working group. *MacTeX*. Wer. MacTeX-2022. 2022. URL: <https://tug.org/mactex/>.
- [22] Richard Koch et al. *TeXShop*. Wer. 4.70. 2022. URL: <https://pages.uoregon.edu/koch/texshop/>.
- [23] Jonathan Kew, Stefan Löffler i Charlie Sharpsteen. *TeXworks*. Wer. 0.6.7. 2022. URL: <https://www.tug.org/texworks/>.
- [24] Andrea Bergia. *PDFView*. Wer. 0.14.3. 2007. URL: <http://pdfview.sourceforge.net/>.



- [25] Christian Schenk. *MiKTeX*. Wer. 22.3. 2022. URL: <https://miktex.org/>.
- [26] Karl Berry. *TeX Live*. Wer. 2022. 2022. URL: <https://www.tug.org/texlive/>.
- [27] The TeXnicCenter Team. *TeXnicCenter*. Wer. 2.02. 2013. URL: <https://www.texniccenter.org/>.
- [28] Krzysztof Kowalczyk. *Sumatra PDF*. Wer. 3.3.3. 2021. URL: <https://www.sumatrapdfreader.org/>.
- [29] L. Peter Deutsch. *Ghostscript*. Wer. 9.56.1. 2022. URL: <https://www.ghostscript.com/>.
- [30] Spencer Kimball i Peter Mattis. *GIMP*. Wer. 2.10.30. 2021. URL: <https://www.gimp.org/>.
- [31] L<sup>A</sup>T<sub>E</sub>X3 Project Team. *L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub> for authors*. Dokument dostępny w pliku `usrguide.tex` w dystrybucji L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub>.
- [32] L<sup>A</sup>T<sub>E</sub>X3 Project Team. *L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub> for Class and Package writers*. Dokument dostępny w pliku `clsguide.tex` w dystrybucji L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub>.
- [33] L<sup>A</sup>T<sub>E</sub>X3 Project Team. *L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub> Font selection*. Dokument dostępny w pliku `fntguide.tex` w dystrybucji L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub>.
- [34] Vladimir Volovich, Werner Lemberg i L<sup>A</sup>T<sub>E</sub>X3 Project Team. *Cyrillic languages support in L<sup>A</sup>T<sub>E</sub>X*. Rozpowszechniany w dystrybucji L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub> w pliku `cyrguide.tex`.
- [35] Graham Williams. *The TeX Catalogue*. Dostępny online w [CTAN://help/Catalogue/catalogue.html](https://ctan.org/help/Catalogue/catalogue.html). Jest to kompletne zestawienie wielu pakietów powiązanych T<sub>E</sub>Xem i L<sup>A</sup>T<sub>E</sub>Xem.
- [36] Kristoffer H. Rose. *X<sub>Y</sub>-pic User's Guide*. Dostępne w CTAN z pakietem X<sub>Y</sub>-pic.

# Indeks

## Symbols

\!, 70  
\,, 54, 69  
-, 25  
—, 25  
—, 25  
—, 25  
\-, 23  
., (spacja po), 32  
..., 27  
\:, 69  
\;, 69  
\@, 32  
\[, 53  
\$, 52  
\\, 21, 38, 39, 42, 135  
\\\*, 21  
\], 53  
~, 32

## A

A4, 12  
A5, 12  
abstract, 40  
\addtolength, 137  
adres URL, 25  
æ, 28  
akcenty  
    matematyczne, 57  
akut, 28  
align, 62  
\Alpha, 77

\alt, 104  
amsbsy, 72  
amsmath, 51, 59, 69, 70, 72  
amssymb, 55, 71, 77  
amsthm, 73, 74  
\and, 35  
\appendix, 34, 35  
arabski, 31  
arabxetex, 31  
\arccos, 57  
\arcsin, 57  
\arctan, 57  
\arg, 57  
argument, 6  
    opcjonalny, 6  
array, 68, 69  
\arraystretch, 43  
\author, 35, 97, 103

## B

B5, 12  
babel, 28  
\backmatter, 35  
\bar, 57  
\baselineskip, 132  
beamer, 101, 102, 104, 106  
\begin, 37, 106, 118  
\Beta, 77  
bezszerfowe, 129  
\bibitem, 87  
bibliografia, 87

