Nie za krótkie wprowadzenie do systemu LaTeX 2ε

Albo LaTeX 2ε w 88 minut

Tobias Oetiker

Hubert Partl, Irene Hyna i Elisabeth Schlegl

wersja 3.2, 21 września 1998

tłumaczenie i opracowanie wersji polskiej: Janusz Gołdasz, Ryszard Kubiak, Tomasz Przechlewski

współpraca redakcyjna: Stanisław Wawrykiewicz

Podziękowania

Większość materiału w tym dokumencie pochodzi z austriackiego *Wprowadzenia do LaTeX-a 2.09* w języku niemieckim, którego autorami są:

Hubert Partl <partl@mail.boku.ac.at>
Zentraler Informatikdienst der Universität für Bodenkultur Wien

Irene Hyna <Irene.Hyna@bmwf.ac.at>
Bundesministerium für Wissenschaft und Forschung Wien

Elisabeth Schlegl <no email>
Graz

Osoby zainteresowane wersją niemiecką, dostosowaną przez Jörga Knappena do LaTeX 2$_\varepsilon$, mogą ją znaleźć w `CTAN://tex-archive/info/lkurz`.

Pisząc *Wprowadzenie* korzystałem z pomocy wielu uczestników grupy dyskusyjnej `comp.text.tex`. Niżej wymienione osoby pomogły mi swoimi poprawkami, sugestiami i propozycjami ulepszeń. Chciałbym niniejszym serdecznie im za to podziękować. Oczywiście za wszystkie błędy, które nieuchronnie pozostały w tekście, wyłącznie ja ponoszę odpowiedzialność. Wszystkie zdania bezbłędne w książce są wyłączną zasługą osób z poniższej listy:

Rosemary Bailey, David Carlisle, Christopher Chin,
Chris McCormack, Wim van Dam, David Dureisseix, Elliot,
David Frey, Robin Fairbairns, Alexandre Guimond,
Cyril Goutte, Greg Gamble, Neil Hammond,
Rasmus Borup Hansen, Martien Hulsen, Eric Jacoboni,
Alan Jeffrey, Byron Jones, David Jones,
Johannes-Maria Kaltenbach, Andrzej Kawalec, Alain Kessi,
Christian Kern, Jörg Knappen, Maik Lehradt,
Martin Maechler, Claus Malten, Hubert Partl, John Refling,
Mike Ressler, Brian Ripley, Young U. Ryu, Chris Rowley,
Craig Schlenter, Josef Tkadlec, Didier Verna, Fritz Zaucker,
i Rick Zaccone.

Przedmowa

LaTeX [4] jest systemem składu umożliwiającym między innymi tworzenie dokumentów naukowych i technicznych o wysokiej jakości typograficznej. Oczywiście, oprócz tego LaTeX umożliwia przygotowywanie dowolnego rodzaju dokumentów, poczynając od listów, a kończąc na grubych książkach. Do formatowania dokumentów LaTeX wykorzystuje program TeX [3].

Niniejsze krótkie *Wprowadzenie* opisuje LaTeX-a w zakresie wystarczającym do większości zastosowań. Pełny opis LaTeX-a można znaleźć w [4, 2].

LaTeX jest dostępny na większości platform sprzętowych, począwszy od PC i Macintosh, a skończywszy na dużych systemach wyposażonych w UNIX-a czy VMS. W 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* [5]. Jeżeli nie wiesz jak zacząć pracę z LaTeX-em to zapytaj się osoby, od której otrzymałeś to *Wprowadzenie.* W tym dokumencie nie poruszamy spraw związanych z instalowaniem i konfiguracją systemu LaTeX. Mówimy wyłącznie o tym, jak pisać dokumenty, aby mogły zostać przetwarzane przez LaTeX-a.
Wprowadzenie zawiera pięć następujących rozdziałów:

Rozdział 1 przedstawia ogólną strukturę dokumentów LaTeX-owych i omawia krótko historię systemu LaTeX. Po przeczytaniu tego rozdziału powinieneś mieć już wyobrażenie, czym jest LaTeX. Chociaż będzie to tylko obraz ogólny, pozwoli ci na połączenie informacji, jakie znajdziesz w pozostałych rozdziałach, w całościowy obraz.

Rozdział 2 podaje szczegóły dotyczące składanie dokumentów. Omawia najważniejsze instrukcje i środowiska. Po przeczytaniu tego rozdziału będziesz umiał tworzyć proste dokumenty LaTeX-owe.

Rozdział 3 poświęcono składaniu wzorów matematycznych. Wiele zamieszczonych tu przykładów pozwoli ci nauczyć się wykorzystywać jedną z najważniejszych możliwości LaTeX-a, jaką jest elegancki skład matematyki. Na końcu rozdziału zamieszczono zestawienie dostępnych w LaTeX-u symboli matematycznych.

Rozdział 4 wyjaśnia, jak tworzyć skorowidze i spisy bibliograficzne, jak dołączać rysunki w formacie EPS i jeszcze parę innych pożytecznych rzeczy.

Rozdział 5 zawiera informacje, których wykorzystanie jest potencjalnie niebezpieczne, bo mówią, jak można zmienić standardowy układ graficzny dokumentów LaTeX-owych. Niewłaściwe ich użycie może spowodować pogorszenie ładnego na ogół składu LaTeX-owego.

Sądzimy, że należy przeczytać wszystkie rozdziały i w powyższej kolejności. Ostatecznie, książka nie jest zbyt gruba. Szczególnie starannie należy czytać przykłady, gdyż właśnie w nich zawarto sporo informacji.

Gdybyś potrzebował innych materiałów dotyczących LaTeX-a, zajrzyj do jednego z archiwów sieci CTAN. Węzeł CTAN w Stanach Zjednoczonych ma adres `ftp://ctan.tug.org`, w Niemczech `ftp://ftp.dante.de` a w Wielkiej Brytanii `ftp://ftp.tex.ac.uk`. Wybierz najbliższy lub jedno z wielu archiwów lustrzanych.

W katalogu `CTAN://tex-archive/systems` znajdziesz rzeczy niezbędne do uruchomienia LaTeX-a na twoim komputerze.

Od tłumaczy

Ponieważ uznaliśmy, że pewne tematy autorzy *Wprowadzenia* przedstawili zbyt lakonicznie, pozwoliliśmy sobie na ich poszerzenie. Fragmenty takie oznaczamy krojem bezszeryfowym. W szczególności, poszerzenia dotyczą problemu przygotowywania dokumentów w języku polskim, czego z oczywistych względów nie omówiono w oryginale.

Jeżeli masz pomysł, co należałoby dodać, usunąć lub zmienić w tym dokumencie – napisz. Jesteśmy szczególnie zainteresowani opiniami początkujących użytkowników LaTeX-a, które fragmenty są trudno zrozumiałe i wymagają lepszego przedstawienia.

Janusz Gołdasz

Ryszard Kubiak <`R.Kubiak@ipipan.gda.pl`>
Polska Grupa Użytkowników Systemu TeX

Tomasz Przechlewski <`T.Przechlewski@GUST.org.pl`>
Polska Grupa Użytkowników Systemu TeX

Z autorem oryginalnej wersji angielskiej możesz się skontaktować, pisząc na adres: Tobias Oetiker (`oetiker@ee.ethz.ch`).

Najnowszą wersję przekładu na język polski można znaleźć w:
`CTAN://tex-archive/info/lshort/polish/`

Najnowsza anglojęzyczna wersja *Wprowadzenia* znajduje się w:
`CTAN://tex-archive/info/lshort`

Spis treści

Podziękowania iii

Przedmowa iv

1. Podstawy, które warto znać 1
- 1.1. Nazwa programu 1
  - 1.1.1. TeX 1
  - 1.1.2. LaTeX 1
- 1.2. Podstawy 3
  - 1.2.1. Autor, redaktor i zecer 3
  - 1.2.2. Układ graficzny 3
  - 1.2.3. Zalety i wady 4
- 1.3. Plik źródłowy 5
  - 1.3.1. Odstępy 5
  - 1.3.2. Znaki specjalne 5
  - 1.3.3. Polecenia LaTeX-a 6
  - 1.3.4. Komentarze 7
- 1.4. Struktura pliku źródłowego 7
- 1.5. Układ graficzny dokumentu 8
  - 1.5.1. Klasy dokumentów 8
  - 1.5.2. Pakiety 9
  - 1.5.3. Style strony 10
- 1.6. Duże dokumenty 11

2. Składanie tekstu 13
- 2.1. Struktura tekstu i języka 13
- 2.2. Składanie akapitów i łamanie stron 15
  - 2.2.1. Składanie akapitów 15
  - 2.2.2. Przenoszenie wyrazów 17
- 2.3. Znaki specjalne i symbole 18
  - 2.3.1. Cudzysłowy 18
  - 2.3.2. Pauzy i myślniki 19
  - 2.3.3. Wielokropek (…) 20

2.3.4. Ligatury 20
2.3.5. Akcenty i znaki specjalne 20
2.3.6. Niełamliwe odstępy 21
2.4. LaTeX wielojęzyczny 21
2.4.1. Wsparcie dla języka polskiego 23
2.5. Odstępy między wyrazami 27
2.6. Tytuły, śródtytuły i punkty 27
2.7. Odsyłacze 29
2.8. Przypisy 30
2.9. Wyróżnienia 30
2.10. Środowiska 31
2.10.1. Środowiska `itemize`, `enumerate` i `description` 31
2.10.2. Środowiska `flushleft`, `flushright` i `center` 32
2.10.3. Środowiska `quote`, `quotation` i `verse` 32
2.10.4. Symulacja maszynopisu 33
2.10.5. Środowisko `tabular` 34
2.11. Wstawki 35
2.12. Listy 38

3. Wyrażenia matematyczne 40
3.1. Wstęp 40
3.2. Grupowanie 42
3.3. Części składowe wyrażeń matematycznych 42
3.4. Odstępy w trybie matematycznym 45
3.5. Wyrównywanie w pionie 46
3.6. Stopień pisma 48
3.7. Twierdzenia, definicje, itp. 49
3.8. Symbole półgrube 50
3.9. Zestawienie symboli matematycznych 51

4. Rysunki, skorowidze ... 58
4.1. Dołączanie grafiki w formacie EPS 58
4.2. Spis literatury 60
4.3. Skorowidze 61
4.4. Paginy górne i dolne 62
4.5. Pakiet `verbatim` 63

5. Adaptowanie LaTeX-a 65
5.1. Definiowane instrukcji i środowisk 65
5.1.1. Instrukcje definiowane przez użytkownika 66
5.1.2. Środowiska definiowane przez użytkownika 67
5.1.3. Własne pakiety 67
5.2. Fonty 68
5.2.1. Instrukcje przełączające stopień pisma 68

5.2.2. Uwaga niebezpieczeństwo! 70
5.3. Odstępy 71
5.3.1. Zmiana wielkości interlinii 71
5.3.2. Wcięcia i odstępy pomiędzy akapitami 72
5.3.3. Odstępy poziome 72
5.3.4. Odstępy pionowe 74
5.4. Układ graficzny strony 74
5.5. Więcej o odległościach 76
5.6. Pudełka 77
5.7. Kreski i podpory 79

Spis literatury **81**

Spis rysunków

1.1. Składniki systemu TEX 2
1.2. Zawartość minimalnego pliku źródłowego 7
1.3. Przykład artykułu 8
4.1. Przykład wykorzystania pakietu fancyhdr 63
5.1. Przykładowy pakiet 68
5.2. Parametry układu graficznego strony 75

Spis tabel

1.1. Klasy dokumentów 8
1.2. Ważniejsze opcje klas dokumentów 9
1.3. Wybrane pakiety z podstawowej dystrybucji LaTeX-a 10
1.4. Standardowe style strony w LaTeX-u 11
2.1. Akcenty i znaki specjalne 21
2.2. Opcjonalny argument środowisk `table` i `figure` 36
3.1. Akcenty matematyczne 51
3.2. Małe litery alfabetu greckiego 51
3.3. Duże litery alfabetu greckiego 51
3.4. Symbole relacji 52
3.5. Symbole operacji dwuargumentowych 53
3.6. Symbole zmiennej wielkości 53
3.7. Strzałki 53
3.8. Ograniczniki 54
3.9. Duże ograniczniki 54
3.10. Symbole różne 54
3.11. Symbole niematematyczne 54
3.12. Ograniczniki (pakiet AMS) 54
3.13. Symbole Greckie i Hebrajskie (pakiet AMS) 54
3.14. Symbole relacji (pakiet AMS) 55
3.15. Strzałki (pakiet AMS) 55
3.16. Negacje symbolów relacji i strzałek (pakiet AMS) 56
3.17. Relacje dwuargumentowe (pakiet AMS) 56
3.18. Różne symbole (pakiet AMS) 57
3.19. Kroje pisma dostępne w trybie matematycznym 57
4.1. Nazwy kluczy w pakiecie graphicx 59
4.2. Przykłady składni polecenia `\index` 62
5.1. Polecenia wyboru krojów i odmian 69
5.2. Polecenia wyboru stopnia pisma i interlinii 69
5.3. Wielkość stopnia pisma w klasach standardowych 70
5.4. Polecenia wyboru fontów w trybie matematycznym 71
5.5. LaTeX-owe jednostki miary 73

Rozdział 1

Podstawy, które warto znać

W pierwszej części tego rozdziału przedstawimy krótko filozofię i historię systemu LaTeX 2ε. W części drugiej przedstawimy podstawowe struktury dokumentu LaTeX-owego. Po przeczytaniu tego rozdziału czytelnik powinien z grubsza wiedzieć, jak działa LaTeX, a wiedza ta pozwoli na łatwiejsze przyswojenie materiału przedstawionego w następnych rozdziałach.

1.1. Nazwa programu

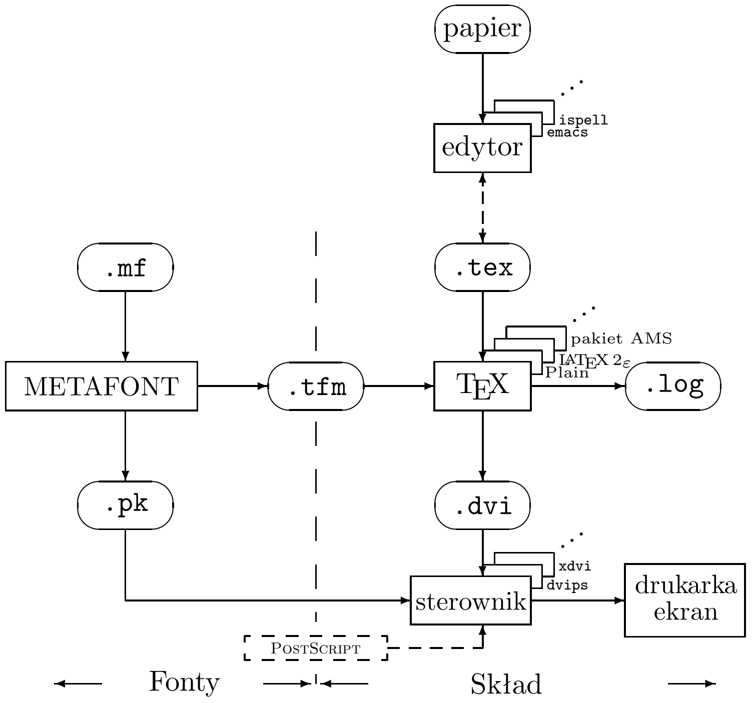
1.1.1. TeX

TeX jest programem komputerowym stworzonym przez Donalda E. Knutha [3]. Jest przeznaczony do składu tekstów oraz wzorów matematycznych. Knuth rozpoczął pracę nad TeX-em w 1977 roku rozczarowany tym, jak wyglądały jego własne prace naukowe publikowane przez *American Mathematical Society*. Pewnego dnia 1974 roku zaprzestał on nawet wysyłania swoich prac do publikacji, ponieważ stwierdził: „zbyt przykre było dla mnie oglądanie ostatecznego wyniku". W postaci używanej obecnie TeX został udostępniony w roku 1982, i od tego czasu zmienił się tylko w niewielkim stopniu. TeX jest programem bardzo stabilnym. Dziś – jak utrzymuje Knuth – jest w zasadzie programem nie zawierającym błędów. Numery wersji TeX-a zbiegają do liczby π, a obecny numer wynosi 3,14159.

Słowo TeX należy wymawiać „tech". W sytuacjach, w których nie można zapisywać nazwy TeX z charakterystycznym obniżeniem litery E, należy zamiennie używać wersji `TeX`.

1.1.2. LaTeX

LaTeX jest zestawem instrukcji (poleceń, makrodefinicji, makr) umożliwiających autorom złożenie i wydrukowanie ich prac na najwyższym poziomie typograficznym. Pierwszą wersję LaTeX-a opracował Leslie Lamport [4]. Do formatowania dokumentu LaTeX wykorzystuje TeX-a.



Prostokąty na rysunku oznaczają programy, a owale – pliki. W szczególności, z papierowego dokumentu za pomocą edytora (np. `emacs`) tworzymy dokument źródłowy (plik z rozszerzeniem `.tex`). Z dokumentu źrółowego po przetworzeniu LaTeX-em otrzymujemy plik `.dvi`, który jest z kolei plikiem wejściowym dla sterowników potrafiących go wydrukować bądź zamienić na inny format, np. PostScript. LaTeX zapisuje do pliku `.log` komunikaty podczas przetwarzania dokumentu. Pliki z rozszerzeniem `.tfm`, które *czyta* LaTeX zawierają informacje o fontach wykorzystywanych do składu.

Rysunek 1.1: Składniki systemu TeX

Kilka lat temu pakiet LaTeX został rozszerzony przez tak zwaną drużynę LaTeX3 (*LaTeX3 team*), kierowaną przez Franka Mittelbacha. Celem tego rozszerzenia było wprowadzenie kilku od dawna postulowanych ulepszeń oraz unifikacja rozmaitych odmian LaTeX-a, jakie rozpowszechniły się od chwili powstania kilkanaście lat temu LaTeX-a w wersji 2.09. Nowa wersja pakietu nazywa się LaTeX 2$_\varepsilon$ w celu odróżnienia jej od wersji poprzednich. Niniejszy dokument opisuje właśnie LaTeX 2$_\varepsilon$.

Słowo LaTeX należy wymawiać „lej-tech” albo „la-tech”. Jeżeli nie można zapisać symbolu LaTeX, można zamiennie używać wersji `LaTeX`. LaTeX 2$_\varepsilon$ wymawiamy „lej-tech tu i” a zamienną wersją nazwy jest `LaTeX2e`.

Rysunek 1.1 pokazuje związek pomiędzy programem TeX i pakietem LaTeX 2$_\varepsilon$. Rysunek zaczerpnięto z pliku `wots.tex` (autor Kees van der Laan).

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 itd.). Redaktor zapisuje swoje decyzje w maszynopisie, w formie odpowiednich instrukcji, i przekazuje go zecerowi. Na podstawie maszynopisu oraz instrukcji zecer wykonuje skład.

Redaktor-człowiek odgaduje intencje autora. Wykorzystując swoje doświadczenie zawodowe, na podstawie zawartości maszynopisu, ustala, które fragmenty tekstu są tytułami rozdziałów, podrozdziałów, cytatami, przykładami, wzorami matematycznymi itd.

LaTeX jest redaktorem, a TeX – zecerem. Z tym, że LaTeX jest „tylko" programem komputerowym i dlatego potrzebuje dodatkowej pomocy autora, który powinien dostarczyć niezbędnych do składu informacji o logicznej strukturze dokumentu. Informacje te zapisuje się w tekście jako „polecenia dla LaTeX-a".

Praca z LaTeX-em zdecydowanie różni się od podejścia stosowanego w procesorach tekstu typu WYSIWYG[1], takich jak *MS Word* albo *Corel WordPerfect*. Pierwszy sposób można określić „formatowaniem logicznym" drugi „formatowaniem wizualnym". Używając programów typu WYSIWYG autor interakcyjnie decyduje o wyglądzie graficznym dokumentu, w miarę dopisywania tekstu. Przez cały czas widzi na ekranie, jak tekst będzie wyglądał po wydrukowaniu.

Używając LaTeX-a, zwykle nie można oglądać dokumentu w jego ostatecznej postaci i zarazem dopisywać tekstu. Można natomiast obejrzeć dokument na ekranie po przetworzeniu go LaTeX-em[2]. Po naniesieniu niezbędnych poprawek dokument można wysłać na drukarkę.

1.2.2. Układ graficzny

Projektowanie książek jest sztuką. Amatorzy często popełniają poważny błąd zakładając, że projektowanie układu graficznego książki jest tylko kwestią estetyki (jeżeli dokument wygląda ładnie, to jest dobrze złożony). Ponieważ jednak dokumenty są przeznaczone do czytania, a nie do powieszenia jak obraz na ścianie w galerii, to o wiele większe znaczenie niż piękny wygląd ma łatwość czytania i przyswajania tekstu. Przykłady:

[1] Ang. *What you see is what you get* (dostaniesz to, co widzisz).

[2] Na szybkim komputerze przetworzenie trwa często zaledwie kilka sekund. Dysponując dużym ekranem, można jednocześnie wyświetlić okno z plikiem źródłowym oraz okno podglądu, otrzymując w ten sposób system (prawie) WYSIWYG.

- stopień pisma oraz numerację rozdziałów, podrozdziałów i punktów należy ustalić tak, aby czytelnik mógł się szybko zorientować w strukturze dokumentu;
- szerokość szpalty nie powinna być zbyt mała, aby nie zmuszać czytelnika do wytężania wzroku, ale zarazem wystarczająco duża, aby tekst elegancko wypełniał stronę.

W systemach wizualnych 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. Następnie LaTeX wybiera najodpowiedniejszy układ graficzny.

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 odwrotnie. Najlepiej podczas takich dyskusji siedzieć cicho. Czasami jednak nie uda się uniknąć zabrania głosu.

Poniżej zamieszczamy listę argumentów, które mogłyby być wykorzystane w takiej dyskusji. Oto największe zalety LaTeX-a w porównaniu ze zwykłymi procesorami tekstu.

- Dostępne są gotowe, przygotowane przez zawodowców układy graficzne, dzięki zastosowaniu których dokumenty wyglądają „jak z drukarni”.
- Wygodnie składa się wzory matematyczne.
- Do rozpoczęcia pracy wystarczy poznać zaledwie kilkanaście łatwych do zrozumienia instrukcji, określających strukturę logiczną dokumentu. Nie trzeba zaprzątać sobie głowy formatowaniem dokumentu.
- Nawet takie elementy jak przypisy, odnośniki, spisy treści, spisy tabel, skorowidze, oraz spisy bibliograficzne przygotowuje się bardzo łatwo.
- Istnieje wiele bezpłatnych pakietów poszerzających typograficzne możliwości LaTeX-a. Przykładowo, dostępne są pakiety umożliwiające wstawianie do dokumentów grafiki w formacie PostScript, tworzenie hipertekstowej wersji dokumentów w formacie PDF czy też przygotowanie spisów bibliograficznych według ściśle określonych reguł, obowiązujących w różnych wydawnictwach. Opis wielu z tych pakietów można znaleźć w podręczniku *The LaTeX Companion* [2].
- LaTeX zachęca autorów do tworzenia dokumentów o dobrze określonej strukturze.
- TeX – program formatujący używany przez LaTeX 2$_\varepsilon$ – jest bezpłatny i w najwyższym stopniu przenośny. Dzięki temu można działać na praktycznie każdej platformie systemowo-sprzętowej.

LaTeX ma także pewne braki, z tym że ciężko mi znaleźć chociaż jeden istotny. Jestem jednak pewien, że inne osoby wskażą ci ich setki`;-)`

- LaTeX nie działa u tych, którzy zaprzedali swoje dusze ...
- Aczkolwiek można dostosowywać predefiniowane układy graficzne do własnych potrzeb przez zmianę wybranych parametrów, to jednak zaprojektowanie całkowicie nowego układu jest czasochłonne[3].
- Trudno stworzyć dokument o nieokreślonej, bałaganiarskiej strukturze.

1.3. Plik źródłowy

Plikiem źródłowym LaTeX-a jest zwykły plik tekstowy (plik ASCII). Taki plik można utworzyć za pomocą dowolnego edytora tekstowego. Zawiera on tekst dokumentu oraz instrukcje, dzięki którym LaTeX wie, jak złożyć tekst.

1.3.1. Odstępy

Znaki *niewidoczne*, takie jak znaki odstępu (spacji) lub tabulacji są przez LaTeX-a traktowane jednakowo – jako odstęp. *Kolejno* po sobie występujące znaki odstępu LaTeX traktuje tak, jak *jeden* taki znak. Znaki odstępu znajdujące się na początku wiersza są prawie zawsze ignorowane. Pojedynczy koniec linii jest traktowany jak odstęp.

Pusty wiersz kończy akapit. Kolejno występujące puste wiersze LaTeX traktuje tak, jak jeden pusty wiersz. Poniższy tekst jest przykładem. 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 rozpoczyna nowy akapit.
```

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

Pusty wiersz rozpoczyna nowy akapit.

1.3.2. Znaki specjalne

Następujące znaki są zarezerwowane: albo mają specjalne znaczenie dla LaTeX-a, albo nie są dostępne we wszystkich standardowych krojach pisma. Użyte dosłownie w pliku źródłowym nie pojawią się na wydruku, tylko (najczęściej) spowodują błąd podczas przetwarzania tekstu.

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

Znaki te można jednak wstawić do dokumentu, o ile poprzedzi się je znakiem w-tył-ciacha (ang. *backslash*):

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

$ & % # _ { }

[3] Plotki mówią, że jest to jeden z ważniejszych problemów, nad jakim pracują twórcy systemu LaTeX3.

Znaku w-tył-ciach *nie można* wstawić do tekstu metodą podwojenia. Kombinacja `\\` jest poleceniem LaTeX-a, opisanym w punkcie 2.2.1. Znak w-tył-ciach możemy wstawić za pomocą polecenia `$\backslash$`. Uwaga: znaki dolara są niezbędne, a ich opuszczenie spowoduje błąd podczas przetwarzania.

1.3.3. Polecenia LaTeX-a

Polecenia LaTeX-a występują w dwóch następujących odmianach:

1. Instrukcje rozpoczynające się znakiem w-tył-ciach „\", po którym występuje ciąg liter. Końcem instrukcji jest odstęp lub inny znak nie będący literą. W nazwach instrukcji LaTeX rozróżnia litery małe i duże. Używanie polskich liter diakrytycznych w nazwach instrukcji nie jest możliwe.
2. Instrukcje składające się ze znaku w-tył-ciach oraz *dokładnie jednego* znaku nie będącego literą.

Znaki odstępu po instrukcjach typu 1 są przez LaTeX-a ignorowane. Jeżeli po instrukcji ma w dokumencie występować odstęp, to należy bezpośrednio po niej umieścić kolejno: parę nawiasów klamrowych `{}` i odstęp. Para znaków `{}` zapobiega zignorowaniu przez LaTeX-a odstępu po nazwie instrukcji. Innym sposobem jest wstawienie specjalnej instrukcji `\␣` (tj. w-tył-ciach i spacja). Niektórzy używają jeszcze innego sposobu zapobiegania połykaniu spacji po nazwach instrukcji, a mianowicie wstawiają instrukcję pomiędzy parę nawiasów `{` i `}`.

```
Czytałem, że Knuth dzieli
użytkowników systemu {\TeX} na
\TeX{}ników oraz \TeX pertów.\\
Dzisiaj jest \today.
```

Czytałem, że Knuth dzieli użytkowników systemu TeX na TeXników oraz TeXpertów.
Dzisiaj jest 11 kwietnia 2000 roku.

Niektóre instrukcje LaTeX-owe mają argumenty. Argumenty podaje się w nawiasach klamrowych `{ }`, każdy argument w osobnej parze nawiasów. Liczba i kolejność argumentów jest dla każdego polecenia z góry ustalona. Instrukcje mogą mieć także argumenty opcjonalne (nieobowiązkowe), podawane w nawiasach kwadratowych `[ ]`. W przypadku większej liczby argumentów opcjonalnych oddziela się je przecinkami. Kolejność argumentów opcjonalnych *nie odgrywa roli*.

Poniższe przykłady ilustrują postać instrukcji LaTeX-owych. Znaczenie tych instrukcji jest na razie nieistotne; zostanie podane później.

```
Możesz na mnie \textsl{polegać}!

Tu wstaw zmianę wiersza.\newline
Dziękuję.
```

Możesz na mnie *polegać*!

Tu wstaw zmianę wiersza.
Dziękuję.

1.3.4. Komentarze

Po napotkaniu znaku `%` LaTeX ignoruje resztę bieżącego wiersza (łącznie ze znakiem końca wiersza) oraz znaki odstępu na początku następnego. Znak `%` jest wykorzystywany do umieszczania komentarzy i dodatkowych informacji w pliku źródłowym.

```
Mao zmarł % sprawdzić!
     w~1975 roku.
```

Mao zmarł w 1975 roku.

Zdarza się, że znak `%` używany jest do rozdzielenia długich linii w pliku wejściowym, gdy niedozwolone jest użycie spacji lub złamanie wiersza.

1.4. Struktura pliku źródłowego

LaTeX oczekuje, że plik źródłowy posiada określoną strukturę. W szczególności, każdy plik źródłowy składa się z dwóch części: preambuły oraz części głównej. Preambuła powinna się rozpoczynać od instrukcji:

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

Polecenie to określa typ tworzonego dokumentu. Po nim można umieścić instrukcje dotyczące stylu całego dokumentu oraz dołączyć pakiety poszerzające możliwości LaTeX-a. Pakiety dołącza się za pomocą polecenia:

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

Część główna dokumentu zaczyna się od instrukcji `\begin{document}`, a kończy poleceniem `\end{document}`. Tekst znajdujący się za tym poleceniem jest przez LaTeX-a ignorowany.

Rysunek 1.2 pokazuje zawartość minimalnego dokumentu LaTeX-owego (użyte tu instrukcje `\usepackage{...}`, niezbędne do składania w języku polskim, omawiamy w punkcie 2.4).

```
\documentclass{article}
\usepackage[MeX]{polski}
\usepackage[latin2]{inputenc}
\begin{document}
Małe jest piękne.
\end{document}
```

Rysunek 1.2: Zawartość minimalnego pliku źródłowego

Rysunek 1.3 przedstawia zawartość nieco bardziej skomplikowanego pliku źródłowego.

```
\documentclass[a4paper,11pt]{article}
\usepackage{latexsym}
\author{H.~Partl}
\title{Minimalism}
\frenchspacing
\begin{document}
\maketitle \tableofcontents
\section{Start}
Well and here begins my lovely article.
\section{End}
\ldots{} and here it ends.
\end{document}
```

Rysunek 1.3: Przykład artykułu

1.5. Układ graficzny dokumentu

1.5.1. Klasy dokumentów

LaTeX musi wiedzieć do jakiej klasy (typu) należy przetwarzany dokument. Jest to określone w instrukcji `\documentclass`:

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

gdzie *klasa* oznacza typ dokumentu. W tabeli 1.1 zestawiono klasy dokumentów opisane w niniejszym wprowadzeniu. W dystrybucji LaTeX 2_ε znajdują się także inne klasy. *Opcje* umożliwiają modyfikowanie klas. Poszczególne opcje rodziela się przecinkami. W tablicy 1.2 zestawiono najczęściej wykorzystywane opcje dla standardowych klas.

Tabela 1.1: Klasy dokumentów

`article`	artykuły, krótkie opracowania ...
`report`	dłuższe opracowania, dysertacje magisterskie i doktorskie ...
`book`	książki.
`slides`	przeźrocza[a].
`letter`	listy.

[a] Przeźrocza składane są bezszeryfowym krojem pisma. Zamiast klasy standardowej można do tworzenia slajdów wykorzystać pakiet FoilTeX `CTAN://tex-archive/macros/latex/packages/supported/foiltex`.

Tabela 1.2: Ważniejsze opcje klas dokumentów

`10pt`, `11pt`, `12pt` Ustalenie stopnia pisma dla tekstu zasadniczego dokumentu. Domyślną wartością jest 10 punktów.

`a4paper`, `letterpaper`, ... Ustalenie wymiarów papieru. Wartością domyślną jest `letterpaper`. 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` Pierwsza powoduje, że LaTeX składa tytuł (instrukcja `maketitle`) oraz streszczenie (instrukcja `abstract`) na odzielnej stronie, druga rozpoczyna skład tekstu na stronie tytułowej. W klasie `article` domyślnie nie są składane na oddzielnych stronach; w stylu `report` i `book` są.

`twocolumn` Składu dwułamowy (dwukolumnowy).

`oneside, twoside` Druk na jednej lub na dwóch stronach kartki papieru. W klasach `article` i `report` domyślną opcją jest `oneside`, natomiast w klasie `book` – `twoside`. Włączenie opcji `oneside` powoduje przy okazji, że LaTeX nie wyrównuje wysokości kolejnych stron, dopuszczając pewną ich zmienność.

`openright, openany` Wybranie pierwszej opcji powoduje, że tytuły rozdziałów będą umieszczane na stronach nieparzystych. W klasie `article` opcja nie ma znaczenia ponieważ w tej klasie nie jest zdefiniowane pojęcie rozdziału. W klasie `report` domyślną wartością jest `openany`, w klasie book – `openright`.

Przykład. Plik źródłowy może rozpoczynać się od następującej instrukcji:

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

W tym wypadku dokument będzie składany w klasie *article*, pismem w stopniu *11 punktów*, będzie przygotowany do wydruku po *dwóch* stronach kartki papieru formatu *A4*.

1.5.2. Pakiety

Pakiety rozszerzają możliwości LaTeX-a. Przykładowo, LaTeX nie umożliwia dołączania grafiki, kolorowania tekstu, automatycznego łamania dużych tabel itp. Do wykonywania tych zadań służą właśnie pakiety. Dołącza się je za pomocą instrukcji:

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

Tabela 1.3: Wybrane pakiety z podstawowej dystrybucji LaTeX-a

doc	Służy do drukowania dokumentacji pakietów oraz innych części składowych LaTeX-a. Opis znajduje się w pliku `doc.dtx`[a].
exscale	Umożliwia skalowanie fontów matematycznych, tak aby były optycznie zgodne z otaczającym tekstem, np. w tytułach rozdziałów. Opis w `ltexscale.dtx`.
fontenc	Definiuje układ znaków w bieżącym zestawie znaków (foncie). Opis w `ltoutenc.dtx`.
ifthen	Umożliwia korzystanie z poleceń typu `if... then do... otherwise do`. Opis w `ifthen.dtx` i [2].
latexsym	Udostępnia specjalny font symboliczny (fonty `lasy`). Opis w `latexsym.dtx` i [2].
makeidx	Udostępnia polecenia do przygotowywania skorowidzów. Opis w punkcie 4.3 i [2].
syntonly	Przetwarza dokument bez składania czegokolwiek. Przydatny do szybkiego sprawdzenia czy dokument nie zawiera błędów. Opis w `syntonly.dtx` i [2].
inputenc	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 `inputenc.dtx`.

[a]Plik ten powinien być zainstalowany w twoim systemie. Aby otrzymać plik `dvi` wystarczy napisać `latex doc.dtx` w katalogu z prawem do zapisu. To samo stosuje się do innych pakietów z tej tabeli.

gdzie *pakiet* oznacza nazwę pakietu, a *opcje* – listę oddzielonych przecinkami opcji. Część pakietów znajduje się w podstawowej dystrybucji LaTeX 2_ε (zobacz tabela 1.3). Pozostałe są rozpowszechniane oddzielnie. Gdy używamy LaTeX-a w systemie, którym zarządza (dobry) administrator, to informacja o dostępnych pakietach powinna się znajdować w *Local Guide* [5]. Podstawowym źródłem informacji o LaTeX-u jest *The LaTeX Companion* [2]. Zawiera on opis setek pakietów a także opisuje, jak rozszerzać możliwości LaTeX-a.

1.5.3. Style strony

Typowa strona składa się z trzech podstawowych części. Powyżej kolumny tekstu głównego znajduje się *pagina górna* (główka), która może zawierać numer strony, tytuł rozdziału czy punktu. Poniżej kolumny tekstu znajduje się *pagina dolna* (stopka). W niniejszym wprowadzeniu pagina dolna jest pu-

Tabela 1.4: Standardowe style strony 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;
`empty` pagina górna i dolna są puste.

sta, a pagina górna zawiera numer strony oraz tytuł rozdziału na stronicach parzystych, a tytuł punktu na nieparzystych[4].

LaTeX umożliwia wybranie jednego spośród trzech sposobów składania pagin za pomocą instrukcji:

`\pagestyle{`*styl*`}`

Dopuszczalne wartości argumentu *styl* zestawiono w tabeli 1.4.

Możliwa jest także zmiana stylu *bieżącej* strony. Do tego celu służy instrukcja:

`\thispagestyle{`*styl*`}`

Podrozdział 4.4 na stronie 54 podręcznika [2] zawiera opis, jak można przygotowywać własne sposoby składania pagin.

1.6. Duże dokumenty

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

`\include{`*plik*`}`

Dołącza on do dokumentu zawartość *pliku.* Przed przystąpieniem do przetwarzania zawartości dołączonego pliku LaTeX rozpocznie nową stronę. Po złożeniu zawartości pliku LaTeX także rozpoczyna nową stronę.

Drugiej instrukcji używa się w preambule. Pozwala ona dołączyć do dokumentu tylko wybrane pliki.

`\includeonly{`*lista-plików*`}`

[4] Paginę zawierającą oprócz kolejnego numeru kolumny (strony), także informację dotyczącą treści tej kolumny drukarze nazywają *żywą paginą*.

Spośrod instrukcji `\include` zostaną wykonane tylko te, dla których nazwy plików są na liście plików polecenia `\includeonly`. Uwaga: w liście plików nie można używać odstępów!

Polecenie `\include` rozpoczyna skład dołączanego tekstu od nowej strony. W połączeniu z poleceniem `\includeonly` w preambule instrukcja `\inculde` umożliwia przetwarzanie wybranych plików bez zmiany miejsc łamania poszczególnych stron i z zachowaniem prawidłowej numeracji stron, rodziałów, tabel itp. Czasami jednak rozpoczynanie składu od nowej strony nie jest pożądane. W takiej sytuacji zamiast `\include` należy dołączyć plik za pomocą instrukcji:

`\input{`*plik*`}`

która po prostu wstawia zawartość podanego pliku bez żadnych dodatkowych efektów.

Rozdział 2

Składanie tekstu

Po lekturze poprzedniego rozdziału znamy już podstawowe pojęcia związane z systemem LaTeX 2ε. W tym rozdziale poznamy polecenia niezbędne do tworzenia prawdziwych dokumentów.

2.1. Struktura tekstu i języka

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

LaTeX tym się różni od innych systemów składu, że do złożenia tekstu wystarczają mu informacje dotyczące logicznej i semantycznej struktury tekstu. Informacje te określone są przez „reguły” zawarte w plikach definiujących klasy dokumentów i innych plikach zawierających instrukcje.

Najważniejszą jednostką podziału tekstu w systemie LaTeX (a także w typografii) jest akapit. Jest to „jednostka podziału” głównie dlatego, że treść akapitu powinna być poświęcona jednej spójnej myśli lub pojęciu. W następnym podrozdziale omówimy instrukcje nakazujące złamanie linii na przykład przy użyciu polecenia `\\`, a także sposób rozpoczęcia nowego akapitu poprzez pozostawienie pustej linii w kodzie źródłowym. Tak więc, rozpoczęcie nowej myśli jest równoznaczne z rozpoczęciem nowego akapitu. W pozostałych przypadkach powinno się jedynie przechodzić do nowej linii. W razie wątpliwości należy pamiętać, że tekst jest zawsze nośnikiem pewnych myśli oraz idei. Kontynuacja dotychczasowej myśli w nowym akapicie jest błędem. Niezgodna z regułami sztuki jest także sytuacja, gdy w tym samym akapicie pojawia się nowa myśl.

Większość ludzi całkowicie lekceważy znaczenie właściwej organizacji akapitów. Co więcej, wiele osób nawet nie zdaje sobie sprawy, czym akapit naprawdę jest i przechodzi do nowego akapitu nawet o tym nie wiedząc (dotyczy to zwłaszcza LaTeX-a). Ten błąd łatwiej popełnić zwłaszcza, gdy w tekście pojawiają się równania. 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 podrozdziału i przeczytać go jeszcze raz).

```
% Przykład 1
\ldots Słynne równanie Einsteina
\begin{equation}
e = m \cdot c^2 \; ,
\end{equation}
jest zarówno najbardziej znanym, ale także
najmniej rozumianym równaniem w~fizyce.

% Przykład 2
\ldots którego wynikiem jest prądowe prawo Kirchhoffa:
\begin{equation}
\sum_{k=1}{n} I_k = 0 \; .
\end{equation}

Natomiast, napięciowe prawo Kirchhoffa ma
swój początek w~\ldots

% Przykład 3
\ldots co ma określone zalety.

\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 angielskojęzycznych odstęp, który pojawia się po kropce kończącej zdanie, jest większy niż ten, który pojawia się po kropce oznaczającej skrót. Zależnie od kontekstu LaTeX stara się użyć krótszego lub dłuższego odstępu. W razie pomyłek z jego strony, powinniśmy mu wskazać nasze intencje – jak to zrobić, wyjaśniamy w dalszej części tego rozdziału.

Właściwa organizacja tekstu dotyczy także części zdań. Wiele języków ma bardzo skomplikowaną interpunkcję, ale w większości przypadków (wliczając angielski i niemiecki), stawiając przecinek w określonym miejscu w zdaniu, na ogół nie popełnimy błędu, jeśli będziemy pamiętać o zasadzie, że przecinek oznacza przerwę/przystanek w zdaniu. Dlatego – jeśli nie jesteśmy pewni, gdzie w zdaniu należy go postawić – przeczytajmy zdanie na głos i postawmy przecinki wszędzie tam, gdzie zrobiliśmy krótką przerwę na wdech. Ale nie sugerujmy się tylko tym! Jeśli w danym miejscu przecinek nie wygląda najlepiej, to usuńmy go; jeśli w innym miejscu odczuwamy potrzebę wzięcia powietrza do płuc (lub zrobienia krótkiej przerwy), to postawmy tam przecinek.

I na koniec: akapit nie jest największą logiczną jednostką podziału tekstu. Można jeszcze wspomnieć o rozdziałach, podrozdziałach, punktach itd. Jednakże od strony typograficznej już same nazwy poleceń w rodzaju `\section{Struktura tekstu i~języka}` są na tyle oczywiste, że sposób ich użycia jest samo przez się zrozumiały (dla znających język angielski).

2.2. Składanie akapitów i łamanie stron

2.2.1. Składanie akapitów

Książki najczęściej składa się tak, że wszystkie wiersze w akapitach są tej samej długości. Podczas składania akapitu, dążąc do optymalnej prezentacji, LaTeX ustala miejsca złamania linii oraz odstępy między słowami. W razie potrzeby LaTeX przenosi wyrazy, których nie jest w stanie zmieścić w wierszu. Konkretny sposób składania akapitów zależy od użytej klasy dokumentu. Najczęściej pierwszy wiersz akapitu jest wcięty, a pomiędzy akapitami nie ma dodatkowych odstępów. Więcej informacji na ten temat można znaleźć w punkcie 5.3.2.

Zgodnie z anglo-amerykańskimi zwyczajami typograficznymi LaTeX nie wstawia wcięcia akapitowego bezpośrednio po tytułach rozdziałów, podrozdziałów itd. Polskie zwyczaje nakazują rozpoczynanie także początkowych akapitów od wcięcia. Efekt ten można osiągnąć przez dołączenie w preambule pakietu `indentfirst`. W niniejszym tłumaczeniu nie stosujemy wcięć akapitowych, ponieważ akurat pod tym względem bardziej nam się podobają zwyczaje amerykańskie.

Czasami przydaje się instrukcja nakazująca LaTeX-owi złamanie linii. Polecenie

`\\` lub `\newline`

rozpoczyna nową linię bez rozpoczynania nowego akapitu.
Natomiast instrukcja:

`\\*`

dodatkowo zakazuje złamania strony (w miejscu złamania linii).
Instrukcja:

`\newpage`

rozpoczyna nową stronę.

Instrukcje:

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

oznaczają – odpowiednio – zachętę do złamania wiersza, niezgodę na złamanie wiersza, zachętę do złamania strony, niezgodę na złamanie strony, w miejscu ich wystąpienia w dokumencie. Opcjonalny argument n, o dopuszczalnej wartości od 0 do 4, określa stopień tej zachęty (niezgody). Domyślna wartość 4, to bezwarunkowy zakaz lub nakaz złamania linii/strony. Wartości mniejsze od 4 pozostawiają LaTeX-owi swobodę zignorowania instrukcji, jeżeli skład otrzymany w rezultacie byłby niskiej jakości.

Poleceń z grupy `break` nie należy mylić z poleceniami z grupy `new`. Mimo wydania polecenia typu `break` LaTeX stara się wypełnić wiersz do prawego skraju czy też wypełnić stronę do całej jej wysokości. Nietrudno zgadnąć, jakiej instrukcji należy użyć, gdy naprawdę zależy nam na rozpoczęciu nowego wiersza (ta zagadka jest łatwa pod warunkiem, że znamy język angielski – jest to polecenie `\newline`).

Jeżeli w wierszu zakończonym instrukcją `\newline` jest zbyt mało tekstu, to LaTeX nie wyrówna tego wiersza do prawego marginesu, tylko po tekście wstawi odpowiedni odstęp. Jeżeli zamiast `\newline` użyjemy `\linebreak`, to LaTeX postara się wyrównać zawartość kończonej linijki do prawego marginesu. Zbyt małe wypełnienie wiersza tekstem zmusi LaTeX-a do wstawienia zbyt dużych odstępów pomiędzy wyrazy w wierszu, co zostanie zasygnalizowane podczas przetwarzania wyświetleniem komunikatu:

```
Underfull \hbox (badness 10000) in paragraph at lines 4--5
```

W komunikacie tym liczba po słowie `badness` wskazuje, jak bardzo LaTeX-owi „nie podoba” się złożony wiersz. W tym przypadku jest to maksymalna wartość, 10000. Liczby na końcu komunikatu to numery pierwszej i ostatniej linii akapitu, w którym LaTeX musiał złożyć wiersz z nadmiernymi odstępami między wyrazami. Chociaż wielu użytkowników LaTeX-a nie zwraca uwagi na podobne ostrzeżenia, warto zdawać sobie sprawę, co one oznaczają.

Oprócz wyżej wymienionych istnieją w LaTeX-u jeszcze instrukcje:

`\clearpage`, `\cleardoublepage`

Obydwie rozpoczynają skład nowej strony. Instrukcja `\cleardoublepage` działa tak, jak `\clearpage`, z tym że rozpoczynana nowa strona jest zawsze nieparzysta. W razie potrzeby tworzona jest strona pusta. W składzie dwuszpaltowym (opcja `twocolumn`) instrukcja `\newpage` kończy szpaltę natomiast `\clearpage` oraz `\cleardoublepage` kończą stronę, w miarę potrzeby pozostawiając pustą prawą szpaltę.

Jeżeli na stronie zakończonej instrukcją `\newline` (albo `\clearpage`) jest zbyt mało tekstu, LaTeX wstawia odpowiedni odstęp u dołu strony, a nie pomiędzy akapity czy też inne elementy na stronie, tak jak ma to miejsce w przypadku instrukcji `\pagebreak`. Jeżeli te odstępy okazują się za duże (a zdarza się to dość często), to sytuacja taka jest sygnalizowana podczas przetwarzania dokumentu odpowiednim ostrzeżeniem, np.:

```
Underfull \vbox (badness 10000) has occurred
while \output is active [7]
```

W powyższym komunikacie (przełamanym, z uwagi na wąską szpaltę na dwie linijki) liczba po słowie `badness` wskazuje, jak bardzo LaTeX-owi nie podoba się złożona strona. Tutaj jest to `10000` – maksymalna w TeX-u ujemna ocena jakości składu. Liczba w nawiasach prostokątnych na końcu komunikatu oznacza numer strony, podczas składania której wystąpił problem.

LaTeX zawsze próbuje podzielić akapit na wiersze w sposób optymalny. Kiedy nie potrafi znaleźć podziału, który spełnia jego wymagania jakościowe, wtedy niektóre wyrazy wystają na prawym marginesie, a sytuacja taka jest sygnalizowana podczas przetwarzania pliku, np.:

```
Overfull \hbox (5.5452pt too wide) in paragraph at lines 79--83
```

W powyższym komunikacie liczba w nawiasie okrągłym oznacza, że pewien fragment tekstu wystaje o 5,5452 punktów drukarskich na prawy margines. Problem wystąpił w akapicie, który w pliku źródłowym jest w wierszach od 79 do 83. Podobne ostrzeżenia pojawiają się najczęściej wówczas, gdy LaTeX nie potrafi przenieść wyrazu zgodnie z zadanymi wzorcami tak, aby nie popsuć jakości składu. Ostrzeżenie powyższej postaci na ogół nie wystarcza do dokładnego ustalenia przyczyny jego wystąpienia. Można wtedy użyć opcji `draft` jako argumentu polecenia `\documentclass`, na skutek czego LaTeX oznaczy wystające wiersze małą czarną sztabką umieszczoną na prawym marginesie szpalty.

Deklaracja `\sloppy` nakazuje LaTeX-owi obniżyć nieco jego domyślnie wysokie standardy. Zapobiega to – w większości wypadków – występowaniu zbyt długich linijek, ale kosztem zwiększenia odstępów międzywyrazowych, nawet wtedy, gdy otrzymany skład nie jest optymalny. Takie linijki LaTeX sygnalizuje ostrzeżeniem (`underfull hbox`). W większości przypadków otrzymany wynik jest zupełnie niezły. Instrukcja `\fussy` działa w odwrotnym kierunku, to znaczy przywraca domyślne, wysokie standardy LaTeX-a.

2.2.2. Przenoszenie wyrazów

W razie potrzeby LaTeX przenosi (dzieli) wyrazy. Jeżeli algorytm podziału wyrazów przenosi jakiś wyraz nieprawidłowo, to prawidłowe miejsca przenoszenia można zadać instrukcją:

`\hyphenation{`*lista słów*`}`

Słowa z listy argumentów można dzielić tylko w miejscach oznaczonych znakiem `-`. Instrukcji tej powinno się używać wyłącznie w preambule dokumentu, a wyrazy-argumenty mogą zawierać (oprócz znaku `-`) wyłącznie litery. Nie ma natomiast znaczenia, czy w tych wyrazach używa się liter wielkich, czy małych. Wykonanie instrukcji `\hyphenation` z przykładu poniżej spowoduje, że słowo „ćwierć-li-trówka” można podzielić tylko w dwóch zaznaczonych miejscach, podczas gdy słowa „szczypce” nie można przenieść w ogóle. Wyrazy z listy nie mogą zawierać żadnych znaków specjalnych ani symboli.

Przykład:

```
\hyphenation{ćwierć-li-trówka szczypce}
```

W językach fleksyjnych, do jakich należy polski, instrukcja `\hyphenation` jest dużo mniej przydatna niż w angielskim. Aby dany wyraz zawsze był dobrze przenoszony, należałoby wypisać wszystkie jego formy. Co więcej, próba wykorzystania polecenia `\hyphenation` w przypadku plików źródłowych zapisanych w notacji prefixowej albo też plików w notacji naturalnej i korzystania z pakietu inputenc zakończy się błędem w czasie przetwarzania dokumentu. Ten fenomen jest spowodowany sposobem w jakim działa TEX, a jego wyjaśnienie daleko wykracza poza ramy niniejszego wprowadzenia.