- bidi, 31
- \Big, 60
- \big, 60
- \Bigg, 60
- \bigg, 60
- \bigskip, 135
- \binom, 59
- block, 103
- bm, 72
- Bmatrix, 69
- bmatrix, 69
- \bmod, 58
- \boldmath, 72
- \boldsymbol, 72
- booktabs, 44
  
- C**
- calc, 137
- \caption, 48
- cases, 68
- \cdot, 56
- \cdots, 56
- center, 38
- \chapter, 33, 90
- \chaptermark, 90, 92
- chiński, 32
- \ci, 124
- \circle, 110
- \circle\*, 110
- \cite, 87
- \cleardoublepage, 49
- \clearpage, 49
- \cline, 42
- color, 101
- comment, 7
- \cos, 57
- \cosh, 57
- \cot, 57
- \coth, 57
- \csc, 57
- csquotes, 29
- cudzysłów, 24
  
- D**
- \date, 35
- dcolumn, 43
- \ddots, 56
- \DeclareMathOperator, 58
- \deg, 57
- \depth, 140
- description, 38
- \det, 57
- \dfrac, 58
- \dim, 57
- displaymath, 52
- \displaystyle, 71
- doc, 13
- \documentclass, 10, 15, 22
- \dum, 123, 124
- dwie kolumny, 12
- dwustronne, 12
- dywiz, 25
- długie równania, 61
  
- E**
- eepic, 110
- \emph, 37, 131
- empty, 14
- Encapsulated POSTSCRIPT, 145
- \end, 37, 106
- enumerate, 38
- eqnarray, 63
- \eqref, 52
- equation, 52, 53, 61, 63
- equation\*, 53, 61
- eurosym, 26
- executive (format arkusza), 12
- \exp, 57
- exscale, 13

- F**
- `\false`, 127
  - `fancyhdr`, 90, 91
  - `\fbox`, 24
  - `figure`, 46
  - `flushleft`, 38
  - `flushright`, 38
  - `\foldera`, 115
  - `\folderb`, 115
  - `font`, 128
    - `\footnotesize`, 129
    - `\Huge`, 129
    - `\huge`, 129
    - `\LARGE`, 129
    - `\Large`, 129
    - `\large`, 129
    - `\mathbf`, 130
    - `\mathcal`, 130
    - `\mathit`, 130
    - `\mathnormal`, 130
    - `\mathrm`, 130
    - `\mathsf`, 130
    - `\mathtt`, 130
    - `\normalsize`, 129
    - `\scriptsize`, 129
    - `\small`, 129
    - `\textbf`, 129
    - `\textit`, 129
    - `\textmd`, 129
    - `\textnormal`, 129
    - `\textrm`, 129
    - `\textsc`, 129
    - `\textsf`, 129
    - `\textsl`, 129
    - `\texttt`, 129
    - `\textup`, 129
    - `\tiny`, 129
  - `fontenc`, 13
  - `fontspec`, 28, 100
  - `\footnote`, 36
  - `\footnotesize`, 129
  - `\footskip`, 136
    - `format arkusza`, 12
  - `\frac`, 58
    - `frame`, 103
  - `\framebox`, 140
  - `\frenchspacing`, 33
  - `\frontmatter`, 35
    - `funkcje częściowe`, 68
  - `\fussy`, 22
- G**
- `\gcd`, 57
  - `geometry`, 92
  - `GhostScript`, 145
  - `GhostView`, 145
  - `Gimp`, 145
  - `grafika`, 10
  - `graphicx`, 44, 101
  - `grawis`, 28
  - `Greka`, 31
  - `grupa`, 128
  - `grupowanie`, 128
- H**
- `\hat`, 57
  - `\headheight`, 136
    - `headings`, 14
  - `\headsep`, 136
    - `hebrajski`, 32
  - `\height`, 140
    - `hipertekst`, 94
    - `hiperłącze`, 94
  - `\hline`, 42
  - `\hom`, 57
  - `\href`, 97
  - `\hspace`, 126, 133
    - `\hspace*`, 133
  - `HTML`, 94
  - `\Huge`, 129