Instrukcja „`\-`” wskazuje, w których miejscach LATEX-owi *wolno* przenieść wyraz w danym miejscu w dokumencie. LATEX nie podzieli słowa w innym miejscu. Instrukcja ta przydaje się szczególnie w przypadku wyrazów zawierających znaki specjalne (np. znaki akcentowane), ponieważ LATEX dzieli automatycznie tylko wyrazy złożone z samych liter.

```
Nie\-bie\-sko\-bia\-ło\-zie\-lo\-%
no\-nie\-bie\-ski
```

Niebieskobiałozielononiebieski

Tekst będący argumentem polecenia:

`\mbox{`*tekst*`}`

nigdy nie zostanie przeniesiony.

```
Numer mojego telefonu wkrótce
się zmieni na \mbox{0116 291 2319}.

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

Numer mojego telefonu wkrótce się zmieni na 0116 291 2319.

Parametr *nazwa* to nazwa pliku.

2.3. Znaki specjalne i symbole

2.3.1. Cudzysłowy

Znaku cudzysłowu `"` używa się inaczej niż na maszynie do pisania, bo w publikacjach drukowanych inaczej oznacza się początek i inaczej koniec cudzysłowu. Występują także różnice w sposobach oznaczania cudzysłowów w różnych językach. Dwa apostrofy `‘` otwierają, a dwa apostrofy `’` zamykają LATEX-owy cudzysłów według reguł języka angielskiego:

```
‘‘Please press the ‘x’ key.’’
```

“Please press the ‘x’ key.”

W języku polskim cudzysłów otwierający oznacza się za pomocą dwóch przecinków „`,,`”, a zamykający za pomocą dwóch apostrofów „`''`”[1].

Gdy zachodzi konieczność użycia cudzysłowu w tekście już objętym cudzysłowem, to stosuje się „cudzysłowy «francuskie»”, oznaczane w pliku źródłowym znakami mniejszości „<<” i – odpowiednio – większości „>>”.

```
,,Przechodź tylko po <<zebrach>>''.
```

„Przechodź tylko po «zebrach»”.

2.3.2. Pauzy i myślniki

Proste maszyny do pisania posiadają tylko jeden znak w kształcie poziomej kreski „-”. W składzie drukarskim występują aż cztery rodzaje kresek poziomych. Są to: łącznik, myślnik, półpauza i minus używany we wzorach matematycznych.

Łącznik (dywiz) jest najkrótszą z kresek; stosuje się go przy dzieleniu i przenoszeniu wyrazów oraz przy łączeniu wyrazów wieloczłonowych (np. „niebiesko-czarny”). Zgodnie z polskimi regułami wyraz wieloczłonowy może zostać podzielony i przeniesiony albo w obrębie wyrazów składowych, albo na łączniku. W tym ostatnim przypadku łącznik powinien zostać powielony, to znaczy znaleźć się zarówno na końcu pierwszego wiersza, jak też na początku drugiego. Oto możliwe miejsca podziału wyrazu niebiesko-czarny:

nie- biesko-czarny	niebie- sko-czarny	niebiesko- -czarny	niebiesko-czar- ny

Standardowy LaTeX nie zna polskich norm i dlatego dzieli wyrazy wieloczłonowe w miejscu połączenia, bez powielania łącznika. Jeżeli do składu w języku polskim korzystamy z pakietu platex to w pliku źródłowym w miejsce łącznika w wyrazach wieloczłonowych powinniśmy zastosować instrukcję `\dywiz` (np. `niebiesko{\dywiz}czarny`).

W pewnych sytuacjach lepiej nie dzielić wyrazów połączonych łącznikiem. Jeżeli na przykład mówimy o wydziale K-2, kodzie pocztowym czy numerze telefonu, to w takich przypadkach łącznik zapisujemy w pliku źródłowym jako pojedynczą kreskę „-”.

Znaku łącznika używa się również do przenoszenia wyrazów. Jednak w LaTeX-u odbywa się to automatycznie i nie wymaga ręcznych ingerencji użytkownika.

Myślnik „—” to dłuższa kreska używana jako znak przestankowy. Zapisujemy go za pomocą trzech następujących po sobie znaków „-”, czyli „`---`”. Przed oraz po myślniku w języku polskim dodajemy odstęp.

Półpauza to kreska o połowę krótsza od myślnika. Stosuje się ją przede wszystkim przy pisaniu zakresów liczbowych, np. „str. 11–13”, czy „w latach 1960–1963”. Przed i po półpauzie nie dodaje się odstępów. Odstępy takie muszą się pojawić w sytuacjach takich jak: „11 października – 13 listopada”. Czasami półpauzy używa się także w roli myślnika (tak jest w niniejszym tłumaczeniu). Półpauzę zapisuje się za pomocą dwóch następujących po sobie minusów „`--`”.

Znaku minusa używa się we wzorach matematycznych; aby go otrzymać piszemy „-” wewnątrz trybu matematycznego. Przykładowo zapis `$-2$` daje w składzie -2, podczas gdy `-2` daje -2.

[1]Porównaj uwagi dotyczące tego sposobu oznaczania cudzysłowów z punktu 2.4 w części dotyczącej fontów i ich kodowania (s. 22). Uwaga ta dotyczy również cudzysłowów francuskich.

2.3.3. Wielokropek (. . .)

W typowym piśmie maszynowym przecinek albo 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 miedzy nimi będą zbyt małe. Wielokropek (trzy kropki) uzyskujemy za pomocą instrukcji `\ldots`. Przykład:

`\ldots`

```
Nie tak ..., tylko raczej tak:\\
Nowy Jork, Tokio, Budapeszt, \ldots
```

Nie tak ..., tylko raczej tak:
Nowy Jork, Tokio, Budapeszt, . . .

2.3.4. Ligatury

Ligatury (spójki) to znaki graficzne, w których połączono dwie lub trzy litery. W niektórych językach ligatury występują jako właściwe danej ortografii znaki pisma, np. œ w języku francuskim. Większość ligatur tworzy się ze względów estetycznych lub zwyczajowych. LaTeX zna pięć następujących ligatur:

ﬀ ﬁ ﬂ ﬃ ﬄ zamiast ff fi fl ffi ffl

LaTeX automatycznie wstawia ligatury zamiast odpowiednich sekwencji znaków. Można temu zapobiec wstawiając instrukcję `\mbox{}` pomiędzy znaki tworzące ligaturę.

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

Jak lepiej: geograﬁi czy geografii?

2.3.5. Akcenty i znaki specjalne

LaTeX umożliwia wstawienie znaków akcentowanych i specjalnych używanych w wielu językach. W tabeli 2.1 zestawiono wszystkie możliwe akcenty dla litery „o”. Instrukcje te można stosować również do dowolnej innej litery.

W przypadku umieszczenia akcentu nad literami „i” oraz „j”, należy usunąć kropkę znad tych liter. Do tego celu służą instrukcje `\i` i `\j`, wstawiające do składu specjalne wersje liter „ı” oraz „ȷ”.

```
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

2.3.6. Niełamliwe odstępy

Polskie zasady typograficzne nie pozwalają przy łamaniu akapitu pozostawiać na końcu wierszy jednoliterowych spójników lub przyimków. Przykładowo w zdaniu „Jan Kochanowski urodził się w Czernolesie" nieładnie wyglądałby przyimek „w" występujący na końcu wiersza.

Odstępy, na których nie wolno złamać wiersza, zaznacza się w pliku źródłowym przez umieszczenie znaku tyldy „~" zamiast odstępu. Na przykład, w powyższej sytuacji powinniśmy zapisać w pliku źródłowym: „`w~Czarnolesie`".

Jest wiele sytuacji, w których związek fragmentów zdania jest tak silny, że nie należy wewnątrz nich łamać na wiersze. Nie zawsze decyzja jest tak prosta, jak w przypadku wspomnianych spójników. Oto garść przykładów:

```
godz.~17.00; od~15 do 40~osób; na str.~2 napisano; rozdz.~2;
2~rozdziały; p.~Jan Nowak; p.~J.~Nowak; I~część IX~Symfonii.
```

2.4. LaTeX wielojęzyczny

Jeżeli LaTeX ma składać tekst w językach innych niż angielski, to zasadniczo są dwa obszary, w których należy go odpowiednio skonfigurować:

1. Wszystkie teksty generowane automatycznie przez LaTeX-a trzeba przystosować do danego języka. Dotyczy to tytułów rozdziałów, spisu treści, spisu rysunków, tabel, dat itp. Zmiany te umożliwia pakiet babel Johannesa Braamsa.
2. LaTeX musi poznać reguły dzielenia wyrazów dla danego języka. „Powiedzenie" mu o nich jest swego rodzaju sztuczką, gdyż jest równoznaczne ze stworzeniem nowego formatu LaTeX-a z nowymi, wbudowanymi weń regułami. Więcej informacji na ten temat można znaleźć w *Local Guide* [5].

Tabela 2.1: Akcenty i znaki specjalne

ò	`\‘o`	ó	`\’o`	ô	`\^o`	õ	`\~o`
ō	`\=o`	ȯ	`\.o`	ö	`\"o`		
ŏ	`\u o`	ǒ	`\v o`	ő	`\H o`	o̧	`\c o`
ọ	`\d o`	o̲	`\b o`	o͡o	`\t oo`		
œ	`\oe`	Œ	`\OE`	æ	`\ae`	Æ	`\AE`
å	`\aa`	Å	`\AA`	ą	`\k{a}`		
ø	`\o`	Ø	`\O`	ł	`\l`	Ł	`\L`
ı	`\i`	ȷ	`\j`	¡	`!‘`	¿	`?‘`

Po odpowiednim skonfigurowaniu naszego systemu można wykorzystać pakiet babel. Wystarczy w tym celu użyć w preambule instrukcję:

`\usepackage[`*lista-języków*`]{babel}`

Argument *lista-języków*, to oddzielone przecinkami nazwy języków, które obsługuje lokalna wersja LaTeX-a. Ostatni na liście jest językiem domyślnym. Podręcznik *Local Guide* [5] zawiera informacje, o tym jakie nazwy mogą się pojawić na *liście-języków*.

Dla niektórych języków babel udostępnia specjalne instrukcje, ułatwiające wprowadzanie znaków diakrytycznych i specjalnych. Przykładowo teksty w języku niemieckim zawierają dużo liter z umlautami: (äöü). Wykorzystując pakiet babel można wprowadzić literę ö wpisując `"o` zamiast `\"o`.

W większości systemów komputerowych znaki akcentowane i specjalne (czyli znaki o kodach ASCII większych od 127) można wprowadzać bezpośrednio z klawiatury. Przykładowo polskie znaki diakrytyczne można wprowadzać naciskając klawisz prawy-Alt i jednocześnie klawisz z odpowiednią literą. LaTeX radzi sobie z takimi znakami. Począwszy od grudnia 1994 r. dystrybucja LaTeX-a zawiera pakiet inputenc, umożliwiający kodowanie znaków diakrytycznych zgodnie z wieloma wariantami, np. ISO-8859-2 (Linux) lub CP 1250 (MS Windows). Więcej informacji na ten temat można znaleźć w dokumentacji pakietu. Wykorzystując pakiet musimy sobie zdawać sprawę, że część użytkowników nie będzie w stanie prawidłowo wyświetlić tak przygotowanych plików źródłowych. Przykładowo znak ą ma numer 161 w standardzie ISO-8859-2, podczas gdy w kodowaniu zgodnym z CP 1250 ta sama litera ma kod 169 (a znak o kodzie 161 oznacza jakiś inny „krzaczek”).

Drugą stroną medalu jest układ znaków (kodowanie) fontu. Kodowanie fontu określa, w jakich miejscach *fontu* znajdują się poszczególne znaki. Oryginalne fonty Computer Modern (CM) są 128-znakowe i nie zawierają na przykład kompletu polskich znaków. Znaki diakrytyczne można tworzyć za pomocą połączenia dwóch znaków: litery i odpowiedniego akcentu. Taki sposób powoduje, że TeX nie może prawidłowo przenosić wyrazów zawierających znaki diakrytyczne, a także, że typograficzna jakość w ten sposób skonstruowanych diakrytyków jest kiepska.

Na szczęście prawie wszystkie współczesne dystrybucje TeX-a zawierają komplet fontów European Computer Modern (EC). Są to fonty ośmiobitowe zawierające 256 znaków. Pierwszych 128 znaków fontu EC są (prawie) identyczne ze znakami z fontów CM. Pozostałe 128 znaków to znaki diakrytyczne wykorzystywane w wielu językach europejskich, w tym komplet znaków niezbędnych do składania polskich tekstów. Fonty EC umożliwiają wykorzystanie pełnych możliwości TeX-owego mechanizmu przenoszenia wyrazów; znacznie lepsza jest też jakość typograficzna znaków diakrytycznych.

Ponadto istnieją jeszcze fonty PL (autorzy B. Jackowski i M. Rybko). Są to fonty zgodne z CM zawierające dodatkowo wszystkie polskie znaki diakrytyczne. Były to pierwsze dobrej jakości fonty typu CM dostępne publicznie. Polskie diakrytyki z fontów EC są kopią odpowiednich znaków z fontów PL. Dużą zaletą fontów PL jest to, że są dostępne także postaci fontów PostScriptowych.

W szczególności znaki cudzysłowów francuskich i polskiego otwierającego są dostępne w fontach EC i PL, ale nie ma ich w fontach CM. Opisany w punkcie 2.3.1 sposób wprowadzania tych znaków za pomocą par „,,", „<<" i „>>", działa tylko wówczas gdy używamy fontów EC lub PL.

Przełączenie na fonty EC następuje przez dołączenie pakietu fontenc w preambule dokumentu:

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

Argument `T1`, określający kodowanie, jest „odpowiedzialny" za przełączenie się na fonty EC.

2.4.1. Wsparcie dla języka polskiego

Poprawny skład w języku polskim wymaga trzech rzeczy: dodatkowego pakietu poleceń, fontów zawierających polskie znaki diakrytyczne oraz LaTeX-a, który zna polskie reguły przenoszenia wyrazów.

W każdej nowej, kompletnej dystrybucji LaTeX-a znajduje się wspomniany wyżej pakiet babel. Mimo pewnych niedociągnięć można go polecić szczególnie początkującym. Bardziej wymagający użytkownicy powinni skorzystać z opisanego dalej zestawu polonizacyjnego platex.

W pakiecie babel nazwy języków wykorzystywanych w dokumencie należy podać w preambule na liście opcji pakietu:

```
\usepackage[język1,język2]{babel}
```

Ostatni z wymienionych na liście języków jest językiem domyślnym dokumentu. Do włączenia języka służy polecenie `\selectlanguage`. Składnia tego polecenia jest następująca:

```
\selectlanguage{język}
```

Pakiet babel dla każdego języka definiuje odpowiednio elementy wpisywane automatycznie przez program (np. dla języka polskiego „Spis treści", a nie „Table of contents") oraz udostępnia polecenia ułatwiające przygotowanie dokumentów w tym języku.

Ta druga sprawa w przypadku języka polskiego sprowadza się w zasadzie do kodowania polskich znaków diakrytycznych. Można wyróżnić trzy sposoby kodowania: standardowe polecenia (opisane w punkcie 2.3.5), notację prefiksową (wyjaśnioną poniżej) i notację „bezpośrednią" (posługującą się znakami o kodach większych od 127).

Standardowe polecenia akcentowe LaTeX-a umożliwiają zapis wszystkich polskich znaków diakrytycznych w następującej postaci:

```
\k{a} \'c \k{e} \l{} \'n \'o \'s
\'z \.z \k{A} \'C \k{E} \L{} \'N
\'O \'S \'Z \.Z
```

ą ć ę ł ń ó ś ź ż Ą Ć Ę Ł Ń Ó Ś Ź Ż

Ponadto zapis np. `\'o` nie oznacza – jak w LaTeX-u 2.09 – „wstaw akcent *acute* nad literę o", tylko „użyj najlepszej metody uzyskania znaku *oacute* dostępnej w bieżącym układzie fontu". Jeżeli w bieżącym foncie jest dostępny znak *oacute*, zostanie on użyty bezpośrednio. Jeżeli go nie ma, to TeX złoży taki znak z litery o oraz odpowiedniego akcentu[2].

Posługiwanie się powyższymi poleceniami do pisania tekstów po polsku jest uciążliwe. Stąd pomysł notacji prefiksowej; polega ona na zapisywaniu polskich znaków diakrytycznych w postaci dwóch znaków: ustalonego znaku specjalnego (prefiksu) i litery łacińskiej podobnej graficznie do odpowiedniej polskiej. Notacja prefiksowa ma wiele wersji: stosowano różne znaki jako prefiks (co najmniej `"`, `@` i `/`) oraz różne konwencje zapisywania ź oraz ż (przynajmniej jeden z nich musi być zapisywany przy pomocy znaku różnego od `z`).

Pakiet babel umożliwia zapis prefiksowy z wykorzystaniem znaku cudzysłowu, oto przykład zapisu minuskuł polskich znaków diakrytycznych w tej notacji (`"r` oznacza ż):

```
"a "c "e "l "n "o "s "z "r
```

Ponadto „`"‘`" oznacza polski cudzysłów otwierający, „`"<`" francuski cudzysłów otwierający a „`">`" francuski cudzysłów zamykający.

Bezpośrednie wprowadzanie polskich znaków umożliwia pakiet inputenc opisany w poprzednim punkcie. W zależności od platformy, na której pracujemy, mamy do wyboru ISO-8859-2 (Unix, Linux), CP1250 (Windows) lub inne, rzadziej spotykane strony kodowe. Przykładowo, jeżeli dokument jest kodowany w standardzie ISO-8859-2 (Latin-2) pakiet inputenc należy dołączyć do dokumentu następująco:

```
\usepackage[latin2]{inputenc}
```

W przypadku, gdy dokument jest kodowany w standardzie CP 1250, zamiast opcji `latin2` powinniśmy wpisać `cp1250`.

Dodatkowo, w preambule dokumentu należy umieścić polecenie dołączenia pakietu fontenc z opcją `T1`, co spowoduje, że LaTeX będzie korzystał z 256 znakowych fontów EC. Fonty te zawierają dobrej jakości polskie znaki diakrytyczne.

Pozostał jeszcze do rozwiązania problem zapoznania LaTeX-a z polskimi regułami dzielenia wyrazów. Jeżeli posługujemy się LaTeX-em w środowisku wielodostępnym, na przykład na uczelni, to przypuszczalnie jest to już zrobione przez administratora, ale jeżeli rozpoczynamy pracę na własnym komputerze, to musimy niezbędną adaptację wykonać sami. W tym celu należy odszukać plik o nazwie `language.dat`. W pliku tym znajdują się informacje o dostępnych w pakiecie wzorcach dzielenia wyrazów. Interesujący nas fragment pliku wygląda mniej więcej tak:

```
%norsk nohyph.tex
%polish plhyph.tex
%portuges pthyph.tex
```

Pierwszy wyraz to nazwa języka (którą potem posługuje się LaTeX, np. jako argumentem polecenia `\selectlanguage`), drugi to *nazwa* pliku, w którym znajdują się reguły podziału wyrazów dla tego języka.

Jeśli chcemy dołączyć do LaTeX-a polskie wzorce przenoszenia, to powinniśmy usunąć znak procentu z linii zawierającej słowo `polish`. Na ogół LaTeX zna wzorce

[2] Z notacją tą wiąże się jednak dobrze znany problem: środowisko `tabbing` zmienia lokalnie definicję kilku makr, w tym `\'`. Dlatego w jego obrębie do uzyskiwania znaków z akcentem acute trzeba używać notacji typu `\a'o`. W konsekwencji zarówno w implementacji notacji prefiksowej, jak i „stron kodowych" pakietu inputenc do akcentu *acute* trzeba dostawać się nieco naokoło.

dzielenia dla maksimum 2–3 języków (przy większej liczbie języków spada szybkość kompilacji a także mogą wystąpić problemy z brakiem pamięci).

Po wprowadzeniu zmian do pliku `\language.dat` powinniśmy wbudować wzorce dla zaznaczonych w nim języków do LaTeX-a, co w praktyce oznacza, że trzeba wygenerować nowy *format* LaTeX. Niestety niezbędne do tego czynności zależą od konkretnej implementacji TeX-a. Omówimy tu, jak to wygląda w przypadku trzech popularnych dystrybucji: teTeX(Unix/Linux), web2c (Unix/Linux i Windows 9x/NT) oraz MikTeX (Windows 9x/NT).

Należy wykonać polecenie „`initex latex.ltx`". Jeżeli wszystko pójdzie dobrze, to w katalogu bieżącym pojawi się plik z formatem o nazwie `latex.fmt`. Następnie musimy odnaleźć w strukturze katalogów instalacji TeX-owej plik o tej samej nazwie i oryginalny zastąpić wygenerowanym przez nas. Otwarcie należy przyznać, że z wielu powodów utworzenie nowego pliku `latex.fmt` może się nie powieść. Jeśli zatem nie jesteśmy w stanie wybrnąć z kłopotów sami, powinniśmy poszukać pomocy u znajomego eksperta TeX-owego.

Podsumowując: mając prawidłowo skonfigurowany pakiet babel, łącznie z LaTeX-em znającym polskie reguły dzielenia wyrazów, wystarczy umieścić w preambule polecenia dołączające pakiety babel, fontenc oraz inputenc i ten sposób otrzymać całkiem dobrze spolonizowaną wersję LaTeX-a.

Zamiast pakietu babel można wykorzystać pakiet polski (wchodzący w skład zestawu polonizacyjnego platex). Został on opracowany w Polsce przez Mariusza Olko i Marcina Wolińskiego. Jego niewątpliwą zaletą jest staranniejsza polonizacja. Problemem – z punktu widzenia początkującego użytkownika – nieco trudniejsza instalacja. Dokumenty zapisane w notacji bezpośredniej nie wymagają modyfikacji, jeżeli zamiast pakietu babel w preambule dołączymy polski-a.

W porównaniu z pakietem babel pakiet platex jest bogatszy o kilka szczegółów polonizacyjnych:

- instrukcja `\dywiz`, opisana w punkcie 2.3.2, zapewnia poprawne przenoszenie wyrazów złożonych;
- instrukcje `\tg`, `\ctg`, `\tgh`, `\ctgh` oraz `\arc` wstawiają polskie symbole funkcji trygonometrycznych;
- opcjonalna zmiana znaczenia standardowych poleceń `\tan`, `\ctan` w taki sposób, że wstawiają one do składu poprawne nazwy funkcji trygonometrycznych (nazwy te są różne od nazw angielskich);
- polskie znaki większe-równe i mniejsze-równe[3].

Ponadto pakiet polski pozwala na wykorzystanie fontów PL oraz umożliwia posługiwanie się znakiem „/" (lub opcjonalnie innym znakiem) przy kodowaniu dokumentu za pomocą notacji prefiksowej, zwanej też – gdy używa się znaku „/" – notacją ciachową.

Pakiet polski dołączamy w preambule dokumentu poleceniem:

`\usepackage[`*opcje*`]{polski}`

Lista dostępnych opcji obejmuje:

`OT4` – wybranie kodowania `OT4` fontów, co w praktyce oznacza skład fontami PL;
`T1` – wybranie kodowania `T1` fontów, co oznacza skład fontami EC;

[3]Więcej informacji na temat poleceń pakietu platex związanych ze składem matematyki podajemy w rodziale 3.

- `OT1` – wybranie kodowania `OT1` fontów, co oznacza skład fontami CM (nie polecane);
- `plmath` – zmienia definicje znaków mniejszy-równy i większy-równy[4];
- `nomathsymbols` – blokuje zmianę znaczenia standardowych poleceń trygonometrycznych oraz relacji mniejszy-równy i większy-równy;
- `MeX` – tryb pełnej polonizacji, w praktyce skrót dla opcji `OT4` + `plmath`.

Jeżeli nie wybrano żadnej z opcji układu kodowania, to platex usiłuje włączyć fonty PL, jeżeli są one zainstalowane. Dotyczy to zarówno fontów tekstowych jak i tych zawierających znaki matematyczne. W instalacji zawierającej fonty PL dołączenie pakietu bez opcji jest równoważne:

```
\usepackage[OT4,plmath]{polski}
```

Polecenie `\selecthyphenation` pozwala przełączyć się na dany zestaw wzorców dzielenia wyrazów. Jest to odpowiednik polecenia `\selectlanguage` w pakiecie babel. Argumentem jest nazwa języka, taka jak w pliku `language.dat`.

Notację ciachową, która domyślnie jest *wyłączona*, można uaktywnić poleceniem `\prefixing`. Polecenie to powinno występować w preambule dokumentu dopiero po poleceniach dołączających pakiety.

Wprowadzanie polskich znaków w sposób bezpośredni wymaga dołączenia opisanego powyżej pakietu inputenc.

Instalowanie pakietu platex trzeba z reguły rozpocząć od jego skopiowania w odpowiednie miejsce do naszej instalacji LaTeX-owej ponieważ w przeciwieństwie do pakietu babel nie jest on dołączany do standardowej dystrybucji LaTeX-a (być może to się zmieni w przyszłości). Pakiet znajduje się w `CTAN://tex-archive/language/polish/platex-1.01.zip` lub `ftp://ftp.gust.org.pl/TeX/language/polish/platex-1.01.zip`. Miejsce, w które należy skopiować pakiet, zależy od instalacji. Dla systemów teTeX, web2c i MikTeX będzie to `.../texmf/tex/latex/`. Trzy kropki z przodu wskazują, że katalog `texmf/` nie musi się znajdować w katalogu głównym, tylko raczej w jakimś miejscu położonym „niżej” w strukturze katalogów. Po znalezieniu właściwego katalogu `texmf/tex/latex/` tworzymy tam podkatalog, `platex`, i rozpakowujemy w nim zawartość pakietu. Następnie wydajemy polecenie:

```
latex platex.ins
```

Do uzyskania poprawnego przenoszenia polskich słów konieczne jest wygenerowanie nowego formatu LaTeX-a. W tym celu należy usunąć (lub zmienić nazwy, tak aby zostały pominięte przez LaTeX-a) oryginalne pliki `hyphen.cfg`, `language.dat`. Ponadto w wielu starszych instalacjach plik `hyphen.tex` jest podmieniony „pod” potrzeby babel-a i nie zawiera angielskich wzorców podziału. W tym przypadku trzeba zastąpić plik `hyphen.tex` „prawdziwym” plikiem z angielskimi regułami dzielenia. Następnie z linii poleceń piszemy:

```
initex latex.ltx
```

Z tak powstałym plikiem `latex.fmt` postępujemy tak samo, jak przy tworzeniu nowego LaTeX-a dla pakietu babel.

---

[4] W systemie muszą być zainstalowane fonty PL. Są one dostępne w `CTAN://tex-archive/language/polish/pl-mf.zip`. Opcjonalnie można także skopiować pliki metryczne tych fontów. Są one dostępne w `CTAN://tex-archive/language/polish/pl-tfm.zip`. Pliki metryczne fontów, w przypadku wykorzystywania nowych instalacji TeX-a typu teTeX, web2c czy MikTeX nie są konieczne, gdyż jeżeli ich nie ma, są tworzone automatycznie. Nie będziemy omawiać szczegółowo instalacji fontów PL, gdyż sposób jej przeprowadzenia zależy mocno od konkretnej dystrybucji.

2.5. Odstępy między wyrazami

Aby wyrównać prawy margines, LaTeX wstawia pomiędzy słowa 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 duża litera, to LaTeX nie traktuje tego miejsca jako końca zdania, lecz jako kropkę po skrócie.

Wyjątki od wymienionych wyżej zasad muszą być jasno zaznaczone w tekście. W-tył-ciach „\" poprzedzający spację oznacza odstęp normalnej wielkości. Tylda `~` wstawia normalny odstęp, na którym LaTeX nie może złamać wiersza. Instrukcja `\@` wstawia większy odstęp, taki jakim kończy się zdania.

```
Mr.~Smith was happy to see her\\
cf.~Fig.~5\\
I like BASIC\@. What about you?
```

Mr. Smith was happy to see her
cf. Fig. 5
I like BASIC. What about you?

Jak wspomniano, wstawianie większych odstępów na końcu zdań to zwyczaj anglosaski. W kontynentalnej Europie tradycyjnie się tego nie robi. Wstawianie większych odstępów na końcu zdań można (należy) wyłączyć, wykonując instrukcję:

```
\frenchspacing
```

Pakiet polski domyślnie wykonuje instrukcję `\frenchspacing` za nas – wystarczy go dołączyć.

2.6. Tytuły, śródtytuły i punkty

Podzielenie dokumentu na rozdziały, podrozdziały, punkty itd. pomaga czytelnikom lepiej orientować się w tekście. Do dzielenia dokumentu na hierarchiczne części służą odpowiednie polecenia LaTeX-owe. Do autora należy wykorzystanie tych poleceń w odpowiednim porządku.

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

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

W klasie `report` i `book` można użyć dwóch dodatkowych instrukcji:

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

Ponieważ w klasie `article` najwyższą jednostką w hierarchii podziału jest `\section` (czyli *punkt*), 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 z wymienionych instrukcji działają nieco inaczej niż pozostałe:

- Instrukcja `\part` nie ma wpływu na numerację rozdziałów.
- Instrukcja `\appendix` nie posiada argumentów. Jest to deklaracja zmieniająca z cyfr na litery sposób numerowania rozdziałów (w klasach `book` i `report`) lub punktów (w klasie `article`).

Argumentu instrukcji podziału dokumentu LaTeX używa do przygotowania spisu treści. Instrukcja:

```
\tableofcontents
```

wstawia spis treści w miejscu, w którym jej użyjemy. Aby w spisie treści otrzymać prawidłowe numery stron, dokument trzeba przetworzyć („zlatechować”) dwukrotnie. Czasami niezbędna jest nawet trzecia kompilacja. Kiedy pod koniec przetwarzania dokumentu LaTeX wyświetli komunikat:

```
LaTeX Warning: Label(s) may have changed.
Rerun to get cross-references right.
```

oznacza to, że niezbędna jest kolejna kompilacja.

LaTeX przetwarza dokument strona po stronie i w pojedynczym przebiegu nie może wstawić spisu treści na początku dokumentu, ponieważ nie jest jeszcze znana jego treść ani numeracja stron. Podobnie ma się sprawa ze spisami tabel czy rysunków. Rozwiązanie tego problemu jest tyle proste, co skuteczne. Podczas przetwarzania dokumentu LaTeX zapisuje odpowiednie informacje do plików pomocniczych. Przeznaczenie danego pliku pomocniczego jest zasygnalizowane w rozszerzeniu nazwy. I tak: plik o rozszerzeniu `.toc` zawiera spis treści, plik `.lot` – spis tabel, plik `.lof` – spis rysunków, plik `.aux` – informacje o odsyłaczach wewnątrz dokumentu (odsyłacze omawiamy w podrozdziale 2.7). Informacje wprowadzone do plików pomocniczych LaTeX wykorzystuje podczas kolejnych przebiegów.

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 podrozdziału zostanie umieszczony w dokumencie, ale nie w spisie treści; tytuł nie jest też objęty numeracją. Przykładowo wersją „z gwiazdką” instrukcji `\section{Help}` jest `\section*{Help}`.

Najczęściej hasła w spisie treści są identyczne z odpowiadającymi im tytułami rozdziałów czy punktów. Czasami jednak nie jest to pożądane, ponieważ na przykład tekst hasła jest zbyt długi. W takich wypadkach hasło do spisu treści można podać jako *opcjonalny* argument instrukcji podziału hierarchicznego, np:

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

W wyniku wykonania tej instrukcji w spisie treści pojawi się „Krótki i ekscytujący rozdział”, natomiast w tytule rozdziału „To jest bardzo długi i wyjątkowo nudny rozdział”.

LaTeX składa stronę tytułową w wyniku wykonania instrukcji:

`\maketitle`

która, powinna być umieszczona po `\begin{document}`.

Zawartość strony tytułowej ustalają polecenia:

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

Należy je umieścić w preambule. Jeżeli dokument ma kilku autorów, to ich nazwiska i imiona rozdzielamy instrukcją `\and`.

Rysunek 1.3 (s. 8) ilustruje sposób wykorzystania powyższych instrukcji.

W LaTeX 2_ε istnieją ponadto trzy instrukcje określające strukturę dokumentu, dostępne wyłącznie w klasie `book`, a mianowicie:

`\frontmatter`, `\mainmatter` oraz `\backmatter`

Są one użyteczne do podziału książki na część wstępną (obejmującą tytulaturę, spisy treści, tabel itp., wstępy itd.), główną i zakończenie (załączniki, skorowidze, kolofon itd). W części wstępnej tytuły rozdziałów są składane mniejszym stopniem pisma (czego należy oczekiwać), a numery stron w notacji rzymskiej (co raczej odbiega od polskich zwyczajów typograficznych).

2.7. Odsyłacze

Książki, raporty i artykuły zawierają często odniesienia do rysunków, tabel i innych fragmentów tekstu. Do wstawiania odsyłaczy LaTeX udostępnia następujące trzy instrukcje:

`\label{`*etykieta*`}`, `\ref{`*etykieta*`}` i `\pageref{`*etykieta*`}`

Argument *etykieta* jest ciągiem liter, cyfr lub znaków interpunkcyjnych. Nazwy etykiet ustala sam użytkownik. LaTeX zamienia `\ref{`*etykieta*`}` na numer tego rozdziału, punktu, rysunku, tabeli czy też równania matematycznego, *bezpośrednio za* którym umieszczona została instrukcja `\label` zawierająca identyczną *etykietę*. Instrukcja `\pageref{`*etykieta*`}` działa identycznie jak `\ref`, z tym że wstawia numer strony, na której znajduje się element oznaczony etykietą[5].

[5]Zwróćmy uwagę, że te instrukcje „nie wiedzą”, do czego tak naprawdę się odnoszą. Zadaniem instrukcji `\label` jest przechowanie wygenerowanej automatycznie liczby.

Podobnie jak w przypadku spisów treści, tabel czy rysunków potrzebne są co najmniej dwie, a z reguły trzy, kompilacje dokumentu do prawidłowego wstawienia numerów odsyłaczy. Podczas pierwszej LaTeX wysyła do pliku pomocniczego (z rozszerzeniem `.aux`) informacje o odsyłaczach, które wykorzystuje podczas kolejnych 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 2.7 na stronie 30.”

2.8. Przypisy

Do składania przypisów u dołu strony służy instrukcja:

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

Instrukcję należy wstawić bezpośrednio po słowie lub zdaniu, do którego się odnosi[6].

LaTeX numeruje przypisy automatycznie. Sposób ich numerowania zależy od wykorzystywanej klasy. W klasie `article` numeracja jest ciągła, a w klasach `report` i `book` przypisy są numerowane w ramach rozdziałów.

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

Przypisy[a] są często wykorzystywane przez użytkowników LaTeX-a

[a]To jest właśnie przypis.

2.9. Wyróżnienia

W tekstach pisanych na maszynie fragmenty tekstu, które mają zostać wyróżnione, podkreśla się. W dokumentach drukowanych wyróżnianie fragmentów odbywa się przez składanie ich *kursywą*. Służy do tego instrukcja:

`\emph{`*text*`}`

Argumentem tej instrukcji jest tekst, który ma zostać wyróżniony.

```
\emph{\emph{Wyróżnienia} w~tekście
już wyróżnionym są składane
\emph{zwykłym} krojem pisma.}
```

Wyróżnienia *w tekście już wyróżnionym są składane* zwykłym *krojem pisma.*

[6]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 (zasadę tę stosujemy w niniejszym wprowadzeniu).

Zwróćmy uwagę, że istnieje różnica między wyróżnieniem części tekstu a złożeniem go inną czcionką:

```
\textit{Można \emph{wyróżnić}
tekst, składając go kursywą,}
\textsf{czcionką \emph{szeryfową},}
\texttt{a nawet \emph{maszynowo}.}
```

Można wyróżnić *tekst, składając go kursywą,* czcionką *szeryfową*, a nawet *maszynowo*.

2.10. Środowiska

Wiele instrukcji to *środowiska*, mające postać:

`\begin{`*nazwa*`}` *tekst* `\end{`*nazwa*`}`

gdzie *nazwa* jest nazwą środowiska. Środowiska można zagnieżdżać jedne w drugich:

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

Niedopuszczalnie jest natomiast „przeplatanie" środowisk:

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

Wiele poleceń można zapisać w formie „środowiskowej". W takich wypadkach nazwa polecenia (bez w tył-ciacha) jest nazwą środowiska.

Przykładowo zamiast polecenia `\em`, które służy do włączenia wyróżnionego kroju pisma, możemy wykorzystać środowisko `\begin{em}...\end{em}`.

W kolejnych punktach przedstawiamy częściej wykorzystywane środowiska.

2.10.1. Środowiska `itemize`, `enumerate` i `description`

Środowiska `itemize` oraz `description` służą do tworzenia wyszczególnień, zaś `enumerate` do tworzenia wyliczeń. W każdym z nich element wyliczenia zaczyna się instrukcją `\item`.

```
\begin{enumerate}
\item Taka lista:
\begin{itemize}
\item wygląda
\item[--] śmiesznie.
\end{itemize}
\item Pamiętaj:
\begin{description}
\item[Głupoty] nie stają się
mądrościami, gdy się je wyliczy.
\item[Mądrości] można elegancko
zestawiać w~wyliczeniach.
\end{description}
\end{enumerate}
```

1. Taka lista:
   - wygląda
   – śmiesznie.
2. Pamiętaj:

   Głupoty nie stają się mądrościami, gdy się je wyliczy.

   Mądrości można elegancko zestawiać w wyliczeniach.

2.10.2. Środowiska flushleft, flushright i center

Środowiska `flushleft` i `flushright` składają akapity wyrównane, odpowiednio, do lewego lub prawego marginesu. Wewnątrz środowiska `center` każdy wiersz akapitu jest wyśrodkowany w osi szpalty. Tak jak zawsze, LaTeX dzieli akapity na wiersze automatycznie, w obrębie powyższych środowisk można jednak poleceniem `\\` wymusić zmianę wiersza.

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

To jest tekst
wyrównany do lewej. LaTeX nie składa tu
wierszy o 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.

2.10.3. Środowiska quote, quotation i verse

Środowisko `quote` nadaje się do składania cytatów oraz przykładów.

```
Jeżeli chodzi o~długości wiersza,
regułą kciuka jest, że:
\begin{quote}
Przeciętnie wiersz nie powinien
zawierać więcej niż 66 znaków.

Dlatego w~\LaTeX-u standardowe
strony mają szerokie marginesy.
\end{quote}
Dlatego też w~gazetach stosuje
się druk wieloszpaltowy.
```

Jeżeli chodzi o długości wiersza, regułą kciuka jest, że:

> Przeciętnie wiersz nie powinien zawierać więcej niż 66 znaków.
> Dlatego w LaTeX-u standardowe strony mają szerokie marginesy.

Dlatego też w gazetach stosuje się druk wieloszpaltowy.

Istnieją ponadto dwa środowiska o podobnym zastosowaniu: `quotation` oraz `verse`. Środowisko `quotation` jest przydatne do formatowania cytatów dłuższych niż jeden akapit. W przeciwieństwie do środowiska `quote`, wewnątrz środowiska `quotation` LaTeX rozpoczyna poszczególne akapity od wcięcia akapitowego. Środowisko `verse` służy do składania wierszy. Poszczególne linijki w zwrotkach należy kończyć instrukcją `\\` zaś poszczególne zwrotki oddzielać pustą linią.

```
Na pamięć znam tylko jeden angielski
wiersz. Ten o~Humptym Dumptym.
\begin{flushleft}
\begin{verse}
Humpty Dumpty sat on a wall:\\
Humpty Dumpty had a great fall.\\
All the King's horses and all
the King's men\\
Couldn't put Humpty together again.
\end{verse}
\end{flushleft}
```

Na pamięć znam tylko jeden angielski wiersz. Ten o Humptym Dumptym.

> Humpty Dumpty sat on a wall:
> Humpty Dumpty had a great
>   fall.
> All the King's horses and all the
>   King's men
> Couldn't put Humpty together
>   again.

2.10.4. Symulacja maszynopisu

Tekst pomiędzy `\begin{verbatim}` oraz `\end{verbatim}` LaTeX składa dosłownie, tak aby wyglądał jak napisany na maszynie, z zachowaniem zmian wiersza i odstępów.

Aby uzyskać ten efekt krój pisma zmienia się na imitujący pismo maszynowe (grotesk). Każdy znak w tym kroju, włączając w to spację, jest jednakowej szerokości. Koniec linii wewnątrz środowiska `verbatim` powoduje rozpoczęcie nowego wiersza na wydruku a *każda* spacja zamienia się na odstęp. Wewnątrz środowiska `verbatim` instrukcje nie są wykonywane.

Wewnątrz akapitów imitację maszynopisu uzyskuje się za pomocą instrukcji:

`\verb+`*tekst*`+`

Znak `+` ogranicza tu tekst, który ma zostać wydrukowany dosłownie. Zamiast `+` można użyć innego znaku, byle to nie była litera, gwiazdka ani spacja. Instrukcję `\verb` wykorzystujemy często w tej książce do składania przykładów LaTeX-owych.

```
Rozważmy przykład\ldots

\begin{verbatim}
{ for (i=1;i<=NF;i++) {l[$i]++; }
END {for (i in l) {print l[i]}
\end{verbatim}
```

Rozważmy przykład . . .

```
{ for (i=1;i<=NF;i++) {l[$i]++; }
END {for (i in l) {print l[i]}
```

Środowiska `verbatim` oraz instrukcja `\verb` mają także wersje „z gwiazdką”, w których spacja w pliku źródłowym jest zamieniana na znak ␣. Jest to jedyna różnica w działaniu w porównaniu do wersji bezgwiazdkowych:

```
\begin{verbatim*}
gwiazdkowa wersja
środowiska verbatim
wyróżnia   spacje
w tekście
\end{verbatim*}
```

```
gwiazdkowa␣wersja
środowiska␣verbatim
wyróżnia␣␣␣spacje
w␣tekście
```

Środowiska `verbatim` oraz instrukcji `\verb` nie wolno używać wewnątrz argumentów innych instrukcji.

2.10.5. Środowisko `tabular`

Do składania tabel służy środowisko `tabular`. LaTeX automatycznie ustala w nich szerokość poszczególnych rubryk. Środowisko ma jeden parametr obowiązkowy, określający liczbę kolumn tabeli i sposób ich justowania:

`\begin{tabular}{`*spec-kolumn*`}`

Dla każdej kolumny należy w argumencie wstawić jedną z liter: `l`, `r` lub `c`, określając w ten sposób justowanie zawartości kolumny. Dosunięcie zawartości kolumny do lewej oznaczamy literą `l`, do prawej – znakiem `r`, a wyśrodkowanie – znakiem `c`. Zapisu `p{`*szer-kolumn*`}` można użyć do zaznaczenia, że kolumna ma mieć szerokość *szer-kolumn*. Wewnątrz takiej kolumny tekst jest składany w prostokąt o zadanej szerokości, z wyrównywaniem obu marginesów. Znak `|` oznacza, że kolumny mają zostać rozdzielone pionową kreską.

Wewnątrz środowiska `tabular` poszczególne wiersze oddzielamy instrukcją `\\`, a rubryki w wierszu znakiem `&`. Instrukcja `\hline` wstawia poziomą linię na całą szerokość tabeli.

```
\begin{tabular}{|r|l|} \hline
7C0 & heksadecymalnie \\
3700 & oktalnie \\
11111000000 & binarnie \\
\hline \hline
1984 & dziesiętnie \\ \hline
\end{tabular}
```

7C0	heksadecymalnie
3700	oktalnie
11111000000	binarnie
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. Instrukcja ta usuwa domyślny odstęp międzykolumnowy, zastępując go tym, jaki wyspecyfikowano wewnątrz `{...}`. Stosuje się ją często do wyrównywania zestawień liczbowych według cyfr znaczących. Można ją także wykorzystać do usunięcia odstępów w pierwszej i ostatniej kolumnie tabeli, co ilustruje poniższy przykład.

```
\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[7], ale efekt ten można osiągnąć, składając liczbę w dwóch kolumnach: część całkowitą w kolumnie wyrównywanej do prawego brzegu oraz część dziesiętną w kolumnie wyrównanej do lewego. Za pomocą instrukcji `@{,}` zastępujemy przecinkiem odstęp normalnie wstawiany 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 posługując się poleceniem `\multicolumn`.

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

Wyrażenie	Wartość
π	3,1416
π^π	36,46
$(\pi^\pi)^\pi$	80662,7

2.11. Wstawki

Współczesne publikacje zawierają dużo rysunków i tabel. Elementów tych nie należy dzielić między strony i dlatego wymagają specjalnego potraktowania. W sytuacjach, gdy nie mieszczą się one na bieżącej stronie, na ogół przenosi się je i wstawia na początku strony następnej. W przypadku przeniesienia rysunku lub tablicy, pozostałe miejsce na stronie wypełnia się tekstem. Tego typu elementy będziemy nazywać *wstawkami*.

Aby w pełni skorzystać z LaTeX-owego mechanizmu wstawek, trzeba chociażby powierzchownie zrozumieć, jak LaTeX manipuluje takimi obiektami. W przeciwnym wypadku wstawki mogą się stać źródłem irytacji, ponieważ LaTeX będzie je umieszczał wszędzie, tylko nie w miejscach, w którym byśmy sobie tego życzyli.

[7] Do wyrównywania cyfr można skorzystać z możliwości pakietu dcolumn z zestawu pakietów „tools”.

Tabela 2.2: Opcjonalny argument środowisk `table` i `figure`

Znak	Dopuszczalne miejsce umieszczenia wstawki
`h`	bez przemieszczenia, dokładnie w miejscu użycia (użyteczne w odniesieniu do niewielkich wstawek);
`t`	na górze strony;
`b`	na dole strony;
`p`	na stronie zawierającej wyłącznie wstawki;
`!`	ignorując większość parametrów kontrolujących umieszczanie wstawek[a], przekroczenie wartości, których może nie pozwolić na umieszczanie następnych wstawek na stronie.

[a]Są to takie parametry, jak na przykład maksymalna dopuszczalna liczba wstawek na stronie.

Do tworzenia wstawek mamy w LaTeX-u dwa środowiska. Środowisko `figure` służy do tworzenia rysunków, a środowisko `table` – do tablic. Oba mają jeden parametr opcjonalny:

`\begin{figure}[`*miejsce*`]` albo `\begin{table}[`*miejsce*`]`

Argument *miejsce* określa, gdzie na stronie można umieścić wstawkę. Jest to sekwencja od jednego do pięciu znaków: `h`, `t`, `b`, `p` oraz `!`. Każdy znak określa dopuszczalny sposób umieszczenia wstawki; szczegółowe informacje zestawiono w tabeli 2.2.

Przykładowa tabela może zaczynać się tak:

```
\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 wstawienie „`!`” oznacza, że LaTeX ma pominąć większość parametrów kontrolujących umieszczanie wstawek. Jeżeli instrukcji `\table` użyto bez opcjonalnego argumentu, to jego domyślnymi wartościami są `[tbp]`.

LaTeX umieszcza każdą wstawkę zgodnie ze specyfikacją autora. Jeżeli wstawki nie można umieścić na bieżącej stronie, to dołącza się ją albo do *kolejki rysunków*, albo do *kolejki tabel*[8]. Po rozpoczęciu składania nowej strony LaTeX sprawdza, czy można zapełnić całą stronę wstawkami czekającymi w kolejce; gdy nie jest to możliwe, pierwszą wstawkę z każdej kolejki traktuje w taki sposób, jak gdyby właśnie pojawiła się w tekście: LaTeX stara się ją umieścić zgodnie z wartościami parametru *miejsce* (za wyjątkiem `h`, gdyż nie jest to już oczywiście możliwe). Nowe wstawki są odpowiednio dołączane do kolejek. LaTeX dba o właściwy porządek wstawek każdego typu.

[8]Są to kolejki typu FIFO (pierwsze weszło – pierwsze wyszło).

W konsekwencji pojedynczy rysunek, którego z jakiś względów nie można poprawnie wstawić „ciągnie” za sobą wszystkie inne rysunki aż na koniec dokumentu. Dlatego:

> Jeżeli LaTeX nie umieszcza wstawek zgodnie z oczekiwaniami, z reguły jest to spowodowane jedną wstawką blokującą kolejkę, a być może nawet wszystkie kolejki wstawek.

Wyjaśniwszy trochę złożony problem umieszczania wstawek, przejdźmy do omówienia kilku pozostałych rzeczy z nimi związanych.

Za pomocą instrukcji:

`\caption{`*tekst*`}`

wstawiamy tytuł rysunku lub tabeli. Kolejny numer rysunku/tabeli oraz słowo „Rysunek” i „Tabela” (lub „Tablica” – w zależności od tego, z jakiego pakietu polonizacyjnego korzystamy) zostaną wstawione automatycznie.

Następujące instrukcje:

`\listoffigures` oraz `\listoftables`

działają w sposób analogiczny do instrukcji `\tableofcontents`, wstawiając do dokumentu odpowiednio spis rysunków oraz spis tabel. Poszczególnymi pozycjami w tych spisach będą tytuły rysunków bądź tabel, podane jako argumenty instrukcji `\caption`. Jeżeli tytuły są długie, to do spisu można przesłać ich wersje skrócone, umieszczając je jako opcjonalny argument instrukcji `\caption`:

```
\caption[Short]{LLLLLoooooonnnnnggggg}
```

Odsyłacze do tabel i rysunków należy tworzyć za pomocą instrukcji `\label` oraz `\ref`.

Polecenie `\label` należy umieszczać bezpośrednio *za* instrukcją `\caption`. Dobrym pomysłem jest też umieszczenie jej wewnątrz argumentu instrukcji `\caption` (na przykład na końcu tytułu rysunku czy tabeli). Niektórzy użytkownicy błędnie sądzą, że wystarczy umieścić instrukcję `label` wewnątrz środowiska `figure` czy `table`, gdy tymczasem umieszczenie jej przed poleceniem `\caption` spowoduje błędy w numerach odsyłaczy.

W kolejnym przykładzie wstawka zawiera prostokąt o wymiarach 5 cm $\times$ 5 cm. Ten sposób postępowania można wykorzystać w celu zarezerwowania miejsca na rysunki, które zostaną wklejone później – do gotowego, wydrukowanego dokumentu.

```
Rysunek~\ref{white} jest przykładem Pop-Artu.
\begin{figure}[!hbp]
\makebox[\textwidth]{\framebox[5cm]{\rule{0pt}{5cm}}}
\caption{Pięć na pięć centymetrów\label{white} }
\end{figure}
```

W powyższym przykładzie[9] LaTeX najpierw spróbuje umieścić rysunek bez przesuwania go dokądkolwiek (`h`). Jeżeli okaże się to niemożliwe, spróbuje umieścić go na dole strony (`b`). Jeżeli i to okaże się niewykonalne, będzie się starał umieścić rysunek na stronie zawierającej wyłącznie wstawki (`p`). Jeżeli w kolejkach rysunków i tabel nie ma wstawek pozwalających wypełnić stronę LaTeX rozpocznie nową stronę i spróbuje umieścić na niej rysunek traktując go znowu tak, jakby właśnie pojawił się w tekście.

Czasami może okazać się niezbędne wykonanie instrukcji:

`\clearpage` albo nawet `\cleardoublepage`

W wyniku jej zadziałania LaTeX umieszcza w dokumencie wszystkie oczekujące w kolejkach wstawki, a następnie rozpoczyna skład od nowej strony. W przypadku użycia instrukcji `\cleardoublepage` LaTeX rozpoczyna od stronie nieparzystej.

W dalszej części tego wprowadzenia przedstawimy, jak można dołączać do LaTeX-owego dokumentu rysunki w formacie PostScript (por. 4.1).

2.12. Listy

Do pisania listów można wykorzystać klasę `letter`. Struktura pliku źródłowego w przypadku listów oczywiście różni się od dokumentów z klasy `article` czy `book`. Klasę `letter` zaprojektowano tak, że bezproblemowo możemy napisać pojedynczy list do przysłowiowej „cioci”, jak i setki listów do różnych osób (korespondencja seryjna).