- `\huge`, 129  
`hyperref`, 31, 94, 95, 98, 101  
`hyphenat`, 92  
`\hyphenation`, 23
- I**
- `ı` („i” bez kropki), 28  
`\idotsint`, 70  
`IEEEeqnarray`, 61, 62, 64  
`\IEEEeqnarraymulticol`, 66  
`\IEEEmulticol`, 67  
`\IEEEnonumber`, 67  
`IEEEtrantools`, 64  
`\IEEEyesnumber`, 67  
`\IEEEyessubnumber`, 68  
`ifthen`, 13  
`\ignorespaces`, 126  
`\ignorespacesafterend`, 126  
`\iiiint`, 70  
`\iiint`, 70  
`\iint`, 70  
`\include`, 16  
`\includegraphics`, 45, 139  
`\includeonly`, 16  
indeks, 88  
    dolny, 55  
    górnny, 55  
`\indent`, 133  
    `indentfirst`, 133  
`\index`, 89, 90  
`\inf`, 57  
`\input`, 16  
    `inputenc`, 13  
`\institute`, 103  
`\int`, 59  
    interlinia, 132  
`\item`, 38  
    `itemize`, 38, 104
- J**
- `J` („j” bez kropki), 28  
japoński, 32  
Jawi, 32  
jedna kolumna, 12  
jednostki, 134, 135  
jednostronne, 12  
język, 28  
języki, 28
- K**
- Kapitaliki, 129  
kashida, 31  
Kashmiri, 31  
`\ker`, 57  
klamra  
    pozioma, 57  
klasa  
    article, 11  
    book, 11  
    minimal, 11  
    proc, 11  
    slides, 11  
klej, 139  
Knuth Donald E., 1  
kolorowy tekst, 10  
komentarz, 7  
koreański, 32  
kreska  
    pozioma, 56  
kropka, 27, 56  
kropki, 56  
    trzy, 56  
kropki przekątne, 56  
kurdyjski, 31  
kursywa, 129  
`\kwle`, 124
- L**
- `\label`, 35, 36, 48

- Lamport Leslie, 2
- \LARGE, 129
- \Large, 129
- \large, 129
- \LaTeX, 24
- $\LaTeX$ 3, 4
- \LaTeXe, 24
- latexsym, 13
- layout, 135
- layouts, 136
- \ldots, 27, 56
- \left, 60
- \lefteqn, 64, 66
- \leftmark, 90, 92
- letter (format arkusza), 12
- \lg, 57
- ligatura, 27
- \lim, 57
- \liminf, 57
- \limsup, 57
- \line, 108, 115
- \linebreak, 22
- \linespread, 132
- \linethickness, 112, 113, 116
- \listoffigures, 48
- \listoftables, 48
- \ln, 57
- \log, 57
- longtable, 43
- lscommand, 124
  
- M**
- Macierz, 69
- MacTeX, 144
- \mainmatter, 35, 97
- \makebox, 139, 140
- makeidx, 88
- makeidx, 13, 89
- makeindex, 88
- \makeindex, 89
- \maketitle, 35
- Malay, 32
- \marginparpush, 136
- \marginparsep, 136
- \marginparwidth, 136
- matematyka, 51
- funkcje, 57
- minus, 25
- \mathbb, 55
- \mathbf, 130
- \mathcal, 130
- \mathit, 130
- \mathnormal, 130
- \mathrm, 130
- mathrsfs, 82
- \mathsf, 130
- \mathtt, 130
- matrix, 69
- \max, 57
- \mbox, 23, 24, 27, 139
- \medskip, 135
- mhchem, 71
- microtype, 101
- MiKTeX, 145
- \min, 57
- minipage, 139
- minipage, 139
- modulo, 58
- \multicolumn, 43
- \multicolumns, 66
- \multirow, 107, 112
- multline, 61–63
- multline\*, 61
  
- N**
- nagłówek, 14
- nawias, 60
- nawiasy klamrowe, 6
- nawiasy kwadratowe, 6
- \negmedspace, 67