Jeśli imię, nazwisko i adres nadawcy mają być identyczne we wszystkich listach, to deklarujemy je za pomocą polecenia (instrukcja `\\` oznacza złamanie wiersza):

`\address{`*imię*`\\`*nazwisko*`\\` *adres* … `}`

Polecenie `\signature` służy do zadeklarowania podpisu pod listem. Wewnątrz argumentu polecenia `\signature` instrukcja `\\` służy do rozpoczęcia nowego wiersza, przykładowo:

```
Dyr. E.~K.~Tor,\\ Przewodniczący
Zastępcy
```

Dyr. E. K. Tor,
Przewodniczący Zastępcy

Data jest wstawiana automatycznie. Jeżeli chcemy wstawić inną, należy zastosować deklarację `\date`:

```
\date{16 Czerwca 1963~r.}
```

Deklaracje `\address`, `\signature` oraz `\date` zwykle umieszcza się w preambule, ale mogą także wystąpić w części zasadniczej pliku źródłowego.

Treść listu powinniśmy wpisać wewnątrz środowiska `letter`. Środowisko to ma jeden argument, którym jest adres osoby, do której piszemy. Wewnątrz środowiska `letter` można skorzystać z kilku prostych poleceń służących do umieszczania w odpowiednim

[9] Zakładając, że kolejka rysunków jest pusta.

miejscu elementów typowego listu[10]. Do złożenia nagłówka listu wykorzystujemy polecenie \opening, a do zakończenia listu – polecenie \closing. Ponadto są jeszcze polecenia \ps, do wstawienia *post scriptum*, oraz \cc, które służy do zdefiniowania wykazu osób, które dostaną kopię listu.

Oto pełny przykład listu:

```
\documentclass{letter}
\usepackage{polski}
\usepackage[latin2]{inputenc}
\address{Dyr. E.~K.~Tor,\\ Przewodniczący Zastępcy\\
 Firma z~o.o.\\ w/m}
\signature{E.~.K.~Thor}
\begin{document}
% pierwszy list
\begin{letter}{Henryk Potrykus\\ul.~Krótka\\Puck}
\opening{Szanowny Panie}
Z~przykrością zawiadamiam, że pańskie podanie
zostało...
\closing{Z~poważaniem}
\cc{cc: Józef Wujke}
\end{letter}

% drugi list
\begin{letter}{Zofia Potrykus\\ul.~Szkolna\\Reda}
\opening{Szanowa Pani}
W~odpowiedzi na pani pismo...
...
\end{letter}
\end{document}
```

[10]Elementy nietypowe zawsze można umieścić, korzystając z innych poleceń poznanych w tym rozdziale.

Rozdział 3

Wyrażenia matematyczne

Nareszcie! W tym rozdziale poznasz najlepszą stronę TEX-a, czyli skład wzorów matematycznych. Ostrzegamy jednak, że przedstawimy tu tylko absolutne podstawy. Chociaż większości użytkowników one wystarczają, nie załamuj rąk, jeżeli nie poradzisz sobie z jakimś skomplikowanym wzorem, tylko raczej zapoznaj się z możliwościami pakietu AMS-LATEX[1] lub innego specjalizowanego pakietu.

3.1. Wstęp

Do składu wyrażeń matematycznych mamy w LATEX-u specjalny *tryb matematyczny*. Oznacza to wpisywanie wzorów pomiędzy `\(` i `\)` albo `$` i `$`, albo pomiędzy `\begin{math}` oraz `\end{math}`.

```
$a$ 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
$\tau\epsilon\chi$.\\[6pt]
100~m$^{3}$ wody \\[6pt]
To płynie z~mojego~$\heartsuit$
```

TEX należy wymawiać jako $\tau\epsilon\chi$.

100 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 albo pomiędzy `\[` i `\]`, albo wewnątrz środowiska `displaymath`. Ta ostatnia konstrukcja dotyczy tworzenia wzorów bez żadnej numeracji. Do automatycznego numerowania wzorów stosujemy natomiast środowisko `equation`.

[1] `CTAN://tex-archive/macros/latex/packages/amslatex`.

```
$a$ do kwadratu plus~$b$ do
kwadratu równa się~$c$ do kwadratu.
Albo, bardziej matematycznie,
\begin{displaymath}
c^{2}=a^{2}+b^{2}
\end{displaymath}
Pierwszy wiersz w~drugim akapicie.
```

a do kwadratu plus b do kwadratu równa się c do kwadratu. Albo, bardziej matematycznie,

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

Pierwszy wiersz w drugim akapicie.

Instrukcje `\label` oraz `\ref` służą do tworzenia odsyłaczy do równań.

```
\begin{equation}
\epsilon > 0 \label{eq:eps}
\end{equation}
Ze wzoru (\ref{eq:eps})
otrzymujemy \ldots
```

$$\epsilon > 0 \tag{3.1}$$

Ze wzoru (3.1) otrzymujemy ...

Zwróćmy uwagę, że inaczej składa się wzory wewnątrz akapitu, a inaczej eksponowane:

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

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

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

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

Między *trybem matematycznym* LaTeX-a a *trybem tekstowym* istnieją duże różnice. Na przykład w *trybie matematycznym*:

1. LaTeX ignoruje prawie wszystkie odstępy oraz znaki końca linii; wszystkie odstępy we wzorach wynikają albo z kontekstu albo z użycia specjalnych poleceń, takich jak `\,`, `\quad` lub `\qquad`.
2. Puste linie są niedozwolone. Obowiązuje zasada: jeden wzór – jeden akapit.
3. Litery we wzorach służą do oznaczania nazw zmiennych; zmienne składamy inaczej niż zwykły tekst. Jeżeli częścią wzoru jest zwykły tekst, to należy posłużyć się instrukcją `\textrm{...}`.

```
\begin{equation}
\forall x \in \mathbf{R}\colon
\qquad x^{2} \geq 0
\end{equation}
```

$$\forall x \in \mathbf{R}\colon \qquad x^{2} \geqslant 0 \tag{3.2}$$

```
\begin{equation}
x^{2} \geq 0\qquad
\textrm{dla każdego }x\in\mathbf{R}
\end{equation}
```

$$x^{2} \geqslant 0 \qquad \textrm{dla każdego } x \in \mathbf{R} \tag{3.3}$$

Matematycy potrafią być niezwykle staranni w doborze odpowiednich symboli. 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 ($\mathbb{A}, \mathbb{B}, \mathbb{C} \ldots$). Symbole takie wstawiamy do wzoru poleceniem `\mathbb` z pakietu amsfonts lub amssymb. Ostatni przykład wygląda wtedy następująco:

```
\begin{displaymath}
x^{2} \geq 0\qquad
\textrm{dla każdego }x\in\mathbb{R}
\end{displaymath}
```

$$x^2 \geqslant 0 \qquad \textrm{dla każdego } x \in \mathbb{R}$$

3.2. Grupowanie

Większość instrukcji składu matematyki dotyczy tylko jednego, następującego po instrukcji znaku. Jeżeli polecenie ma dotyczyć grupy znaków, to należy je umieścić wewnątrz pary nawiasów klamrowych `{...}`.

```
\begin{equation}
a^x+y \neq a^{x+y}
\end{equation}
```

$$a^x + y \neq a^{x+y} \tag{3.4}$$

3.3. Części składowe wyrażeń matematycznych

W tym podrozdziale opiszemy ważniejsze instrukcje do składu wyrażeń. Zestawienie wszystkich dostępnych symboli i znaków podajemy w punkcie 3.9 na stronie 51.

Małe litery alfabetu greckiego wprowadzamy, używając instrukcji typu: `\alpha`, `\beta`, `\gamma` itd., a duże[2]: `\Delta`, `\Gamma`, itd.

```
$\lambda,\xi,\pi,\mu,\Phi,\Omega$
```

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

Indeksy górne i **wykładniki** otrzymujemy za pomocą znaku `^`, a **dolne** stosując `_`.

```
$a_{1} x^{2} e^{-\alpha t}
a^{3}_{ij} e^{x^2} \neq {e^x}^2$
```

$a_{1} x^{2} e^{-\alpha t} a^{3}_{ij} e^{x^2} \neq {e^x}^2$

Pierwiastek kwadratowy składamy poleceniem `\sqrt`. Wielkość znaku pierwiastka LaTeX ustala automatycznie. Zapis samego znaku pierwiastka

[2] Obecnie brak dużej litery *Alpha*, ponieważ wygląda ona identycznie jak pierwsza litera *A* alfabetu łacińskiego. Sytuacja ta ma się zmienić po wprowadzeniu nowego sposobu kodowania symboli matematycznych.

umożliwia nam instrukcja `\surd`[3]. Natomiast pierwiastek stopnia n składamy konstrukcją `\sqrt[`n`]`.

```
$\sqrt{x} \sqrt{ x^{2}+\sqrt{y}}
\sqrt[3]{2} \surd[x^2 + y^2]$
```

$\sqrt{x} \sqrt{ x^{2}+\sqrt{y}} \sqrt[3]{2} \surd[x^2 + y^2]$

Polecenia `\overline` oraz `\underline` umieszczają poziome kreski nad i pod wyrażeniami.

```
$\overline{m+n} \underline{x+y}$
```

$\overline{m+n} \underline{x+y}$

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

```
$\underbrace{ a+b+\cdots+z }_{26}$
```

$\underbrace{ a+b+\cdots+z }_{26}$

Akcenty matematyczne, takie jak daszki czy tyldy nad zmiennymi, umieszczamy we wzorze za pomocą instrukcji zestawionych w tabeli 3.1. Szerokie daszki i tyldy, obejmujące wiele symboli, wstawiamy używając instrukcji `\widetilde` oraz `\widehat`. Użycie znaku „`'`" powoduje wstawienie symbolu „prim".

```
\begin{displaymath}
\hat y=x^{2}\quad y'=2x'''
\end{displaymath}
```

$$\hat y=x^{2}\quad y'=2x'''$$

Niekiedy **wektory** wyróżniamy, wstawiając akcent w postaci strzałki nad nazwą zmiennej. Służy do tego polecenie `\vec`. Natomiast do oznaczenia wektora od punktu A do punktu B korzystamy z poleceń `\overrightarrow` oraz `\overleftarrow`.

```
\begin{displaymath}
\vec a\quad\overrightarrow{AB}
\end{displaymath}
```

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

Nazwy funkcji typu „logarytm" należy składać odmianą prostą, nie zaś kursywą, zarezerwowaną dla nazw zmiennych. Oto lista poleceń LaTeX-a służących do składu rozmaitych funkcji matematycznych:

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

[3]Taki zapis jest wykorzystywany raczej w literaturze anglosaskiej.

```
\[\lim_{x \rightarrow 0}
\frac{\sin x}{x}=1\]
```

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

W Polsce nazwy niektórych funkcji trygonometrycznych różnią się od anglosaskich. Pakiet platex – po dołączeniu do dokumentu – na życzenie zmienia standardowe funkcje LaTeX-a tak, że są one zgodne z polskimi zwyczajami. Oto angielskie oryginały: tan, coth, tanh, arccos, arcsin i ich polskie odpowiedniki: tg, ctgh, tgh, arc cos, arc sin.

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

Ułamki piętrowe składa się poleceniem `\frac{...}{...}`. Jednak stosując ułamki zwykłe typu 1/2, lepiej zapisywać je z ukośną kreską:

```
$1\frac{1}{2}$~godziny
\begin{displaymath}
\frac{ x^{2} }{ k+1 }\quad
x^{ \frac{2}{k+1} }\quad x^{ 1/2 }
\end{displaymath}
```

$1\frac{1}{2}$ godziny

$$\frac{x^{2}}{k+1}\quad x^{\frac{2}{k+1}}\quad x^{1/2}$$

Do składu dwumianów lub podobnych konstrukcji możemy wykorzystać polecenie `{... \choose ...}` albo `{... \atop ...}`. Instrukcja `\atop` daje w rezultacie to samo co `\choose`, tyle tylko że bez nawiasów.

```
\begin{displaymath}
{n \choose k}\qquad {x \atop y+2}
\end{displaymath}
```

$$\binom{n}{k}\qquad \genfrac{}{}{0pt}{}{x}{y+2}$$

Znak całki składamy poleceniem `\int`, natomiast **znak sumowania** za pomocą instrukcji `\sum`. Górne granice całkowania/sumowania określamy za pomocą `^` a dolne `_`, podobnie jak w przypadku indeksów górnych i dolnych.

```
\begin{displaymath}
\sum_{i=1}^{n} \quad
\int_{0}^{\frac{\pi}{2}}
\end{displaymath}
```

$$\sum_{i=1}^{n}\quad\int_{0}^{\frac{\pi}{2}}$$

Do składu **nawiasów** i innych **ograniczników** (takich, jak: $[\ \langle\ \|\ \updownarrow$) używa się wielu różnych symboli. Nawiasy okrągłe i kwadratowe wstawiamy bezpośrednio z klawiatury. Nawiasy klamrowe wstawiamy za pomocą poleceń `\{` oraz `\}`. Wszystkie inne ograniczniki wstawiamy z użyciem specjalnych poleceń (np. `\updownarrow`). Zestawienie wszystkich dostępnych ograniczników znajduje się w tabeli 3.8 na stronie 54.

```
\begin{displaymath}
{a,b,c}\neq\{a,b,c\}
\end{displaymath}
```

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

Poprzedzenie otwierającego ogranicznika poleceniem `\left`, a zamykającego poleceniem `\right`, powoduje automatyczne ustalenie jego rozmiaru w zależności od wielkości zawartego między nimi wyrażenia. Uwaga: 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 konstrukcji `\left.` (po lewej) lub `\right.` (po prawej).

```
\begin{displaymath}
1 + \left( \frac{1}{ 1-x^{2} }
\right) ^3
\end{displaymath}
```

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

W pewnych sytuacjach należy samemu określić właściwą wielkość ogranicznika. Do tego celu służą instrukcje `\big`, `\Big`, `\bigg` oraz `\Bigg`, poprzedzające odpowiedni ogranicznik[4].

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

$\Big((x+1) (x-1) \Big) ^{2}$
$\big(\Big(\bigg(\Bigg($ $\big\}\Big\}\bigg\}\Bigg\}$ $\big\|\Big\|\bigg\|\Bigg\|$

Wielokropek w wyrażeniach matematycznych wprowadzamy poleceniem `\ldots`. Kropki pojawiają się wtedy na linii podstawowej, to znaczy na jednakowej wysokości z przecinkiem czy kropką. Instrukcja `\cdots` wstawia natomiast inny rodzaj wielokropka, w którym kropki znajdują się w osi znaków $+$, $-$, $=$. Ponadto są jeszcze instrukcje `\vdots` oraz `\ddots`. Pierwszą z nich składamy wielokropki pionowe, a drugą skośne (zobacz przykład w punkcie 3.5).

```
\begin{displaymath}
x_{1},\ldots,x_{n} \qquad
x_{1}+\cdots+x_{n}
\end{displaymath}
```

$$x_{1},\ldots,x_{n} \qquad x_{1}+\cdots+x_{n}$$

3.4. Odstępy w trybie matematycznym

Zdarzają się sytuacje, kiedy wielkość odstępów wewnątrz wyrażeń matematycznych jest nieodpowiednia. Można jednak skorygować je samemu za pomocą odpowiednich instrukcji. Do wprowadzenia niewielkich odstępów służy kilka poleceń: `\,` wstawia odstęp równy $\frac{3}{18}$ kwadratu (␣), `\:` pozwala uzyskać

[4] Polecenia te nie działają zgodnie z oczekiwaniami, jeżeli uprzednio wykorzystano instrukcje zmieniające stopień pisma albo opcje `11pt` lub `12pt`. W takiej sytuacji należy skorzystać z pakietu exscale albo pakietu amsmath.

odstęp równy $\frac{4}{18}$ kwadratu (␣) a `\;` odstęp równy $\frac{5}{18}$ kwadratu (␣). Użycie instrukcji `\␣` (tj. spacja po znaku „\") jest równoznaczne z utworzeniem zwykłego odstępu międzywyrazowego; `\quad` z odstępem równym kwadratowi (␣␣), a `\qquad` dwóm kwadratom (␣␣␣␣). Odstęp uzyskany poleceniem `\quad` odpowiada szerokości litery „M" w bieżącym kroju pisma. Instrukcja `\!` wstawia odstęp „ujemny", to znaczy zamiast zwiększać, zmniejsza odstęp między znakami. Wielkość tego odstępu wynosi $-\frac{3}{18}$ kwadratu (␣).

```
\newcommand{\ud}{\mathrm{d}}
\begin{displaymath}
\int\!\!\!\int_{D} g(x,y)
\, \ud x\, \ud y
\end{displaymath}
 %
zamiast
\begin{displaymath}
\int\int_{D} g(x,y)\ud x \ud y
\end{displaymath}
```

$$\int\!\!\!\int_D g(x,y)\,\mathrm{d}x\,\mathrm{d}y$$

zamiast

$$\int\int_D g(x,y)\mathrm{d}x\mathrm{d}y$$

Zwróćmy uwagę, że litera „d" w symbolu różniczki jest złożona odmianą prostą pisma[5].

Dzięki zdefiniowanym w pakiecie $\mathcal{A}\!\mathcal{M}\!\mathcal{S}$-LaTeX[6] takim instrukcjom jak `\iint`, `\iiint`, `\iiiint` oraz `\idotsint` powyższy przykład można złożyć dużo prościej:

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

\begin{displaymath}
\iint_{D} \, \ud x \, \ud y
\end{displaymath}
```

$$\iint_D \,\mathrm{d}x\,\mathrm{d}y$$

Więcej wiadomości na ten temat znajdziemy w pliku `testmath.tex`, który jest częścią pakietu $\mathcal{A}\!\mathcal{M}\!\mathcal{S}$-LaTeX lub w rozdziale ósmym podręcznika *The LaTeX Companion*[7].

3.5. Wyrównywanie w pionie

Do składania macierzy wykorzystuje się środowisko `array`. Jest ono podobne do omawianego uprzednio środowiska `tabular`. Używane tu polecenie `\\` oznacza przejście do nowego wiersza macierzy.

[5] W Polsce do składu litery „d" w różniczkach stosuje się kursywę matematyczną.
[6] Ściślej mówiąc, w pakiecie amsmath.
[7] `CTAN://tex-archive/info/ch8`.

```
\begin{displaymath}
\mathbf{X} =
\left( \begin{array}{ccc}
x_{11} & x_{12} & \ldots \\
x_{21} & x_{22} & \ldots \\
\vdots & \vdots & \ddots
\end{array} \right)
\end{displaymath}
```

$$\mathbf{X} = \left(\begin{array}{ccc} x_{11} & x_{12} & \ldots \\ x_{21} & x_{22} & \ldots \\ \vdots & \vdots & \ddots \end{array} \right)$$

Środowiskiem `array` możemy się posłużyć także do składania wyrażeń zawierających tylko jeden ogranicznik, po prawej lub lewej stronie, stosując konstrukcję `\right.` (lub `\left.`).

```
\begin{displaymath}
y = \left\{ \begin{array}{ll}
a & \textrm{jeżeli $d>c$}\\
b+x & \textrm{rano}\\
l & \textrm{w~ciągu dnia}
\end{array} \right.
\end{displaymath}
```

$$y = \left\{ \begin{array}{ll} a & \textrm{jeżeli } d>c \\ b+x & \textrm{rano} \\ l & \textrm{w ciągu dnia} \end{array} \right.$$

Do składu wyrażeń wielowierszowych można zamiast środowiska `equation` użyć środowisk `eqnarray` lub `eqnarray*`. W środowisku `eqnarray` każdy wiersz zawartego w nim wyrażenia posiada osobny numer; w środowisku `eqnarray*` wiersze nie są numerowane. Działanie środowisk `eqnarray` oraz `eqnarray*` jest zbliżone do trzykolumnowej tabeli typu `{rcl}`. W takiej tabeli w środkowej kolumnie wstawiamy zwykle znaki równości lub nierówności. Poleceniem `\\` łamiemy poszczególne wiersze tej tabeli (środowiska).

```
\begin{eqnarray}
f(x) & = & \cos x \\
f'(x) & = & -\sin x \\
\int_{0}^{x} f(y)dy &
= & \sin x
\end{eqnarray}
```

$$f(x) \quad = \quad \cos x \qquad (3.5)$$
$$f'(x) \quad = \quad -\sin x \qquad (3.6)$$
$$\int_{0}^{x} f(y)dy \quad = \quad \sin x \qquad (3.7)$$

Zwróćmy uwagę, że po obu stronach znaku równości odstęp, który wstawił tam LaTeX, jest zbyt duży. Możemy go zmniejszyć za pomocą polecenia `\setlength\arraycolsep{2pt}`.

LaTeX nie dzieli automatycznie wyrażeń nie mieszczących sie w jednym wierszu. Musimy to zrobić sami. Najczęściej w taki oto sposób:

```
{\setlength\arraycolsep{2pt}
\begin{eqnarray}
\sin x & = & x -\frac{x^{3}}{3!}
+\frac{x^{5}}{5!}-{}
\nonumber\\
& & {}-\frac{x^{7}}{7!}+{}\cdots
\end{eqnarray}}
```

$$\sin x = x - \frac{x^3}{3!} + \frac{x^5}{5!} -$$
$$- \frac{x^7}{7!} + \cdots \qquad (3.8)$$

```
\begin{eqnarray}
\lefteqn{ \cos x = 1
-\frac{x^{2}}{2!} +{} }
\nonumber\\
& & {}+\frac{x^{4}}{4!}
-\frac{x^{6}}{6!}+{}\cdots
\end{eqnarray}
```

$$\begin{aligned}&\cos x = 1 - \frac{x^2}{2!} + \\ &\quad + \frac{x^4}{4!} - \frac{x^6}{6!} + \cdots \qquad (3.9)\end{aligned}$$

LaTeX nie wstawia numeru w tym wierszu wyrażenia, w którym pojawia się polecenie `\nonumber`.

Złożenie w ten sposób dużych i skomplikowanych wyrażeń może się jednak okazać zbyt trudne; lepiej wtedy użyć pakietu amsmath.

3.6. Stopień pisma

W trybie matematycznym stopień pisma dobierany jest automatycznie, w zależności od kontekstu. Przykładowo, indeksy górne LaTeX składa mniejszą czcionką. Gdy zachodzi potrzeba złożenia fragmentu normalnego tekstu wewnątrz wyrażenia matematycznego, a wykorzystamy polecenie `\textrm`, to nie zadziała mechanizm przełączania stopni pisma. Dzieje się tak dlatego, że polecenie `\textrm` powoduje tymczasowe przejście do trybu tekstowego.

Zamiast `\textrm` można użyć polecenia `\mathrm`, które zachowuje mechanizm zmiany stopni pisma. Pamiętajmy jednak, że działa on dobrze w zasadzie tylko dla pojedynczych liter. W dalszym ciągu ignorowane są bowiem odstępy między wyrazami, a mechanizm akcentowania nie działa zgodnie z naszymi oczekiwaniami[8].

```
\begin{equation}
2^{\textrm{nd}} \quad
2^{\mathrm{nd}}
\end{equation}
```

$$2^{\textrm{nd}} \quad 2^{\mathrm{nd}} \qquad (3.10)$$

Czasami musimy wyraźnie określić stopień pisma, jakim chcemy się posłużyć. W trybie matematycznym możemy do tego wykorzystać cztery następujące polecenia:

`\displaystyle` (123), `\textstyle` (123), `\scriptstyle` ($\scriptstyle 123$) oraz `\scriptscriptstyle` ($\scriptscriptstyle 123$).

Zmiana stylu dotyczy także sposobu składania indeksów górnych i dolnych takich, jak między innymi granice sumowania czy całkowania.

[8] W pakiecie $\mathcal{AMS}$-LaTeX automatyczną zmianę stopnia pisma w zależności od kontekstu umożliwia polecenie `\textrm`.

```
\begin{displaymath}
\mathop{\mathrm{cov}}(X,Y)=
\frac{\displaystyle
\sum_{i=1}^n(x_i-\overline x)
(y_i-\overline y)}
{\displaystyle\biggl[
\sum_{i=1}^n(x_i-\overline x)^2
\sum_{i=1}^n(y_i-\overline y)^2
\biggr]^{1/2}}
\end{displaymath}
```

$$\mathop{\mathrm{cov}}(X,Y)=\frac{\displaystyle\sum_{i=1}^n(x_i-\overline x)(y_i-\overline y)}{\displaystyle\biggl[\sum_{i=1}^n(x_i-\overline x)^2\sum_{i=1}^n(y_i-\overline y)^2\biggr]^{1/2}}$$

Powyższy przykład ilustruje sytuację, w której należy wykorzystać polecenie `\bigg`, ponieważ nawiasy utworzone za pomocą konstrukcji `\left[` i `\right]` są zbyt małe.

3.7. Twierdzenia, definicje, itp.

W pracach matematycznych występuje potrzeba wyróżniania w składzie zapisu lematów, definicji, aksjomatów i tym podobnych elementów. Do tego celu mamy w LaTeX-u polecenia:

`\newtheorem{`*nazwa*`}[`*nazwa'*`]{`*tekst*`}[`*punkt*`]`

Argument *nazwa* oznacza nazwę środowiska. Argument *tekst* to nazwa elementu, która pojawi się na wydruku. Może to być „Twierdzenie”, „Definicja” itp. Argumenty w nawiasach kwadratowych są nieobowiązkowe. Za ich pomocą określamy sposób numerowania twierdzeń. Argument *nazwa'* to nazwa elementu, który zdefiniowano uprzednio poleceniem `\newtheorem`. Jeśli ten argument podano, to środowisko *nazwa* będzie numerowane w taki sam sposób co środowisko *nazwa'*. Ponadto oba środowiska mają wspólną a nie osobną numerację. Argument *punkt* dotyczy natomiast numerowania twierdzeń wewnątrz określonej jednostki podziału[9].

Po umieszczeniu instrukcji `\newtheorem{`*nazwa*`}...` w preambule, środowisko *nazwa* można wykorzystywać w dokumencie w następujący sposób:

> `\begin{`*nazwa*`}[`*tekst*`]`
> Oto moje interesujące twierdzenie
> `\end{`*nazwa*`}`

Tyle teoria. Następujące przykłady, miejmy nadzieję, usuną wszelkie wątpliwości i jednocześnie uświadomią nam, że polecenie `\newtheorem` może być cokolwiek trudne do zrozumienia.

[9] Gdy umieścimy tam na przykład `chapter`, to elementy będą numerowane w obrębie rozdziałów.