- `\newcommand`, 70, 124, 125
- `\newenvironment`, 125, 126
- `\newline`, 21
- `\newpage`, 22
- `\newsavebox`, 115
- `\newtheorem`, 72
- `\noindent`, 133
- `\nolinebreak`, 22
- `\nonumber`, 67
- `\nopagebreak`, 22
- `\normalsize`, 129
- `\not`, 78
- `ntheorem`, 74
  
- O**
- `\oddsidemargin`, 136
- odstęp, 5
  - na początku wiersza, 5
  - pionowy, 135
  - po komendzie, 6
  - poziomy, 133
  - w trybie matematycznym, 69
- odstęp
  - tryb matematyczny, 54
- odsyłacze, 35
- œ, 28
- ogranicznik, 60
- `\only`, 104
- opcje, 10
- operator
  - iloczynu, 59
  - sumowania, 59
- otoczenia
  - `Bmatrix`, 69
  - `IEEEeqnarray`, 61, 62, 64
  - `Vmatrix`, 69
  - abstract, 40
  - align, 62
  - array, 68, 69
  - block, 103
  - `bmatrix`, 69
  - cases, 68
  - center, 38
  - comment, 7
  - description, 38
  - `displaymath`, 52
  - enumerate, 38
  - `eqnarray`, 63
  - `equation*`, 53, 61
  - equation, 52, 53, 61, 63
  - figure, 46
  - `flushleft`, 38
  - `flushright`, 38
  - frame, 103
  - itemize, 38, 104
  - `lscommand`, 124
  - matrix, 69
  - minipage, 139
  - `multline*`, 61
  - `multline`, 61–63
  - picture, 105–107, 110, 111
  - `pmatrix`, 69
  - proof, 74
  - quotation, 39
  - quote, 39
  - table, 46, 47
  - tabular, 41, 43, 44, 139
  - `thebibliography`, 87
  - `tikzpicture`, 119
  - verbatim, 40, 92
  - verse, 39
  - `vmatrix`, 69
- Ottoman, 31
- `\oval`, 113, 115
- `\overbrace`, 57
- overfull hbox, 22
- `\overleftarrow`, 57
- `\overline`, 56
- `\overrightarrow`, 57

**P**

- `\pagebreak`, 22, 135
- `\pageref`, 35, 94
- `\pagestyle`, 14
- pagina dolna, 14
- pagina górna, 14
- pakiet, 8, 10
- pakiety
  - ambsy, 72
  - amfonts, 71, 82
  - amsmath, 51, 59, 69, 70, 72
  - amssymb, 55, 71, 77
  - amsthm, 73, 74
  - arabxetex, 31
  - babel, 28
  - beamer, 101, 102, 104, 106
  - bidi, 31
  - bm, 72
  - booktabs, 44
  - calc, 137
  - color, 101
  - csquotes, 29
  - dcolum, 43
  - doc, 13
  - eeepic, 110
  - eurosym, 26
  - exscale, 13
  - fancyhdr, 90, 91
  - fontenc, 13
  - fontspec, 28, 100
  - geometry, 92
  - graphicx, 44, 101
  - hyperref, 31, 94, 95, 98, 101
  - hyphenat, 92
  - IEEEtrantools, 64
  - ifthen, 13
  - indentfirst, 133
  - inputenc, 13
  - latexsym, 13
  - layout, 135
  - layouts, 136
  - longtable, 43
  - makeidx, 13, 89
  - mathrsfs, 82
  - mhchem, 71
  - microtype, 101
  - minipage, 139
  - ntheorem, 74
  - pgf, 106, 118, 119, 121
  - pgfplot, 121
  - polyglossia, 23, 28, 30–32
  - ppower4, 101
  - prosper, 101
  - pstricks, 110
  - showidx, 90
  - syntonly, 13, 16
  - textcomp, 26
  - tikz, 106, 119
  - verbatim, 7, 92
  - xeCJK, 32
  - xepersian, 31
  - xgreek, 31
- `\paperheight`, 136
- `\paperwidth`, 136
- `\par`, 130, 132
- `\paragraph`, 33
- `\parbox`, 139
- `\parindent`, 132
- `\parskip`, 132
- `\part`, 33, 34
- `\partial`, 59
- Pashto, 31
- `\pause`, 104
- pauza, 25
- PDF, 94, 98
- pdfL<sup>A</sup>T<sub>E</sub>X, 101
- PDFView, 144
- perski, 31
- pgf, 106, 118, 119, 121
- pgfplot, 121



- `\phantom`, 70
- picture, 105–107, 110, 111
- pierwiastek kwadratowy, 56
- pionowe
  - kropki, 56
- plain, 14
- plik źródłowy, 8
- pmatrix, 69
- `\pmod`, 58
- pochodna częściowa, 59
- pochylone, 129
- podwójna interlinia, 132
- pogrubione, 129
- pogrubione symbole, 72
- polecenia, 6
  - `\!`, 70
  - `\,`, 54, 69
  - `\-`, 23
  - `\:`, 69
  - `\;`, 69
  - `\@`, 32
  - `\[`, 53
  - `\`, 21, 38, 39, 42, 135
  - `\*`, 21
  - `\]`, 53
  - `\addtolength`, 137
  - `\Alpha`, 77
  - `\alt`, 104
  - `\and`, 35
  - `\appendix`, 34, 35
  - `\arccos`, 57
  - `\arcsin`, 57
  - `\arctan`, 57
  - `\arg`, 57
  - `\arraystretch`, 43
  - `\author`, 35, 97, 103
  - `\backmatter`, 35
  - `\bar`, 57
  - `\baselineskip`, 132
  - `\begin`, 37, 106, 118
  - `\Beta`, 77
  - `\bibitem`, 87
  - `\Big`, 60
  - `\big`, 60
  - `\Bigg`, 60
  - `\bigg`, 60
  - `\bigskip`, 135
  - `\binom`, 59
  - `\bmod`, 58
  - `\boldmath`, 72
  - `\boldsymbol`, 72
  - `\caption`, 48
  - `\cdot`, 56
  - `\cdots`, 56
  - `\chapter`, 33, 90
  - `\chaptermark`, 90, 92
  - `\ci`, 124
  - `\circle`, 110
  - `\circle*`, 110
  - `\cite`, 87
  - `\cleardoublepage`, 49
  - `\clearpage`, 49
  - `\cline`, 42
  - `\cos`, 57
  - `\cosh`, 57
  - `\cot`, 57
  - `\coth`, 57
  - `\csc`, 57
  - `\date`, 35
  - `\ddots`, 56
  - `\DeclareMathOperator`, 58
  - `\deg`, 57
  - `\depth`, 140
  - `\det`, 57
  - `\dfrac`, 58
  - `\dim`, 57
  - `\displaystyle`, 71
  - `\documentclass`, 10, 15, 22
  - `\dum`, 123, 124
  - `\emph`, 37, 131

`\end`, 37, 106  
`\eqref`, 52  
`\exp`, 57  
`\false`, 127  
`\fbox`, 24  
`\foldera`, 115  
`\folderb`, 115  
`\footnote`, 36  
`\footskip`, 136  
`\frac`, 58  
`\framebox`, 140  
`\frenchspacing`, 33  
`\frontmatter`, 35  
`\fussy`, 22  
`\gcd`, 57  
`\hat`, 57  
`\headheight`, 136  
`\headsep`, 136  
`\height`, 140  
`\hline`, 42  
`\hom`, 57  
`\href`, 97  
`\hspace`, 126, 133  
`\hspace*`, 133  
`\hyphenation`, 23  
`\idotsint`, 70  
`\IEEEeqnarraymulticol`, 66  
`\IEEEmulticol`, 67  
`\IEEEnonumber`, 67  
`\IEEEyesnumber`, 67  
`\IEEEyessubnumber`, 68  
`\ignorespaces`, 126  
`\ignorespacesafterend`, 126  
`\iiiint`, 70  
`\iiint`, 70  
`\iint`, 70  
`\include`, 16  
`\includegraphics`, 45, 139  
`\includeonly`, 16  
`\indent`, 133  
`\index`, 89, 90  
`\inf`, 57  
`\input`, 16  
`\institute`, 103  
`\int`, 59  
`\item`, 38  
`\ker`, 57  
`\kwle`, 124  
`\label`, 35, 36, 48  
`\LaTeX`, 24  
`\LaTeXe`, 24  
`\ldots`, 27, 56  
`\left`, 60  
`\lefteqn`, 64, 66  
`\leftmark`, 90, 92  
`\lg`, 57  
`\lim`, 57  
`\liminf`, 57  
`\limsup`, 57  
`\line`, 108, 115  
`\linebreak`, 22  
`\linespread`, 132  
`\linethickness`, 112, 113, 116  
`\listoffigures`, 48  
`\listoftables`, 48  
`\ln`, 57  
`\log`, 57  
`\mainmatter`, 35, 97  
`\makebox`, 139, 140  
`\makeindex`, 89  
`\maketitle`, 35  
`\marginparpush`, 136  
`\marginparsep`, 136  
`\marginparwidth`, 136  
`\mathbb`, 55  
`\max`, 57  
`\mbox`, 23, 24, 27, 139  
`\medskip`, 135  
`\min`, 57  
`\multicolumn`, 43