```
% definicje w~preambule
\newtheorem{twr}{Twierdzenie}
\newtheorem{lem}[twr]{Lemat}
% po \begin{document}
\begin{lem} Pierszy
lemat\dots\label{lem:1} \end{lem}
\begin{twr}[Dyzma]
Przyjmując w~lemacie~\ref{lem:1},
że $\epsilon=0$\dots \end{twr}
\begin{lem}Trzeci lemat\end{lem}
```

Lemat 1 *Pierszy lemat ...*

Twierdzenie 2 (Dyzma) *Przyjmując w lemacie 1, że* $\epsilon = 0$ *...*

Lemat 3 *Trzeci lemat*

Elementy Twierdzenie i Lemat wykorzystują ten sam licznik. Argument nieobowiązkowy (wewnątrz nawiasów kwadratowych) służy do umieszczenia nazwiska twórcy, komentarza itp.

```
\newtheorem{mur}{Murphy}[section]
\begin{mur} Jeżeli istnieją dwa
sposoby lub więcej wykonania czegoś,
przy czym jeden z~nich prowadzi do
katastrofy, to sposób ten zostanie
przez kogoś wybrany.\end{mur}
```

Murphy 3.7.1 *Jeżeli istnieją dwa sposoby lub więcej wykonania czegoś, 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. Możliwe jest także wykorzystanie do numeracji twierdzeń innych jednostek podziału dokumentu, jak rozdziały czy podpunkty.

3.8. Symbole półgrube

Wstawianie symboli półgrubych jest w LaTeX-u zadaniem dość trudnym. Być może jest tak celowo, ponieważ amatorzy-składacze mają skłonności do ich nadużywania. Poleceniem `\mathbf` uzyskamy odmianę półgrubą. Niestety, nie będzie to półgruba kursywa, jaką zwykle składane są symbole matematyczne. Istnieje, co prawda, polecenie `\boldmath`, ale można go użyć jedynie *poza trybem matematycznym*. Jego działanie obejmuje też symbole.

```
\begin{displaymath}
\mu, M \qquad \mathbf{M} \qquad
\mbox{\boldmath $\mu, M$}
\end{displaymath}
```

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

Zwróćmy uwagę, że przecinek też jest półgruby; efekt to raczej niepożądany.

Pakiet amsbsy (dołączany przez amsmath) czyni nasze zadanie dużo łatwiejszym. W pakiecie tym występują polecenia `\boldsymbol` oraz `\pmb`. Instrukcja `pmb` imituje znak półgruby przez wydrukowanie dwóch nakładających się na siebie znaków, złożonych w odmianie normalnej. Umożliwia to stosowanie symboli półgrubych, nawet wtedy, gdy w systemie brak odpowiednich fontów.

```
\begin{displaymath} \mu, M \quad
\boldsymbol{\mu},\boldsymbol{M}\quad
\pmb{\mu}, \pmb{M} \end{displaymath}
```

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

3.9. Zestawienie symboli matematycznych

W poniższych tabelach zestawiono wszystkie symbole standardowo dostępne w *trybie matematycznym*. Symbole w tabelach 3.12–3.16[10] są dostępne, jeżeli mamy zainstalowane dodatkowe fonty matematyczne (*AMS math fonts*) i do dokumentu dołączymy pakiet amssymb. Jeżeli brak jest fontów/pakietu to jest on dostępny w: `CTAN://tex-archive/macros/latex/packages/amslatex`.

Tabela 3.1: Akcenty matematyczne

$\hat{a}$	`\hat{a}`	$\check{a}$	`\check{a}`	$\tilde{a}$	`\tilde{a}`	$\acute{a}$	`\acute{a}`
$\grave{a}$	`\grave{a}`	$\dot{a}$	`\dot{a}`	$\ddot{a}$	`\ddot{a}`	$\breve{a}$	`\breve{a}`
$\bar{a}$	`\bar{a}`	$\vec{a}$	`\vec{a}`	$\widehat{A}$	`\widehat{A}`	$\widetilde{A}$	`\widetilde{A}`

Tabela 3.2: Małe litery alfabetu greckiego

α	`\alpha`	θ	`\theta`	o	`o`	υ	`\upsilon`
β	`\beta`	ϑ	`\vartheta`	π	`\pi`	ϕ	`\phi`
γ	`\gamma`	ι	`\iota`	ϖ	`\varpi`	φ	`\varphi`
δ	`\delta`	κ	`\kappa`	ρ	`\rho`	χ	`\chi`
ϵ	`\epsilon`	λ	`\lambda`	ϱ	`\varrho`	ψ	`\psi`
ε	`\varepsilon`	μ	`\mu`	σ	`\sigma`	ω	`\omega`
ζ	`\zeta`	ν	`\nu`	ς	`\varsigma`		
η	`\eta`	ξ	`\xi`	τ	`\tau`		

Tabela 3.3: Duże litery alfabetu greckiego

Γ	`\Gamma`	Λ	`\Lambda`	Σ	`\Sigma`	Ψ	`\Psi`
Δ	`\Delta`	Ξ	`\Xi`	Υ	`\Upsilon`	Ω	`\Omega`
Θ	`\Theta`	Π	`\Pi`	Φ	`\Phi`		

[10]Tabele przygotowano na podstawie pliku `symbols.tex` (David Carlisle), gruntownie zmodyfikowanego zgodnie z sugestiami Josefa Tkadleca.

Tabela 3.4: Symbole relacji

Odpowiednie symbole negacji można utworzyć poprzedzając każde z poniższych poleceń instrukcją `\not`.

$<$	`<`	$>$	`>`	$=$	`=`
$\leqslant$	`\leq` albo `\le`	$\geqslant$	`\geq` albo `\ge`	$\equiv$	`\equiv`
$\ll$	`\ll`	$\gg$	`\gg`	$\doteq$	`\doteq`
$\prec$	`\prec`	$\succ$	`\succ`	$\sim$	`\sim`
$\preceq$	`\preceq`	$\succeq$	`\succeq`	$\simeq$	`\simeq`
$\subset$	`\subset`	$\supset$	`\supset`	$\approx$	`\approx`
$\subseteq$	`\subseteq`	$\supseteq$	`\supseteq`	$\cong$	`\cong`
$\sqsubset$	`\sqsubset` [a]	$\sqsupset$	`\sqsupset` [a]	$\Join$	`\Join` [a]
$\sqsubseteq$	`\sqsubseteq`	$\sqsupseteq$	`\sqsupseteq`	$\bowtie$	`\bowtie`
$\in$	`\in`	$\ni$	`\ni` , `\owns`	$\propto$	`\propto`
$\vdash$	`\vdash`	$\dashv$	`\dashv`	$\models$	`\models`
$\mid$	`\mid`	$\parallel$	`\parallel`	$\perp$	`\perp`
$\smile$	`\smile`	$\frown$	`\frown`	$\asymp$	`\asymp`
$:$	`:`	$\notin$	`\notin`	$\neq$	`\neq` albo `\ne`

[a] Dostępne po dołączeniu pakietu latexsym

W trybie matematycznym LaTeX wstawia dodatkowy mały odstęp po przecinku i średniku. Natomiast po dwukropku wstawia odstęp przed i za znakiem, bo traktuje go jako znak relacji. Różnicę przedstawiono w poniższym przykładzie:

```
Nie $f:A\to B$ ale $f\colon A\to B$
```

Nie $f : A \to B$ ale $f\colon A \to B$

Jeżeli przecinek oddziela część całkowitą liczby od części dziesiętnej, to wskazane jest zakazać LaTeX-owi wstawiania dodatkowego odstępu, jak to zwykle robi w trybie matematycznym. Do tego celu wystarczy otoczyć przecinek parą nawiasów klamrowych; porównajmy:

```
Nie $22,115$
ale $22{,}115$
```

Nie $22,115$ ale $22{,}115$

Kształty znaków mniejszy-równy i większy-równy są różne od tych stosowanych w krajach anglosaskich. Po dołączeniu pakietu platex standardowe LaTeX-owe polecenia `\leq` oraz `\geq` dają w rezultacie polskie wersje tych relacji, tj. $\leqslant$, $\geqslant$, a nie $\leq$, $\geq$.

Tabela 3.5: Symbole operacji dwuargumentowych

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

Tabela 3.6: Symbole zmiennej wielkości

$\sum$	`\sum`	$\bigcup$	`\bigcup`	$\bigvee$	`\bigvee`	$\bigoplus$	`\bigoplus`
$\prod$	`\prod`	$\bigcap$	`\bigcap`	$\bigwedge$	`\bigwedge`	$\bigotimes$	`\bigotimes`
$\coprod$	`\coprod`	$\bigsqcup$	`\bigsqcup`			$\bigodot$	`\bigodot`
$\int$	`\int`	$\oint$	`\oint`			$\biguplus$	`\biguplus`

Tabela 3.7: Strzałki

$\leftarrow$	`\leftarrow` albo `\gets`	$\longleftarrow$	`\longleftarrow`	$\uparrow$	`\uparrow`
$\rightarrow$	`\rightarrow` albo `\to`	$\longrightarrow$	`\longrightarrow`	$\downarrow$	`\downarrow`
$\leftrightarrow$	`\leftrightarrow`	$\longleftrightarrow$	`\longleftrightarrow`	$\updownarrow$	`\updownarrow`
$\Leftarrow$	`\Leftarrow`	$\Longleftarrow$	`\Longleftarrow`	$\Uparrow$	`\Uparrow`
$\Rightarrow$	`\Rightarrow`	$\Longrightarrow$	`\Longrightarrow`	$\Downarrow$	`\Downarrow`
$\Leftrightarrow$	`\Leftrightarrow`	$\Longleftrightarrow$	`\Longleftrightarrow`	$\Updownarrow$	`\Updownarrow`
$\mapsto$	`\mapsto`	$\longmapsto$	`\longmapsto`	$\nearrow$	`\nearrow`
$\hookleftarrow$	`\hookleftarrow`	$\hookrightarrow$	`\hookrightarrow`	$\searrow$	`\searrow`
$\leftharpoonup$	`\leftharpoonup`	$\rightharpoonup$	`\rightharpoonup`	$\swarrow$	`\swarrow`
$\leftharpoondown$	`\leftharpoondown`	$\rightharpoondown$	`\rightharpoondown`	$\nwarrow$	`\nwarrow`
$\rightleftharpoons$	`\rightleftharpoons`	$\iff$	`\iff`	$\leadsto$	`\leadsto` [a]

[a] Dostępne po dołączeniu pakietu latexsym

Tabela 3.8: Ograniczniki

<table>
<tr><td>$($</td><td>(</td><td>$)$</td><td>)</td><td>$\uparrow$</td><td>\uparrow</td><td>$\Uparrow$</td><td>\Uparrow</td></tr>
<tr><td>$[$</td><td>[albo \lbrack</td><td>$]$</td><td>] albo \rbrack</td><td>$\downarrow$</td><td>\downarrow</td><td>$\Downarrow$</td><td>\Downarrow</td></tr>
<tr><td>$\{$</td><td>\{ albo \lbrace</td><td>$\}$</td><td>\} albo \rbrace</td><td>$\updownarrow$</td><td>\updownarrow</td><td>$\Updownarrow$</td><td>\Updownarrow</td></tr>
<tr><td>$\langle$</td><td>\langle</td><td>$\rangle$</td><td>\rangle</td><td>$\vert$</td><td>| albo \vert</td><td>$\Vert$</td><td>\| albo \Vert</td></tr>
<tr><td>$\lfloor$</td><td>\lfloor</td><td>$\rfloor$</td><td>\rfloor</td><td>$\lceil$</td><td>\lceil</td><td>$\rceil$</td><td>\rceil</td></tr>
<tr><td>$/$</td><td>/</td><td>$\backslash$</td><td>\backslash</td><td></td><td></td><td></td><td></td></tr>
</table>

Tabela 3.9: Duże ograniczniki

<table>
<tr><td>$\lgroup$</td><td>\lgroup</td><td>$\rgroup$</td><td>\rgroup</td><td>$\lmoustache$</td><td>\lmoustache</td><td>$\rmoustache$</td><td>\rmoustache</td></tr>
<tr><td>$\arrowvert$</td><td>\arrowvert</td><td>$\Arrowvert$</td><td>\Arrowvert</td><td>$\bracevert$</td><td>\bracevert</td><td></td><td></td></tr>
</table>

Tabela 3.10: Symbole różne

$\dots$	`\dots`	$\cdots$	`\cdots`	$\vdots$	`\vdots`	$\ddots$	`\ddots`
$\hbar$	`\hbar`	$\imath$	`\imath`	$\jmath$	`\jmath`	ℓ	`\ell`
$\Re$	`\Re`	$\Im$	`\Im`	$\aleph$	`\aleph`	$\wp$	`\wp`
$\forall$	`\forall`	$\exists$	`\exists`	$\mho$	`\mho` [a]	∂	`\partial`
$'$	'	$\prime$	`\prime`	$\emptyset$	`\emptyset`	∞	`\infty`
∇	`\nabla`	$\triangle$	`\triangle`	$\Box$	`\Box` [a]	$\Diamond$	`\Diamond` [a]
$\bot$	`\bot`	$\top$	`\top`	$\angle$	`\angle`	$\surd$	`\surd`
$\diamondsuit$	`\diamondsuit`	$\heartsuit$	`\heartsuit`	$\clubsuit$	`\clubsuit`	$\spadesuit$	`\spadesuit`
$\neg$	`\neg` albo `\lnot`	$\flat$	`\flat`	$\natural$	`\natural`	$\sharp$	`\sharp`

[a] Dostępne po dołączeniu pakietu latexsym

Tabela 3.11: Symbole niematematyczne

†	`\dag`	§	`\S`	©	`\copyright`	‡	`\ddag`	¶	`\P`	£	`\pounds`

Wszystkie polecenia są także dostępne w trybie tekstowym

Tabela 3.12: Ograniczniki (pakiet AMS)

$\ulcorner$	`\ulcorner`	$\urcorner$	`\urcorner`	$\llcorner$	`\llcorner`	$\lrcorner$	`\lrcorner`

Tabela 3.13: Symbole Greckie i Hebrajskie (pakiet AMS)

$\digamma$	`\digamma`	$\varkappa$	`\varkappa`	$\beth$	`\beth`	$\daleth$	`\daleth`	$\gimel$	`\gimel`

Tabela 3.14: Symbole relacji (pakiet AMS)

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

Tabela 3.15: Strzałki (pakiet AMS)

$\dashleftarrow$	`\dashleftarrow`	$\dashrightarrow$	`\dashrightarrow`	$\multimap$	`\multimap`
$\leftleftarrows$	`\leftleftarrows`	$\rightrightarrows$	`\rightrightarrows`	$\upuparrows$	`\upuparrows`
$\leftrightarrows$	`\leftrightarrows`	$\rightleftarrows$	`\rightleftarrows`	$\downdownarrows$	`\downdownarrows`
$\Lleftarrow$	`\Lleftarrow`	$\Rrightarrow$	`\Rrightarrow`	$\upharpoonleft$	`\upharpoonleft`
$\twoheadleftarrow$	`\twoheadleftarrow`	$\twoheadrightarrow$	`\twoheadrightarrow`	$\upharpoonright$	`\upharpoonright`
$\leftarrowtail$	`\leftarrowtail`	$\rightarrowtail$	`\rightarrowtail`	$\downharpoonleft$	`\downharpoonleft`
$\leftrightharpoons$	`\leftrightharpoons`	$\rightleftharpoons$	`\rightleftharpoons`	$\downharpoonright$	`\downharpoonright`
$\Lsh$	`\Lsh`	$\Rsh$	`\Rsh`	$\rightsquigarrow$	`\rightsquigarrow`
$\looparrowleft$	`\looparrowleft`	$\looparrowright$	`\looparrowright`	$\leftrightsquigarrow$	`\leftrightsquigarrow`
$\curvearrowleft$	`\curvearrowleft`	$\curvearrowright$	`\curvearrowright`		
$\circlearrowleft$	`\circlearrowleft`	$\circlearrowright$	`\circlearrowright`		

Tabela 3.16: Negacje symboli relacji i strzałek (pakiet AMS)

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

Tabela 3.17: Relacje dwuargumentowe (pakiet AMS)

$\dotplus$	\dotplus	$\centerdot$	\centerdot	$\intercal$	\intercal
$\ltimes$	\ltimes	$\rtimes$	\rtimes	$\divideontimes$	\divideontimes
$\Cup$	\Cup or \doublecup	$\Cap$	\Cap or \doublecap	$\smallsetminus$	\smallsetminus
$\veebar$	\veebar	$\barwedge$	\barwedge	$\doublebarwedge$	\doublebarwedge
$\boxplus$	\boxplus	$\boxminus$	\boxminus	$\circleddash$	\circleddash
$\boxtimes$	\boxtimes	$\boxdot$	\boxdot	$\circledcirc$	\circledcirc
$\leftthreetimes$	\leftthreetimes	$\rightthreetimes$	\rightthreetimes	$\circledast$	\circledast
$\curlyvee$	\curlyvee	$\curlywedge$	\curlywedge		

Tabela 3.18: Różne symbole (pakiet AMS)

$\hbar$	`\hbar`	$\hslash$	`\hslash`	$\Bbbk$	`\Bbbk`
$\square$	`\square`	$\blacksquare$	`\blacksquare`	$\circledS$	`\circledS`
$\vartriangle$	`\vartriangle`	$\blacktriangle$	`\blacktriangle`	$\complement$	`\complement`
$\triangledown$	`\triangledown`	$\blacktriangledown$	`\blacktriangledown`	$\Game$	`\Game`
$\lozenge$	`\lozenge`	$\blacklozenge$	`\blacklozenge`	$\bigstar$	`\bigstar`
$\angle$	`\angle`	$\measuredangle$	`\measuredangle`	$\sphericalangle$	`\sphericalangle`
$\diagup$	`\diagup`	$\diagdown$	`\diagdown`	$\backprime$	`\backprime`
$\nexists$	`\nexists`	$\Finv$	`\Finv`	$\varnothing$	`\varnothing`
$\eth$	`\eth`	$\mho$	`\mho`		

Tabela 3.19: Kroje pisma dostępne w trybie matematycznym

Przykład	Polecenie	Wymagany pakiet
ABCdef	`\mathrm{ABCdef}`	
ABCdef	`\mathit{ABCdef}`	
$\mathnormal{ABCdef}$	`\mathnormal{ABCdef}`	
$\mathcal{ABC}$	`\mathcal{ABC}`	
$\mathscr{ABC}$	`\mathcal{ABC}`	mathrsfs
$\mathcal{ABC}$	`\mathcal{ABC}`	eucal z opcją: mathcal albo
	`\mathscr{ABC}`	eucal z opcją: mathscr
$\mathfrak{ABCdef}$	`\mathfrak{ABCdef}`	eufrak
$\mathbb{ABC}$	`\mathbb{ABC}`	amsfonts albo amssymb

Rozdział 4

Rysunki, skorowidze ...

Kolej na opis możliwości LaTeX-a przydatnych podczas prac nad większymi dokumentami, tj. dołączania rysunków, tworzenia skorowidzów i spisów literatury. Bardziej szczegółowy opis tych i podobnych zagadnień znajdziemy w LaTeX Manual *[4] i* The LaTeX Companion *[2].*

4.1. Dołączanie grafiki w formacie EPS

Środowisko `figure` pozwala zarezerwować miejsce na wstawienie rysunku do dokumentu LaTeX-owego.

Rysunki możemy tworzyć na wiele sposobów przy wykorzystaniu standardowych oraz niestandardowych pakietów LaTeX-a. Trzeba jednak przyznać, że dla większości użytkowników, nauka posługiwanie się tymi pakietami jest trudna a tworzenie rysunków za ich pomocą pracochłonne. Dlatego nie będziemy się już nimi dalej zajmować. Zainteresowanych odsyłamy do *The LaTeX Companion* [2] oraz *LaTeX Manual* [4].

Znacznie prostszym sposobem wstawiania rysunków do dokumentu jest uprzednie ich przygotowanie za pomocą wyspecjalizowanych programów graficznych, takich jak XFig, CorelDraw!, Freehand, Gnuplot, itd., a następnie dołączenia gotowych rysunków do dokumentu. Także i tym przypadku można to wykonać na wiele sposobów. W tym wprowadzeniu przedstawimy tylko dołączanie grafiki w formacie EPS (*Encapsulated PostScript*) z uwagi na to, że jest to sposób prosty a do tego zdecydowanie najpowszechniej wykorzystywany. Po to żeby wykorzystywać grafikę w formacie EPS potrzebna jest drukarka z wbudowanym interpreterem języka PostScript[1]

Program ghostscript jest interpreterem języka PostScript o statusie oprogramowania *freeware*. Obsługuje kilkadziesiąt różnorakich urządzeń, w tym wiele popularnych drukarek oraz kart graficznych. Pozwala on obejrzeć dokument na ekranie przed drukiem.

[1] Zamiast interpretera w drukarce możemy użyć programu ghostscript dostępnego, np. w `CTAN://tex-archive/support/ghostscript`.

Nie mamy takiej możliwości wykorzystując interpreter w drukarce. Program ghostscript jest dostępny na wszystkich popularnych platformach systemowych.

Do dołączenia rysunków do dokumentu można skorzystać z poleceń pakietu graphicx (autor: D. P. Carlisle). Pakiet ten jest częścią zestawu pakietów zwanego „graphics”[2].

Zakładając, że dysponujemy interpreterem języka PostScript oraz pakietem graphicx, dołączanie grafiki można przedstawić następująco:

1. Zapisujemy rysunek w formacie EPS[3].
2. Dołączamy pakiet graphicx do preambuły dokumentu:

   `\usepackage[`*dvi-ps*`]{graphicx}`

   gdzie *dvi-ps* oznacza nazwę sterownika wykorzystywanego do zamiany pliku `.dvi` na plik postscriptowy. Najczęściej używanym do tego celu programem jest `dvips`. Nazwa sterownika jest tu konieczna z uwagi na brak standardu dotyczącego dołączania grafiki PostScriptowej do dokumentów LaTeX-owych. Podanie nazwy pozwala na wybór metody specyficznej dla danego sterownika.
3. Instrukcją:

   `\includegraphics[`*klucz*=*wartość*, ... `]{`*plik*`}`

   dołączamy *plik* do dokumentu. Parametr opcjonalny jest listą, oddzielonych przecinkami *kluczy* o określonych przez nas *wartościach*. Klucze wykorzystujemy do zmian parametrów rysunku, tj. szerokości czy wysokości rysunku oraz kąta obrotu. W tablicy 4.1 zamieszczono najważniejsze klucze.

Tabela 4.1: Nazwy kluczy w pakiecie graphicx

`width`	skalowanie rysunku do szerokości *width*;
`height`	skalowanie rysunku do wysokości *height*;
`angle`	obrót (przeciwnie do ruchu wskazówek zegara);
`scale`	skalowanie równomierne.

Poniższy przykład pozwoli nam lepiej zrozumieć to, co zostało przedstawione w powyższych punktach:

[2] `CTAN://tex-archive/macros/latex/packages/graphics`

[3] Jeżeli program graficzny, którym się posługujemy nie pozwala nam na to, spróbujmy zainstalować sterownik do drukarki postscriptowej w rodzaju Apple Laser Writer i drukować za pomocą tego sterownika do pliku. Przy odrobinie szczęścia plik wynikowy będzie w formacie EPS. Pamiętajmy jednak, że pliki w tym formacie mają tylko jedną stronę. Sterowniki niektórych drukarek można jawnie ustawić właśnie do tworzenia plików w tym formacie.

```
\begin{figure}
\begin{center}
\includegraphics[angle=90,width=0.5\textwidth]{test.eps}
\end{center}
\end{figure}
```

Dołączony zostaje rysunek zawarty w pliku `test.eps`. Rysunek najpierw obracamy o 90 stopni w kierunku przeciwnym do ruchu wskazówek zegara a następnie przeskalowujemy tak, aby jego szerokość była równa połowie szerokości szpalty. Skalowanie grafiki jest równomierne z uwagi na brak klucza `height`. Szerokość i wysokość rysunku możemy także określić przy użyciu miar bezwzględnych. W tabeli 5.5 zamieszczono zestawienie jednostek miar w LaTeX-u. Więcej informacji dotyczącej powyższych zagadnień znajdziemy w [1] oraz [11].

4.2. Spis literatury

Do przygotowania spisu literatury używamy środowiska `thebibliography`. Każda pozycja w tym spisie rozpoczyna się poleceniem:

`\bibitem{`*etykieta*`}`

Etykietą posługujemy się do cytowania oznaczonej nią pozycji w dokumencie:

`\cite{`*etykieta*`}`

Numerowanie pozycji literaturowych jest automatyczne (polecenia `\bibitem` i `\cite` działają podobnie jak opisane już instrukcje `\label` i `\ref`). Środowisko `thebibliography` ma jeden parametr, który powinien zawierać tekst równy szerokości (lub szerszy) od najszerszej etykiety ze spisu. W przykładzie poniżej „99” oznacza, że numery pozycji w spisie będą co najwyżej dwucyfrowe.