`\multicolumns`, 66  
`\multipt`, 107, 112  
`\negmedspace`, 67  
`\newcommand`, 70, 124, 125  
`\newenvironment`, 125, 126  
`\newline`, 21  
`\newpage`, 22  
`\newsavebox`, 115  
`\newtheorem`, 72  
`\noindent`, 133  
`\nolinebreak`, 22  
`\nonumber`, 67  
`\nopagebreak`, 22  
`\not`, 78  
`\oddsidemargin`, 136  
`\only`, 104  
`\oval`, 113, 115  
`\overbrace`, 57  
`\overleftarrow`, 57  
`\overline`, 56  
`\overrightarrow`, 57  
`\pagebreak`, 22, 135  
`\pageref`, 35, 94  
`\pagestyle`, 14  
`\paperheight`, 136  
`\paperwidth`, 136  
`\par`, 130, 132  
`\paragraph`, 33  
`\parbox`, 139  
`\parindent`, 132  
`\parskip`, 132  
`\part`, 33, 34  
`\partial`, 59  
`\pause`, 104  
`\phantom`, 70  
`\pmod`, 58  
`\Pr`, 57  
`\printindex`, 90  
`\prod`, 59  
`\providecommand`, 125  
`\ProvidesPackage`, 128  
`\put`, 107–111, 113, 115  
`\qbezier`, 105, 107, 116  
`\qedhere`, 74, 75  
`\qquad`, 54, 70  
`\quad`, 54, 66, 69, 70  
`\raisebox`, 140  
`\ref`, 35, 36, 48, 94  
`\renewcommand`, 125  
`\renewenvironment`, 126  
`\right`, 60, 68  
`\right.`, 60  
`\rightmark`, 90, 92  
`\rule`, 44, 126, 141  
`\savebox`, 115  
`\scriptscriptstyle`, 71  
`\scriptstyle`, 71  
`\sec`, 57  
`\section`, 33, 34, 103  
`\sectionmark`, 90, 92  
`\setlength`, 107, 132, 137  
`\settodepth`, 138  
`\settoheight`, 138  
`\settowidth`, 138  
`\sin`, 57  
`\sinh`, 57  
`\slash`, 25  
`\sloppy`, 22  
`\smallskip`, 135  
`\smash`, 54  
`\sqrt`, 56  
`\stackrel`, 59  
`\stretch`, 126, 134, 135  
`\subparagraph`, 33  
`\subsection`, 33, 103  
`\subsectionmark`, 92  
`\substack`, 59  
`\subsubsection`, 33  
`\sum`, 59  
`\sup`, 57