```
Partl~\cite{pa} zaproponował,
żeby \ldots

\begin{thebibliography}{99}
\bibitem{pa} H.~Partl:
\emph{German \TeX},
TUGboat Vol.~9, No.~1 ('88)
\end{thebibliography}
```

Partl [1] zaproponował, żeby ...

Spis literatury

[1] H. Partl: *German TEX*, TUGboat Vol. 9, No. 1 ('88)

Do większych projektów przydaje się program o nazwie BibTEX. Program ten znajduje się w każdej współczesnej dystrybucji TEX-a. BibTEX 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.

Po przetworzeniu pliku, na podstawie zawartości etykiet, zapisanych do pliku `.aux` przez LaTeX-a, BibTeX tworzy spis literatury obejmujący tylko te pozycje z biblioteki (zwykle plik ten ma rozszerzenie `.bib`), które cytowano w dokumencie. Format spisu zależy od specyfikacji szablonu znajdującej się w pliku o rozszerzeniu `.bst` i zapisywany jest do pliku o rozszerzeniu `.bbl`. Do poprawnego sformatowania bibliografii i cytowań konieczne jest przynajmniej trzykrotne przetworzenie dokumentu LaTeX-em.

4.3. Skorowidze

Niezwykle użytecznym elementem wielu książek jest skorowidz. Można go bardzo łatwo utworzyć za pomocą LaTeX-a oraz programu wspomagającego o nazwie `makeindex`[4]. W tym wprowadzeniu omówimy jedynie podstawowe polecenia dotyczące przygotowania skorowidzy. Znowu więcej informacji znajdziemy w *The LaTeX Companion* [2].

Generowanie haseł do skorowidza jest możliwe dopiero po dołączeniu w preambule dokumentu pakietu o nazwie makeidx:

```
\usepackage{makeidx}
```

oraz wstawieniu (także w obrębie preambuły) instrukcji:

```
\makeindex
```

Każde hasło wstawiamy do skorowidza poleceniem:

```
\index{hasło}
```

gdzie *hasło* oznacza pozycję w skorowidzu. Polecenie `\index{`*hasło*`}` umieszczamy w pliku źródłowym bezpośrednio w miejscu związanym z określonym hasłem. W tabeli 4.2 przedstawiono przykłady użycia *haseł*.

Podczas przetwarzania pliku źródłowego przez LaTeX-a każda instrukcja `\index` powoduje zapisanie odpowiedniej pozycji skorowidza oraz numeru strony, którego ta pozycja dotyczy do pliku pomocniczego. Plik ten ma tę samą nazwę co główny plik źródłowy i rozszerzenie `.idx`. Z kolei plik `.idx` musi być przetworzony za pomocą programu `makeindex`, piszemy:

```
makeindex plik
```

[4] Lub `makeidx`, jeśli nasz system operacyjny nie pozwala używać nazw dłuższych niż 8 znaków.

Tabela 4.2: Przykłady składni polecenia `\index`

Przykład	**Argument**	**Uwagi**
`\index{kot}`	kot, 1	hasło pierwszego stopnia
`\index{kot!rudy}`	rudy, 3	hasło drugiego stopnia
`\index{Sam@\textsl{Sam}}`	*Sam*, 2	hasło sformatowane
`\index{Lin@\textbf{Lin}}`	**Lin**, 7	*ditto*
`\index{Jenny|textbf}`	Jenny, **3**	formatowanie numeru strony
`\index{Joe|textit}`	Joe, *5*	*ditto*

W rezultacie program `makeindex` tworzy posortowany skorowidz i zapisuje go do pliku o nazwie identycznej z nazwą głównego pliku źródłowego i rozszerzeniu `.ind`. Jeżeli teraz jeszcze raz przetworzymy plik źródłowy, to tym razem taki skorowidz zostanie dołączony do dokumentu w miejscu, w którym znajduje się polecenie:

```
\printindex
```

Pakiet showidx, który znajduje się w standardowej dystrybucji LaTeX-a drukuje pozycje skorowidza na lewym marginesie. Jest on przydatny do korekt lub/i tworzenia skorowidza.

Reguły sortowania skorowidza są specyficzne dla danego języka. W przypadku pracy nad polskim dokumentem program `makeindex` jest nieprzydatny gdyż sortuje hasła tylko według zasad języka angielskiego. Zmodyfikowaną wersją programu `makeindex` jest `plmindex`, (autor: Włodzimierz Macewicz) zdolny do tworzenia skorowidza zarówno według reguł angielskich, jak i polskich. Program jest dostępny, np. w `ftp://ftp.gust.org.pl/TeX/GUST/contrib/GUSTPROG/plmindex.zip`. Więcej informacji na temat polskich zasad tworzenia skorowidzów można znaleźć w [9].

4.4. Paginy górne i dolne

Pakiet fancyhdr (autor Piet van Oostrum)[5] udostępnia polecenia, którymi definiujemy własne paginy. Zwróćmy uwagę na różnice w wyglądzie pagin na stronach: bieżącej i poprzedniej. Oprócz numeru strony, w paginie górnej na stronie parzystej znajduje się tytuł rozdziału, a na stronie nieparzystej dodatkowo tytuł punktu. Taką paginę fachowo nazywa się żywą paginą.

Za pomocą LaTeX-a można łatwo poradzić sobie z automatycznym umieszczaniem odpowiedniej informacji w żywej paginie. LaTeX rozwiązuje ten problem następująco. W definicjach poleceń składających paginy wykonanie instrukcji `\rightmark` oraz `\leftmark` wstawia odpowiedni tytuł rozdziału, punktu lub cokolwiek innego. Poleceniom `\rightmark` i `\leftmark`

[5] Dostępny w: `CTAN://tex-archive/macros/latex/contrib/supported/fancyhdr/`.

```
\documentclass{book}
\usepackage{fancyhdr}
\pagestyle{fancy}
% zmiana liter w~żywej paginie na małe
\renewcommand{\chaptermark}[1]{\markboth{#1}{}}
\renewcommand{\sectionmark}[1]{\markright{\thesection\ #1}}
\fancyhf{} % usuń bieżące ustawienia pagin
\fancyhead[LE,RO]{\small\bfseries\thepage}
\fancyhead[LO]{\small\bfseries\rightmark}
\fancyhead[RE]{\small\bfseries\leftmark}
\renewcommand{\headrulewidth}{0.5pt}
\renewcommand{\footrulewidth}{0pt}
\addtolength{\headheight}{0.5pt} % pionowy odstęp na kreskę
\fancypagestyle{plain}{%
\fancyhead{} % usuń p. górne na stronach pozbawionych
% numeracji (plain)
\renewcommand{\headrulewidth}{0pt} % pozioma kreska
}
```

Rysunek 4.1: Przykład wykorzystania pakietu fancyhdr

jest nadawane nowe znaczenie (nowe wartości) za każdym wykonaniem instrukcji składania tytułu rozdziału i punktu (`\chapter`, `\section`).

W rzeczywistości, `\chapter` oraz inne polecenia podziału logicznego dokumentu nie zmieniają definicji poleceń `\rightmark` i `\leftmark`. Odwołują się one natomiast do poleceń `\chaptermark`, `\sectionmark`, lub `\subsectionmark`. Dopiero użycie tych instrukcji powoduje zmianę definicji poleceń `\rightmark` i `\leftmark`.

Do zmiany postaci tytułu rozdziału w główce wystarczy modyfikacja polecenia `\chaptermark`. Rysunek 4.1 przedstawia takie wykorzystanie pakietu, że paginy górne będą wyglądać mniej więcej tak, jak w tym podręczniku. Kompletny opis pakietu znajduje się w jego dokumentacji dostępnej pod adresem podanym w ostatnim przypisie.

4.5. Pakiet verbatim

Pakiet `verbatim` udostępnia poprawioną wersję standardowego środowiska `verbatim` (opisanego na stronie 33). Oprócz wielu drobnych ale istotnych ulepszeń pakiet udostępnia polecenie:

`\verbatiminput{`*plik*`}`

które pozwala dołączyć dosłownie (*verbatim*) plik tekstowy do dokumentu, tak jakby jego zawartość znajdowała się wewnątrz środowiska `verbatim`.

Pakiet verbatim jest częścią zestawu pakietów „tools". Zestaw ten wchodzi w skład standardowej dystrybucji LaTeX-a. Więcej szczegółów jest w [12].

Rozdział 5

Adaptowanie LaTeX-a

Dokumenty składane z wykorzystaniem poznanych do tej pory poleceń zapewne będą się podobać zdecydowanej większości czytelników. Chociaż ich wygląd nie będzie może wyrafinowany, z pewnością jednak spełnią podstawowe zasady składu, dzięki czemu łatwo i przyjemnie będzie się je czytało.

W niektórych wypadkach może się jednak okazać, że brakuje polecenia czy środowiska, za pomocą którego moglibyśmy złożyć dany fragment tekstu w sposób odpowiadający potrzebom, albo też, że sposób działania dostępnej w LaTeX-u instrukcji nie spełnia naszych wymagań.

W tym rozdziale przedstawimy, jak nauczyć LaTeX-a formatować dokumenty tak, aby wyglądały inaczej, niż w przypadku korzystania tylko ze standardowych klas i pakietów.

5.1. Definiowane instrukcji i środowisk

Czytelnicy zapewne zauważyli, że nowo wprowadzane w tej książce polecenia ukazują się w ramkach oraz, że znajdują się w skorowidzu. Aby to osiągnąć, nie korzystaliśmy za każdym razem z wbudowanych w LaTeX-a instrukcji, tylko utworzyliśmy własny pakiet, w którym zawarliśmy nowe, potrzebne nam polecenia i środowiska. Mając taki pakiet, wystarczy po prostu napisać:

```
\begin{command}
\ci{polecenie}
\end{command}
```

`\polecenie`

W tym przykładzie użyliśmy zarówno nowego środowiska o nazwie `command`, odpowiedzialnego za rysowanie ramek dookoła instrukcji, jak też nowego polecenia `\ci`, służącego do składu nazw poleceń i wprowadzania ich do skorowidza. Proponujemy Czytelnikom odszukanie hasła `\polecenie` w skorowidzu; przy haśle powinny być podane numery stron, na których ta instrukcja występuje w książce.

We wstępie do tego opracowania wspomnieliśmy, że w LaTeX-u możemy się skupić na logicznej strukturze dokumentów. Wskazane jest rozróżnianie w tekście źródłowym wszystkich elementów logicznych dokumentu, nawet jeżeli ich formatowanie jest identyczne. Często to, co dzisiaj formatujemy jednakowo, w przyszłości możemy chcieć rozróżnić. Przykładowo, adresy internetowe zwykło się składać imitacją kroju maszynowego, a ponieważ w adresach mogą wystąpić znaki specjalne LaTeX-a, można by do tego celu wykorzystywać po prostu instrukcję `\verb`. Jednak lepiej użyć specjalnej instrukcji, np. `\url`. W dokumencie drukowanym na papierze nie ma to znaczenia, ale oznakowanie logiczne umożliwia przedstawienie dokumentu nie tylko w formie drukowanej, ale też w wersji hipertekstowej, w formacie HTML czy PDF.

5.1.1. Instrukcje definiowane przez użytkownika

Do definiowania nowych poleceń służy instrukcja:

`\newcommand{`*nazwa*`}[`*num*`]{`*tekst*`}`

Instrukcja ta wymaga *co najmniej* dwóch argumentów. Pierwszy z nich, *nazwa*, oznacza nazwę nowej instrukcji, natomiast *tekst* to jej znaczenie, czyli tekst, który ma zostać wstawiony do składu w momencie wykonania instrukcji. Podawany w nawiasach kwadratowych argument *num* to liczba od 1 do 9 określająca liczbę (obowiązkowych) argumentów instrukcji. Argument *num* jest opcjonalny, a jego pominięcie oznacza, że definiowana instrukcja jest bezargumentowa.

W części *tekst* wolno używać zarówno standardowych instrukcji LaTeX-a, jak i zdefiniowanych wcześniej przez użytkownika (za wyjątkiem tych instrukcji, które same definiują inne polecenia, jak `\newcommand`, `\newenvironment` itp.). Niedozwolona jest rekursja. W nazwie instrukcji nie wolno umieszczać polskich liter diakrytycznych.

Następujące przykłady pomogą lepiej zrozumieć zagadnienie. W pierwszym definiujemy instrukcję o nazwie `\kwle`, mającą być skrótem dla słów „Krótkie wprowadzenie do systemu LaTeX 2_ε". Takie polecenie mogłoby się przydać, gdyby tytuł książki miał być wypisywany wielokrotnie.

```
\newcommand{\kwle}{Krótkie
wprowadzenie do systemu \LaTeXe}
% następnie po \begin{document}:
\kwle; \emph{\kwle}
```

Krótkie wprowadzenie do systemu LaTeX 2_ε;
Krótkie wprowadzenie do systemu LaTeX 2_ε

Następny przykład ilustruje sposób wykorzystania opcjonalnego argumentu *num*. Znacznik `#1` oznacza pierwszy parametr formalny (`#2` oznaczałby drugi parametr formalny, `#3` trzeci itd.). W chwili wykonania instrukcji parametry formalne są zamieniane na argumenty podane przy jej wywołaniu (parametry aktualne). Instrukcja w poniższym przykładzie ma jeden parametr.

```
\newcommand{\wle}[1]
{\emph{#1} wprowadzenie
do systemu \LaTeXe}
% następnie po \begin{document}:
\wle{Krótkie}; \wle{Długie}
```

Krótkie wprowadzenie do systemu LaTeX 2_ε; *Długie* wprowadzenie do systemu LaTeX 2_ε

LaTeX nie pozwala definiować instrukcji, której nazwa, jest nazwą zdefiniowanego już polecenia. W wypadku, gdy chcemy zmienić znaczenie już istniejącej instrukcji, należy użyć polecenia `\renewcommand`. Za wyjątkiem nazwy ma ono składnię identyczną jak `\newcommand`. Czasami może się przydać polecenie `\providecommand`. Działa ono jak `\newcommand` z tym, że jeśli istnieje już polecenie o takiej samej nazwie, to nie zastępuje ono starej i zwyczajnie ignoruje nową.

5.1.2. Środowiska definiowane przez użytkownika

Odpowiednikiem instrukcji `\newcommand`, definiującej nowe polecenia, jest dla środowisk instrukcja `\newenvironment`. Ma ona następującą składnię:

`\newenvironment{`*nazwa*`}[`*num*`]{`*początek*`}{`*koniec*`}`

Podobnie jak w wypadku `\newcommand`, instrukcji `\newenvironment` można użyć z argumentem opcjonalnym lub bez. LaTeX wstawia tekst *początek* w miejsce `\begin{`*nazwa*`}`. Natomiast tekst *koniec* jest wstawiany w miejsce `\end{`*nazwa*`}`.

Poniższy przykład ilustruje sposób użycia instrukcji `\newenvironment`.

```
\newenvironment{zrodlo}
{Źródło: }{\par}
% następnie po \begin{document}:
\begin{zrodlo} Rocznik Statystyczny
GUS. \end{zrodlo}
```

Źródło: Rocznik Statystyczny GUS.

Znaczenie argumentu *num* jest takie samo jak w instrukcji `\newcommand`.

LaTeX nie pozwala zdefiniować środowiska o już istniejącej nazwie. W wypadku, gdy chcemy zmienić znaczenie już istniejącego środowiska, należy użyć polecenia `\renewenvironment`, mającego taką samą składnię jak `\newenvironment`.

5.1.3. Własne pakiety

W wypadku zdefiniowania wielu nowych środowisk i poleceń preambuła dokumentu może się znacznie wydłużyć. Dobrze w takiej sytuacji jest stworzyć pakiet zawierający własne instrukcje i definicje środowisk. Taki pakiet można dołączyć do dokumentu za pomocą polecenia `\usepackage`.

```
% Przykładowy pakiet ***
\ProvidesPackage{demopack}
\newcommand{\kwle}{Krótkie wprowadzenie do systemu \LaTeXe}
\newcommand{\wle}[1]{\emph{#1} wprowadzenie
do systemu \LaTeXe}
\newenvironment{zrodlo}{Źródło: }{\par}
```

Rysunek 5.1: Przykładowy pakiet

Tworzenie pakietu polega na skopiowaniu poleceń z preambuły do oddzielnego pliku o rozszerzeniu `.sty`. Na początku pakietu należy użyć polecenia:

`\ProvidesPackage{`*nazwa*`}`

Dzięki instrukcji `\ProvidesPackage` LaTeX poznaje nazwę pakietu, a to pozwoli mu na przykład ostrzec użytkownika w przypadku próby powtórnego dołączenia pakietu do dokumentu. Rysunek 5.1 przedstawia niewielki pakiet z instrukcjami z przedstawionych wyżej przykładów.

5.2. Fonty

5.2.1. Instrukcje przełączające stopień pisma

LaTeX automatycznie wybiera krój i stopień pisma dla różnych elementów dokumentu (tytuły rozdziałów, punktów, przypisy itp.). Czasami jednak zachodzi potrzeba „ręcznego” przełączenia kroju/stopnia pisma. W tym celu można skorzystać z poleceń zestawionych w tabelach 5.1 i 5.2. Stopień pisma jest kwestią układu graficznego dokumentu i zależy od wybranej klasy dokumentu i ustawienia odpowiednich opcji. W tabeli 5.3 zestawiono stopnie pisma w jednostkach absolutnych dla poleceń zmieniających wielkość kroju w przypadku standardowych klas dokumentu.

```
{\small Nieliczni lecz
\textbf{odważni} Rzymianie rządzili}
{\Large wielką \textit{Italią}.}
```

Nieliczni lecz **odważni** Rzymianie rządzili wielką *Italią*.

Bieżący font charakteryzuje w LaTeX-u pięć elementów: układ (zestaw znaków), krój (rodzinę), grubość i szerokość, odmianę, stopień i interlinię. Każdy z nich można dobrać niezależnie od ustawienia pozostałych. Oznacza to na przykład, że zmiana stopnia pisma nie powoduje zmiany kroju czy odmiany.

Tabela 5.1: Polecenia wyboru krojów i odmian

`\textrm{...}`	krój szeryfowy	`\textsf{...}`	krój bezszeryfowy
`\texttt{...}`	`grotesk`		
`\textmd{...}`	pismo jasne	`\textbf{...}`	**pismo grube**
`\textup{...}`	odmiana prosta	`\textit{...}`	*kursywa*
`\textsl{...}`	*odmiana pochyła*	`\textsc{...}`	KAPITALIKI
`\emph{...}`	*wyróżnienie*		
`\textnormal{...}`	główny font dokumentu		

Tabela 5.2: Polecenia wyboru stopnia pisma i interlinii

`\tiny`	mikroskopijny	`\Large`	większy
`\scriptsize`	bardzo mały	`\LARGE`	bardzo duży
`\footnotesize`	mniejszy		
`\small`	mały	`\huge`	ogromny
`\normalsize`	normalny	`\Huge`	największy
`\large`	duży		

Wewnątrz trybu matematycznego można wykorzystać instrukcje zmiany fontu do chwilowego opuszczenia trybu matematycznego i złożenia tekstu normalnym krojem.

Przy okazji omawiania poleceń dotyczących fontów trzeba wspomnieć o koncepcji *grupowania tekstu*. Grupa rozpoczyna się od znaku „`{`", a kończy znakiem „`}`". Grupy służą do ograniczania zasięgu działania poleceń LaTeX-a. Przyjrzyjmy się następującemu przykładowi:

```
Lubię {\LARGE duże oraz
{\small małe} litery} i~cyfry.
```

Lubię duże oraz małe litery i cyfry.

Pierwszy nawias klamrowy rozpoczyna grupę, potem polecenie `\LARGE` zmienia stopień pisma na *bardzo duży*, w którym zostanie złożony napis „duże oraz". Kolejny otwierający nawias klamrowy zaczyna następną grupę. Polecenie `\small` zmienia stopień pisma na mały. W tym stopniu złożone jest tylko słowo „małe", ponieważ nawias „`}`" zamyka grupę. Po zamknięciu grupy następuje powrót do stopnia pisma aktualnego przed jej rozpoczęciem, czyli `\LARGE`. W nim zostanie złożone słowo „litery". Zamknięcie tej grupy spowoduje, że resztę tekstu LaTeX złoży w wyjściowym stopniu pisma. Jak widać, grupy można zagnieżdżać (nawet wielokrotnie).

Tabela 5.3: Wielkość stopnia pisma w klasach standardowych

Stopień	*10pt*	*opcja 11pt*	*opcja 12pt*
`\tiny`	5pt	6pt	6pt
`\scriptsize`	7pt	8pt	8pt
`\footnotesize`	8pt	9pt	10pt
`\small`	9pt	10pt	11pt
`\normalsize`	10pt	11pt	12pt
`\large`	12pt	12pt	14pt
`\Large`	14pt	14pt	17pt
`\LARGE`	17pt	17pt	20pt
`\huge`	20pt	20pt	25pt
`\Huge`	25pt	25pt	25pt

Polecenia zmieniające stopień pisma zmieniają także interlinię. Dzieje się tak jednak tylko wtedy, gdy *przed* zamknięciem odpowiedniej grupy kończy się akapit (przez wstawienie pustego wiersza *lub* polecenia `\par`). Zwróćmy uwagę na miejsce, w którym umieszczono instrukcję `\par` w poniższych dwóch przykładach.

```
{\Large Zdanie, które ma więcej
niż pięć słów nie ma sensu!\par}
```

Zdanie, które ma więcej niż pięć słów nie ma sensu!

```
{\Large Zdanie, które ma więcej
niż pięć słów nie ma sensu!}\par
```

Zdanie, które ma więcej niż pięć słów nie ma sensu!

Gdy zachodzi konieczność zmiany stopnia pisma dla całego akapitu lub jeszcze dłuższego tekstu, możemy skorzystać ze składni przyjętej dla środowisk:

```
\begin{Large} Zdanie, które ma
więcej niż pięć słów nie ma sensu!
\end{Large}
```

Zdanie, które ma więcej niż pięć słów nie ma sensu!

Zapis taki pozwala unikać łatwych do popełnienia błędów wynikających z opuszczania nawiasów otwierających lub zamykających w grupach.

W *trybie matematycznym* w celu złożenia fragmentu wzoru innym krojem pisma można wykorzystać polecenia zestawione w tabeli 5.4.

5.2.2. Uwaga niebezpieczeństwo!

Jak już zaznaczyliśmy na początku rozdziału, nie należy *explicite* wstawiać do pliku źródłowego instrukcji zmiany fontu. Byłoby to niezgodne z pod-

Tabela 5.4: Polecenia wyboru fontów w trybie matematycznym

Polecenie	*Przykład*	*Wynik*
`\mathcal{...}`	`$\mathcal{B}=c$`	$\mathcal{B}=c$
`\mathrm{...}`	`$\mathrm{K}_2$`	K_2
`\mathbf{...}`	`$\sum x=\mathbf{v}$`	$\sum x=\mathbf{v}$
`\mathsf{...}`	`$\mathsf{G\times R}$`	$\mathsf{G\times R}$
`\mathtt{...}`	`$\mathtt{L}(b,c)$`	$\mathtt{L}(b,c)$
`\mathnormal{...}`	`$\mathnormal{R_{19}}\neq R_{19}$`	$\mathnormal{R_{19}}\neq R_{19}$
`\mathit{...}`	`$\mathit{ffi}\neq ffi$`	$\mathit{ffi}\neq ffi$

stawową ideą LaTeX-a, jaką jest oddzielenie formy od treści dokumentu i posługiwanie się formatowaniem logicznym, a nie wizualnym. Jeżeli fragment tekstu ma zostać wyróżniony poprzez złożenie go innym krojem lub stopniem pisma, to należy zdefiniować odpowiednie polecenie i jego właśnie używać w treści dokumentu.

```
% w~preambule albo pakiecie
\newcommand{\uwaga}[1]{\textbf{#1}}
% po \begin{document}
\uwaga{Baczność}. Przewody sieci
trakcyjnej są pod napięciem.
Dotknięcie grozi \uwaga{śmiercią}.
```

Baczność. Przewody sieci trakcyjnej są pod napięciem. Dotknięcie grozi **śmiercią**.

Niewątpliwą zaletą tego podejścia jest to, że kiedy później będziemy chcieli wyróżnić wszystkie elementy, na które czytelnik powinien zwrócić *szczególną uwagę*, w inny sposób niż składając je pismem półgrubym, 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.

Na zakończenie rada: nie należy przesadzać z wykorzystywaniem wielu różnych krojów pisma w dokumencie.

5.3. Odstępy

5.3.1. Zmiana wielkości interlinii

Wielkość interlinii (odstępu pomiędzy wierszami) w dokumencie można zmieniać za pomocą polecenia:

`\linespread{`*czynnik*`}`

Polecenie to należy umieścić w preambule dokumentu. Parametr *czynnik* określa wielkość interlinii. Podwójną interlinię uzyskamy za pomocą `\linespread{1.6}`. Używając `\linespread{1.3}` uzyskamy interlinię równą 1,5. Pojedynczej interlinii, zarazem domyślnej odpowiada wartość 1.

5.3.2. Wcięcia i odstępy pomiędzy akapitami

Dwa dodatkowe parametry określają, odpowiednio, wielkości wcięcia akapitowego oraz odstępu między akapitami. Na przykład, wpisując:

```
\setlength{\parindent}{0pt}
\setlength{\parskip}{1ex plus 0.5ex minus 0.2ex}
```

do preambuły dokumentu ustalamy wielkość wcięcia akapitowego na 0 pt (co powoduje, że akapity będą się zaczynać bez wcięć), a odstęp między akapitami ustalamy na 1 ex plus 0,5 ex minus 0,2 ex. Drugi zapis oznacza, że normalny odstęp między akapitami, wynoszący 1,0 ex (jednostki miary w LaTeX-u podaje tabela 5.5 na stronie 73), może się zwiększyć do $1{,}0+0{,}5 = 1{,}5$ ex lub zmniejszyć do $1{,}9 - 0{,}2 = 0{,}8$ ex. W Europie kontynentalnej akapity składa się czasami bez wcięcia akapitowego, tylko z dodatkowym odstępem między nimi. Ale uwaga, ten efekt pojawi się także w spisie treści, tabel i rysunków, gdzie poszczególne pozycje w spisie będą od siebie bardziej oddalone (w spisach większość akapitów jest jednowierszowa). Aby uniknąć tego trochę śmiesznego efektu należy usunąć powyższe instrukcje `setlenght` z preambuły dokumentu i wstawić je w części głównej, po poleceniach `\tableofcontents` itp. Najlepiej jednak wcale nie korzystać z tego sposobu, gdyż znakomita większość książek jest składana z wcięciem akapitowym, a nie z dodatkowymi odstępami między akapitami[1].

Wcięcie akapitowe wstawiamy za pomocą polecenia:

```
\indent
```

umieszczonego na początku akapitu[2]. Wstawienie `\indent`, kiedy wartość `\parindent` wynosi zero nie przyniesie oczywiście żadnego efektu.

Do złożenia akapitu bez wcięcia należy rozpocząć akapit od polecenia:

```
\noindent
```

5.3.3. Odstępy poziome

LaTeX automatycznie ustala wielkości odstępów między słowami oraz między zdaniami. Dodatkowy odstęp poziomy (przez odstęp poziomy rozumiemy

[1] Jednoczesne użycie wcięć i odstępów uważa się w Polsce za poważny błąd typograficzny.

[2] Do uzyskania efektu wstawiania wcięcia w pierwszym akapicie po tytule rozdziału, podrozdziału itd. należy dołączyć do dokumentu pakiet indentfirst z zestawu pakietów „tools”.

odstęp między wyrazami, przez odstęp pionowy rozumiemy odstęp między wierszami i akapitami) wstawiamy za pomocą polecenia:

\hspace{*odległość*}

Jeżeli taki odstęp, w wyniku złamania akapitu na wiersze, wypadnie na początku lub końcu wiersza, to zostanie usunięty – po to, by akapit nie był justowany „w chorągiewkę”. Jeżeli LaTeX ma wstawić odstęp także w wypadku, gdy rozpoczyna sie lub kończy wiersz, to zamiast `\hspace` należy użyć „gwiazdkowej” wersji `\hspace*`. Argument *odległość* oznacza wymiar LaTeX-owy. W najprostszej postaci jest to liczba oraz jednostka odległości. Wykaz ważniejszych dostępnych w LaTeX-u jednostek odległości znajduje się w tabeli 5.5.