- `\surd`, 56
  - `\tabcolsep`, 43
  - `\tableofcontents`, 34, 48
  - `\tag`, 52
  - `\tan`, 57
  - `\tanh`, 57
  - `\temporal`, 104
  - `\TeX`, 24
  - `\texorpdfstring`, 98
  - `\textbackslash`, 6
  - `\textbf`, 131
  - `\textcelsius`, 26
  - `\textdegree`, 26
  - `\texteuro`, 26
  - `\textheight`, 136
  - `\textstyle`, 71
  - `\textwidth`, 136
  - `\tfrac`, 58
  - `\theoremstyle`, 73
  - `\thicklines`, 109, 113, 116
  - `\thinlines`, 113, 116
  - `\thispagestyle`, 14
  - `\title`, 35, 103
  - `\titlegraphic`, 103
  - `\today`, 24
  - `\topmargin`, 136
  - `\totalheight`, 140
  - `\ud`, 70
  - `\uncover`, 104
  - `\underbrace`, 57
  - `\underline`, 37, 56
  - `\unitlength`, 106, 107, 109
  - `\updownarrow`, 60
  - `\usebox`, 115
  - `\usepackage`, 11, 15, 26, 127
  - `\usetikzlibrary`, 120
  - `\vdots`, 56
  - `\vec`, 57
  - `\vector`, 109
  - `\verb`, 40, 41
  - `\verbatiminput`, 92
  - `\vspace`, 134
  - `\vspace*`, 135
  - `\widehat`, 57
  - `\widetilde`, 57
  - `\width`, 140
  - polyglossia, 23, 28, 30–32
  - POSTSCRIPT, 4, 106, 143, 146
    - Encapsulated, 145
  - poziome
    - klamra, 57
    - kreska, 56
    - kropki, 56
  - ppower4, 101
  - `\Pr`, 57
    - preambuła, 8
    - prim, 57
  - `\printindex`, 90
  - `\prod`, 59
    - proof, 74
  - prosper, 101
  - proste, 129
  - `\providecommand`, 125
  - `\ProvidesPackage`, 128
    - przecinek, 27
    - pstricks, 110
    - pudełko, 138
  - `\put`, 107–111, 113, 115
    - półpauza, 25
- Q**
- `\qbezier`, 105, 107, 116
  - `\qedhere`, 74, 75
  - `\qqquad`, 54, 70
  - `\quad`, 54, 66, 69, 70
    - quotation, 39
    - quote, 39
- R**
- `\raisebox`, 140

- `\ref`, 35, 36, 48, 94
- `\renewcommand`, 125
- `\renewenvironment`, 126
- report
  - klasa, 11
- `\right`, 60, 68
- `\right.`, 60
- `\rightmark`, 90, 92
- rozmiar fontu, 129
- rozmiar kartki, 12, 135
- rozszerzenia, 14
  - .aux, 15
  - .cls, 15
  - .dtx, 15, 93
  - .dvi, 15
  - .fd, 15
  - .glo, 93
  - .idx, 15, 89
  - .ilg, 16
  - .ind, 16, 90
  - .ins, 15, 93
  - .lof, 15
  - .log, 15
  - .lot, 15
  - .sty, 15, 93, 127
  - .tex, 9, 14
  - .toc, 15
- `\rule`, 44, 126, 141
- równanie, 52
  - L<sup>A</sup>T<sub>E</sub>X, 53
  - amsmath, 53
  - wiele, 62
- S**
- `\savebox`, 115
- `\scriptscriptstyle`, 71
- `\scriptsize`, 129
- `\scriptstyle`, 71
- `\sec`, 57
- `\section`, 33, 34, 103
- `\sectionmark`, 90, 92
- `\setlength`, 107, 132, 137
- `\settodepth`, 138
- `\settoheight`, 138
- `\settowidth`, 138
- showidx, 90
- `\sin`, 57
- Sindhi, 31
- `\sinh`, 57
- skorowidz, 88
- `\slash`, 25
- `\sloppy`, 22
- `\small`, 129
- `\smallskip`, 135
- `\smash`, 54
- `\sqrt`, 56
- `\stackrel`, 59
- stopka, 14
- `\stretch`, 126, 134, 135
- struktura, 8
- strut, 44
- strzałki, 57
- styl strony
  - empty, 14
- styl pokazowy, 52, 53
- styl strony, 14
  - headings, 14
  - plain, 14
- styl tekstowy, 52, 53
- `\subparagraph`, 33
- `\subsection`, 33, 103
- `\subsectionmark`, 92
- `\substack`, 59
- `\subsubsection`, 33
- `\sum`, 59
- `\sup`, 57
- `\surd`, 56
- symbol
  - końca dowodu, 74

- syntonly, 13, 16  
 szeryfowe, 129  
 Słowo, 90
- T**
- $\backslash$ tabcolsep, 43  
 tabele, 41  
 table, 46, 47  
 $\backslash$ tableofcontents, 34, 48  
 tabular, 41, 43, 44, 139  
 $\backslash$ tag, 52  
 $\backslash$ tan, 57  
 $\backslash$ tanh, 57  
 $\backslash$ temporal, 104  
 $\backslash$ TeX, 24  
 TeXnicCenter, 145  
 $\backslash$ texorpdfstring, 98  
 $\backslash$ textbackslash, 6  
 $\backslash$ textbf, 129, 131  
 $\backslash$ textcelsius, 26  
 textcomp, 26  
 $\backslash$ textdegree, 26  
 $\backslash$ texteuro, 26  
 $\backslash$ textheight, 136  
 $\backslash$ textit, 129  
 $\backslash$ textmd, 129  
 $\backslash$ textnormal, 129  
 $\backslash$ textrm, 129  
 $\backslash$ textsc, 129  
 $\backslash$ textsf, 129  
 $\backslash$ textsl, 129  
 $\backslash$ textstyle, 71  
 $\backslash$ texttt, 129  
 $\backslash$ textup, 129  
 $\backslash$ textwidth, 136  
 $\backslash$ tfrac, 58  
 The L<sup>A</sup>T<sub>E</sub>X Project, 2  
 thebibliography, 87  
 $\backslash$ theoremstyle, 73  
 $\backslash$ thicklines, 109, 113, 116
- $\backslash$ thinlines, 113, 116  
 $\backslash$ thispagestyle, 14  
 tikz, 106, 119  
 tikzpicture, 119  
 $\backslash$ tiny, 129  
 $\backslash$ title, 35, 103  
 $\backslash$ titlegraphic, 103  
 $\backslash$ today, 24  
 $\backslash$ topmargin, 136  
 $\backslash$ totalheight, 140  
 tryb matematycznym, 54  
 tryb tekstowy, 54  
 turecki, 31  
 tylda, 57  
 tylda (~), 32  
 typy plików, 14  
 tytuł, 12  
 tytuł dokumentu, 12
- U**
- $\backslash$ ud, 70  
 Uighur, 32  
 Ukośnik, 25  
 ukośnik wsteczny, 6  
 układ strony, 135  
 umlaut, 28  
 $\backslash$ uncover, 104  
 $\backslash$ underbrace, 57  
 underfull hbox, 22  
 $\backslash$ underline, 37, 56  
 $\backslash$ unitlength, 106, 107, 109  
 $\backslash$ updownarrow, 60  
 Urdu, 31  
 $\backslash$ usebox, 115  
 $\backslash$ usepackage, 11, 15, 26, 127  
 $\backslash$ usetikzlibrary, 120  
 ułamek  
     piętrowy, 58

**V**

`\vdots`, 56  
`\vec`, 57  
`\vector`, 109  
`\verb`, 40, 41  
  `verbatim`, 7, 92  
  `verbatim`, 40, 92  
`\verbatiminput`, 92  
  `verse`, 39  
  `Vmatrix`, 69  
  `vmatrix`, 69  
`\vspace`, 134  
`\vspace*`, 135

**W**

wektor, 57  
`\widehat`, 57  
`\widetilde`, 57  
`\width`, 140  
wielokropek, 27  
współczynnik dwumianowy, 59  
wstawki nieszttywne, 46  
wykładnik, 55  
wymiar strony, 135  
wymiary, 134  
wymiary kartki, 12  
wyrównanie  
  do lewej, 38  
  do prawej, 38  
WYSIWYG, 3

**X**

`xeCJK`, 32  
`XLaTeX`, 98  
`xepersian`, 31  
`XYTeX`, 98  
`xgreek`, 31

**Z**

zalety L<sup>A</sup>T<sub>E</sub>Xa, 3  
zbiory (symbole), 55

znak

  całki, 59  
  sumowania, 59  
znak minus, 25  
znaki zarezerwowane, 5  
znaki niedrukowalne, 5