```
To jest \hspace{1.5cm}odstęp
równy 1,5~cm.
```

To jest $\hspace{1.5cm}$ odstęp równy 1,5 cm.

Tabela 5.5: LaTeX-owe jednostki miary

mm	milimetr ≈ 1/25 cala
cm	centymetr = 10 mm
in	cal = 25,4 mm
pt	punkt ≈ 1/72 cala ≈ $\frac{1}{3}$ mm
em	w przybliżeniu szerokość „M” w bieżącym foncie
ex	w przybliżeniu wysokość „x” w bieżącym foncie

Często wygodnie jest użyć „elastycznej” odległości, zostawiając LaTeX-owi w pewnym zakresie swobodę wyboru takiej odległości, jaką uzna za najlepszą z punktu widzenia jakości składu. Taką elastyczną odległość zapisujemy następująco: *n* `plus` *p* `minus` *m*. Części „`plus` *p*” i „`minus` *m*” są opcjonalne (każdą z nich można pominąć). Tego typu odległości mają naturalną wielkość *n*, mogą się kurczyć lub rozciągać w zakresie od $n - m$ do $n + p$. Omawiany wcześniej odstęp między akapitami (`\parskip`) jest przykładem wymiaru o zmiennej wielkości. Część wymiarów LaTeX-owych może mieć zmienne wielkości, ale część musi mieć wartość stałą. Na przykład, jest zrozumiałe, że wcięcie akapitowe musi być wielkością stałą, podobnie jak szerokość i wysokość łamu.

Polecenie:

\stretch{*n*}

wstawia specjalny odstęp rozciągliwy o tej właściwości, że potrafi wypełnić całą wolną przestrzeń wiersza. Jeżeli w linii wstawimy dwa polecenia

`\hspace{\stretch{`*n*`}}`, to wielkość odstępów przez nie uzyskanych będzie wg proporcji ustalonych przez argument n. W poniższym przykładzie odstęp między x a y jest trzy razy mniejszy od odstępu między y a z.

```
x\hspace{\stretch{1}}
y\hspace{\stretch{3}}z
```

x	y	z

5.3.4. Odstępy pionowe

LaTeX wstawia odstępy pomiędzy akapitami, rozdziałami, punktami itp. automatycznie. Jeżeli istnieje potrzeba wstawienia dodatkowego odstępu pionowego, należy użyć polecenia:

`\vspace{`*odległość*`}`

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 bezpośrednio zaczynał lub kończył kolumnę tekstu), to zostanie usunięty. Jeżeli ma zostać wstawiony także na początku lub końcu strony, to należy użyć wersji „gwiazdkowej" `\vspace*`[3]. Argument *odległość* oznacza wymiar LaTeX-owy.

Polecenia `\stretch`, wykonanego łącznie z `\pagebreak`, można używać do rozmieszczania tekstu kolumny w pionie. W poniższym przykładzie tekst zostanie rozmieszczony tak, że odstęp u dołu będzie dwa razy mniejszy niż odstęp u góry strony.

```
\vspace{\stretch{1}}
Tytuł i~autor
\vspace{\stretch{2}}\pagebreak
```

Dodakowy odstęp między dwoma wierszami tego samego akapitu lub między wierszami w tabeli możemy wstawić za pomocą instrukcji:

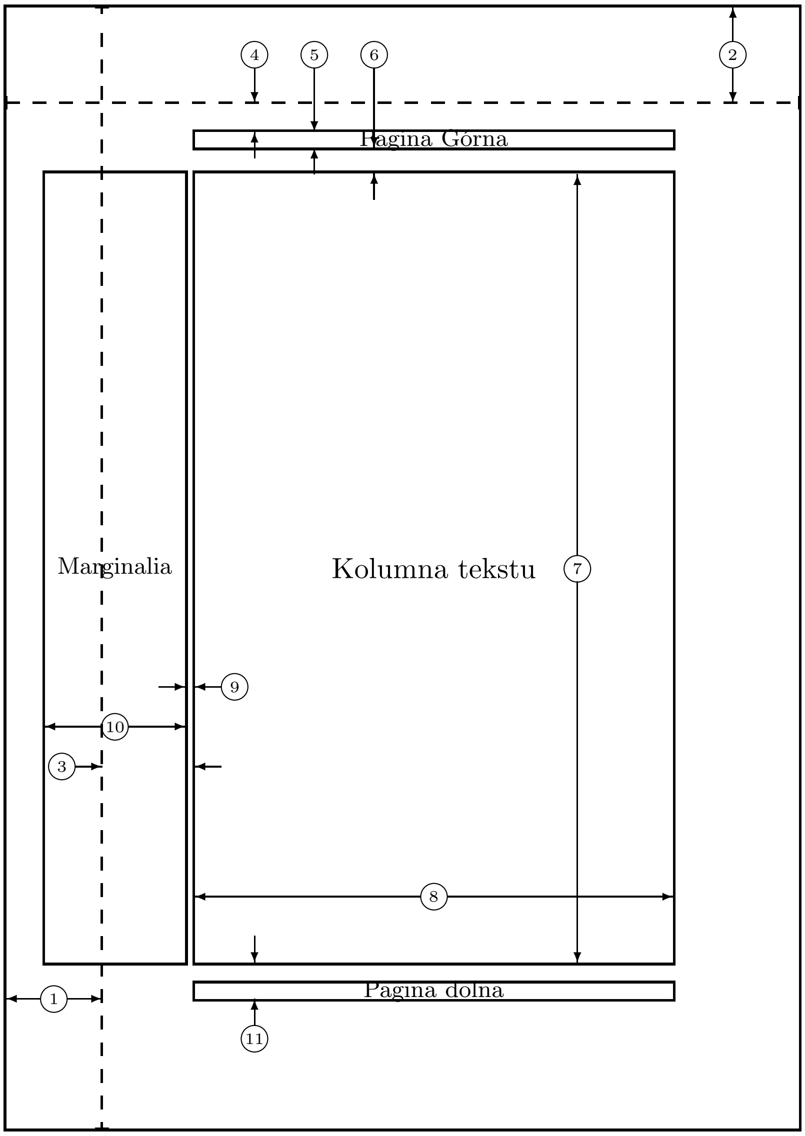
`\\[`*odległość*`]`

5.4. 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 5.2 przedstawiono wszystkie parametry graficznego układu strony. Do przygotowania rysunku użyliśmy pakietu layout z zestawu „tools"[4]. Jeżeli obliczone przez LaTeX-a wartości są z pewnych względów nieodpowiednie, można je zmienić.

[3] Zwróćmy uwagę, że w takim wypadku wysokości kolumny tekstu na sąsiednich stronach nie będzie jednakowa, zatem korzystajmy z polecenia `\vspace*` z pewną ostrożnością.

[4] `CTAN://tex-archive/macros/latex/packages/tools`.



1	`1 cal+ \hoffset`	2	`1 cal+ \voffset`
3	`\evensidemargin = 70pt`	4	`\topmargin = 22pt`
5	`\headheight = 12pt`	6	`\headsep = 19pt`
7	`\textheight = 595pt`	8	`\textwidth = 360pt`
9	`\marginparsep = 7pt`	10	`\marginparwidth = 106pt`
11	`\footskip = 27pt`		`\marginparpush = 5pt (nie pokazane)`
	`\hoffset = 0pt`		`\voffset = 0pt`
	`\paperwidth = 597pt`		`\paperheight = 845pt`

Rysunek 5.2: Parametry układu graficznego strony

Zanim jednak zaczniemy eksperymentować, zwiększając na przykład szerokość szpalty, chwilę pomyślmy. Jak dla większości rzeczy w LaTeX-u, istnieją ważne powody, dlaczego szerokość kolumny jest taka a nie inna.

W LaTeX-u mamy dwie instrukcje do zmiany wielkości wymiarów, zazwyczaj używane w obrębie preambuły dokumentu:

Pierwsza instrukcja nadaje parametrowi określoną wielkość:

`\setlength{`*parametr*`}{`*odległość*`}`

Drugie polecenie *zwiększa* wielkość parametru o określoną wielkość:

`\addtolength{`*parametr*`}{`*length*`}`

Druga instrukcja jest częściej wykorzystywana ponieważ pozwala na modyfikowanie wymiarów. Przykładowo, aby zwiększyć szerokość szpalty o jeden centymetr, należy w preambule dokumentu umieścić następujące polecenia:

```
\addtolength{\hoffset}{-0.5cm}
\addtolength{\textwidth}{1cm}
```

Wykonywanie operacji arytmetycznych na wymiarach ułatwia pakiet calc. Pakiet geometry pozwala na łatwe modyfikowanie parametrów układu graficznego strony.

5.5. Więcej o odległościach

Kiedy tylko jest to możliwe, unikajmy wykorzystywania wymiarów zdefiniowanych w jednostkach absolutnych, takich jak punkty czy milimetry. Starajmy się raczej wyrazić wymiary w odniesieniu do już istniejących, takich jak wysokość czy szerokość kolumny. W poniższym przykładzie szerokość rysunku jest zdefiniowana jako połowa szerokość bieżącej szpalty:

```
\includegraphics[width=0.5\textwidth]{test.eps}
```

Następujące trzy polecenia pozwalają określić szerokość, wysokość i głębokość *tekstu*.

`\settoheight{`*nazwa*`}{`*tekst*`}`
`\settodepth{`*nazwa*`}{`*tekst*`}`
`\settowidth{`*nazwa*`}{`*tekst*`}`

Następujący przykład pokazuje możliwe 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}%
$a$, $b$ -- przyprostokątne, \par
$c$ -- przeciwprostokątna.
\end{vardesc}
```

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

gdzie: a, b – przyprostokątne,
c – przeciwprostokątna.

5.6. Pudełka

LaTeX tworzy każdą stronę z pudełek, które odpowiednio skleja. Elementarnymi pudełkami są litery, z których sklejane są słowa. Słowa są następnie łączone w wiersze, a wiersze w akapity.

Trzeba przyznać, że powyższe jest mocno uproszczoną wersją tego, co się naprawdę dzieje, ale zasadniczo biorąc działanie LaTeX-a można wyjaśnić w terminach pudełek oraz kleju (odstępu wstawianego między pudełkami). Nie tylko litery są pudełkami. Do pudełka można włożyć praktycznie wszystko, także inne pudełka. Każde pudełko LaTeX traktuje jak pojedynczą literę.

Chciaż o tym nie mówiliśmy wprost, pudełka pojawiały się już w poprzednich rozdziałach. Na przykład polecenie `includegraphics` albo środowisko `tabular` tworzą pudełka. Oznacza to, że łatwo można zestawić dwa rysunki albo tabele obok siebie. Trzeba jedynie zadbać o to, by łączna szerokość połączonych obiektów nie była większa niż szerokość szpalty.

To samo odnosi się do akapitów, które – jeśli tego potrzebujemy – możemy składać w pudełka o zadanej szerokości:

`\parbox[`*pos*`]{`*szerokość*`}{`*tekst*`}`

Do tego samego celu można wykorzystać środowisko:

`\begin{minipage}[`*pos*`]{`*szerokość*`}` *tekst* `\end{minipage}`

Argument *pos* jest jednoliterowy i może przyjmować jedną z wartości: `c`, `t` lub `b`. Wartości te określają, jak LaTeX ma umieścić pudełko względem otaczającego tekstu. Wartość `c` oznacza umieszczenie środka wysokości pudełka na linii podstawowej, `t` – umieszczenie linii podstawowej pierwszego wiersza w pudełku na linii podstawowej otaczającego tekstu, natomiast `b` umieszczenie dolnej krawędzi pudełka na linii podstawowej. Wynik zastosowania parametrów ilustruje poniższy przykład (linię podstawową oznaczono kreską). Argument *szerokość* to wymiar określający szerokość pudełka.

```
\makebox[0pt][l]{\rule{66mm}{.4pt}}%
\parbox[c]{9mm}{5 5 5 5 5 5 5 5 5}
\parbox[t]{9mm}{6 6 6 6 6 6 6 6 6}
\parbox[b]{9mm}{8 8 8 8 8 8 8 8 8}
```

```
          8 8 8
5 5 5     8 8 8
5 5 5 6 6 6 8 8 8
5 5 5 6 6 6
      6 6 6
```

Polecenie `\parbox` składa tekst w pudełku, w razie potrzeby dzieląc tekst na linijki. Inaczej jest z pudełkami uzyskiwanymi za pomocą polecenia `\mbox`, których zawartość nigdy nie jest dzielona na wiersze. Polecenia tego używamy, gdy chcemy zapobiec dzieleniu wyrazu lub sekwencji wyrazów na wiersze. Polecenie `\mbox` jest uproszczoną wersją instrukcji `\makebox` o składni:

`\makebox[`*szerokość*`][`*pos*`]{`*tekst*`}`

Za pomocą opcjonalnego argumentu *szerokość* możemy zadać szerokość pudełka. Może ona się różnić od naturalnej szerokości tekstu w pudełku; może wynosić zero, a nawet być wielkością ujemną! W obrębie argumentu *szerokość* możemy także posługiwać się wielkościami `\width` (szerokość), `\height` (wysokość), `\depth` (głębokość) oraz `\totalheight` (suma wysokości i głębokości). Ponadto argument *pos* określa sposób umieszczenia tekstu. Litera `c` oznacza wyśrodkowanie, `l` dosunięcie do lewej, `r` dosunięcie do prawej, a `s` wyspacjowanie zawartości.

Poniższy przykład ilustruje wykorzystanie polecenia `\width` w obrębie argumentu *szerokość*. Pierwsze pudełko ma szerokość równą połowie naturalnej szerokości tekstu:

```
\makebox[.5\width][l]{oooooooo}%
\makebox{xxxxxxxx}
```

ooooxxxxxxxx

Polecenie `\framebox` działa dokładnie jak `\makebox`, z tym że naokoło pudełka kreślona jest ramka.

Oto przykład zastosowania poleceń `\makebox` i `\framebox`.

```
\makebox[\textwidth]{%
p o ś r o d k u}\par
\makebox[\textwidth][s]{%
r o z s t r z e l o n y}\par
\framebox[1.1\width]{Teraz jestem
obramowany!} \par
\framebox[0.8\width][r]{Trochę
tu jest za szeroko} \par
\framebox[1cm][l]{Nie
ma sprawy}
Da się to czytać?
```

```
                 p o ś r o d k u
r   o   z   s   t   r   z   e   l   o   n   y
Teraz jestem obramowany!
Trochę tu jest za szeroko
Nie ma sprawy Da się to czytać?
```

Pudełka można też przesuwać w pionie. Służy do tego polecenie:

```
\raisebox{przesunięcie}[wysokość][głębokość]{tekst}
```

Argument *przesunięcie* określa wielkość przesunięcia w górę (lub w dół, jeżeli wielkość przesunięcia jest ujemna). Ponadto za pomocą parametrów opcjonalnych *wysokość* oraz *głębokość* można zadać nominalną wysokość oraz głębokość pudełka (LaTeX będzie traktował pudełko tak, jakby miało zadane wymiary, bez względu na wymiary naturalne). Wewnątrz parametrów znowu można skorzystać z poleceń `\width`, `\height`, `\depth` oraz `\totalheight`.

```
\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.
```

Aaaaaaargh Krzyczała, ale nikt nie zauważył, że coś się jej przytrafiło.

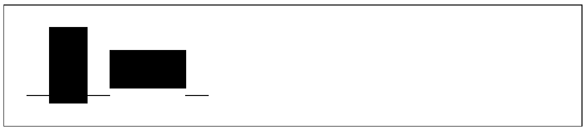
5.7. Kreski i podpory

Parę stron wcześniej pojawiło się polecenie:

```
\rule[przesunięcie]{szerokość}{wysokość}
```

W wyniku wykonania tej instrukcji najczęściej otrzymujemy w składzie czarny prostokąt:

```
\rule{3mm}{.1pt}%
\rule[-1mm]{5mm}{1cm}%
\rule{3mm}{.1pt}%
\rule[1mm]{1cm}{5mm}%
\rule{3mm}{.1pt}
```



Polecenia `\rule` używamy do rysowania kresek pionowych i poziomych. Na przykład gruba czarna krecha na stronie tytułowej niniejszego *Wprowadzenia* to wynik zadziałania instrukcji:

```
\rule[-1ex]{\textwidth}{5pt}
```

Parametr *przesunięcie* określa, jak wysoko przesunąć krechę ponad linię podstawową (lub opuścić poniżej linii podstawowej, jeżeli parametr jest ujemny).

Specjalnym przypadkiem jest kreska o zerowej szerokości, ale niezerowej wysokości. Taką kreskę nazywamy *podporą* (*strut*). Podpora to często stosowana metoda nadawania wszystkim elementom jednakowej wysokości.

Spójrzmy na poniższy przykład. Dzięki wstawieniu podpory drugi wiersz ma tę samą wysokość co pierwszy. Zawartość wiersza trzeciego jest identyczna jak drugiego, za wyjątkiem niewidocznej podpory. Gdyby jej brakowało, to wysokość drugiego wiersza byłaby mniejsza.

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

Pittprop …
Strut
Strut

Spis literatury

[1] Carlisle David P., *Packages in the 'graphics' bundle*. Dokument dostępny w zestawie pakietów „graphics" w pliku `grfguide.tex`.

[2] Goossens Michel, Mittelbach Frank, Samarin Alexander. *The LaTeX Companion*. Addison-Wesley, Reading 1994, ISBN 0-201-54199-8.

[3] Knuth Donald E., *The TeXbook*, Addison-Wesley Publishing Company 1984, ISBN 0-201-13448-9.

[4] Lamport Leslie, *LaTeX: A Document Preparation System*. 2nd ed., Addison-Wesley, Reading 1994, ISBN 0-201-52983-1.

[5] *LaTeX Local Guide*. Każda wielodostępna instalacja LaTeX-owa powinna zawierać *LaTeX 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 zwrócenie się o pomoc do lokalnego LaTeX-owego guru.

[6] LaTeX3 Project Team. *LaTeX 2ε for authors*. Dokument dostępny w pliku `usrguide.tex` w dystrybucji LaTeX 2ε.

[7] LaTeX3 Project Team. *LaTeX 2ε for Class and Package writers*. Dokument dostępny w pliku `clsguide.tex` w dystrybucji LaTeX 2ε.

[8] LaTeX3 Project Team. *LaTeX 2ε Font selection*. Dokument dostępny w pliku `fntguide.tex` w dystrybucji LaTeX 2ε.

[9] Lichoński Bogusław, *TeX na indeksie*. Biuletyn GUST 1994 (3). Dokument dostępny w `http://www.gust.org.pl/PDF/BIUL/03/02-bl.pdf`.

[10] Nowacki Janusz M., *TeXnologia a typografia*. Biuletyn GUST 1995 (6). Dokument dostępny także w `http://www.gust.org.pl/PDF/BIUL/06/01-jmn.pdf`.

[11] Reckdahl Keith, *Using EPS Graphics in LaTeX 2ε Documents* (szczegółowe wyjaśnienie jak dołączać grafikę w formacie EPS do dokumentów LaTeX-owych). Dostępny w `CTAN://tex-archive/info/epslatex.ps`.

[12] Schöpf Rainer, Raichle Bernd, Rowley Chris, *A New Implementation of LaTeX's verbatim Environments*. Dokument dostępny w zestawie pakietów „tools" w pliku `verbatim.dtx`.

[13] Williams Graham, *The TeX Catalogue* (katalog pakietów dla TeX-a oraz LaTeX-a). Dokument dostępny w `CTAN://tex-archive/help/Catalogue/catalogue.html`.

Skorowidz

\!, 46
$, 40
\', 24
', 43
\(, 40
\), 40
\,, 41, 45
\-, 18
. . . , 20
\:, 45
\;, 46
\[, 40
%, 7
\\, 15, 32, 34, 38, 74
*, 15
\], 40
^, 42
_, 42
|, 34

\a'o, 24
a5paper, 9
acute, 21
\addtolength, 76
æ, 21
akcenty matematyczne, 43
amsbsy, 50
amsfonts, 42, 57
amsmath, 45, 46, 48, 50
amssymb, 42, 51
\and, 29
\appendix, 27, 28
\arc, 25
argument, 6
— opcjonalny, 6
array, 46, 47
arraycolsep, 47
\atop, 44
\author, 29

b5paper, 9
babel, 21–26
\backmatter, 29
backslash, 5
\begin, 31
\bibitem, 60
\big, 45
\Big, 45
\bigg, 45
\Bigg, 45
\bmod, 44
\boldmath, 50
\boldsymbol, 50

calc, 76
\caption, 37
\cc, 39
\cdots, 45
center, 32
\chapter, 27
\chaptermark, 63
\choose, 44
\cite, 60
\cleardoublepage, 16, 38
\clearpage, 16, 38
\closing, 39
\ctan, 25
\ctg, 25
\ctgh, 25

\date, 29
dcolumn, 35
\ddots, 45
\depth, 78, 79

description, 31
displaymath, 40
\displaystyle, 48
doc, 10
\documentclass, 8, 9, 17
dvips, 59
\dywiz, 19

\emph, 30, 69
\end, 31
enumerate, 31
EPS, 58
eqnarray, 47
equation, 40
eucal, 57
eufrak, 57
executivepaper, 9
exscale, 10, 45

fancyhdr, 62, 63
figure, 36
fleqn, 9
flushleft, 32
flushright, 32
foiltex, 8
fontenc, 10, 23–25
fonty
— CM, 22
— EC, 22
— PL, 22
\footnote, 30
\footnotesize, 69
\frac, 44
\framebox, 78
\frenchspacing, 27
\frontmatter, 29
\fussy, 17

geometry, 76
\geq, 52
ghostscript, 58
graphicx, 59
grave, 21
grupa, 69

\height, 78, 79
\hline, 34
\hspace, 73
\hspace*, 73
\huge, 69
\Huge, 69
\hyphenation, 17, 18

ı („i” bez kropki), 21
\idotsint, 46
ifthen, 10
\iiiint, 46
\iiint, 46
\iint, 46
\include, 11
\includegraphics, 59
\includeonly, 11
indeks
— dolny, 42
— górny, 42
\indent, 72
indentfirst, 72
\index, 61, 62
\input, 12
inputenc, 10, 18, 22, 24–26
instrukcja
— \!, 46
— \', 24
— \(, 40
— \), 40
— \,, 41, 45
— \-, 18
— \:, 45
— \;, 46
— \[, 40
— \\, 15, 32, 34, 38, 74
— *, 15
— \], 40
— \a'o, 24
— \addtolength, 76
— \and, 29
— \appendix, 27, 28
— \arc, 25
— \atop, 44
— \author, 29

— \backmatter, 29
— \begin, 31
— \bibitem, 60
— \big, 45
— \Big, 45
— \bigg, 45
— \Bigg, 45
— \bmod, 44
— \boldmath, 50
— \boldsymbol, 50
— \caption, 37
— \cc, 39
— \cdots, 45
— \chapter, 27
— \chaptermark, 63
— \choose, 44
— \cite, 60
— \cleardoublepage, 16, 38
— \clearpage, 16, 38
— \closing, 39
— \ctan, 25
— \ctg, 25
— \ctgh, 25
— \date, 29
— \ddots, 45
— \depth, 78, 79
— \displaystyle, 48
— \documentclass, 8, 9, 17
— \dywiz, 19
— \emph, 30, 69
— \end, 31
— \footnote, 30
— \footnotesize, 69
— \frac, 44
— \framebox, 78
— \frenchspacing, 27
— \frontmatter, 29
— \fussy, 17
— \geq, 52
— \height, 78, 79
— \hline, 34
— \hspace, 73
— \hspace*, 73
— \huge, 69
— \Huge, 69
— \hyphenation, 17, 18
— \idotsint, 46
— \iiiint, 46
— \iiint, 46
— \iint, 46
— \include, 11
— \includegraphics, 59
— \includeonly, 11
— \indent, 72
— \index, 61, 62
— \input, 12
— \int, 44
— \label, 29
— \large, 69
— \Large, 69
— \LARGE, 69
— \ldots, 20, 45
— \left, 45
— \leftmark, 62, 63
— \leq, 52
— \linebreak, 16
— \linespread, 71
— \listoffigures, 37
— \listoftables, 37
— \mainmatter, 29
— \makebox, 78
— \makeindex, 61
— \maketitle, 29
— \mathbb, 42
— \mathbf, 71
— \mathcal, 71
— \mathit, 71
— \mathnormal, 71
— \mathrm, 71
— \mathsf, 71
— \mathtt, 71
— \mbox, 18, 20, 78
— \multicolumn, 35
— \newcommand, 66
— \newenvironment, 67
— \newline, 15, 16
— \newpage, 15
— \newtheorem, 49

— \noindent, 72
— \nolinebreak, 16
— \nopagebreak, 16
— \normalsize, 69
— \not, 52
— \opening, 39
— \overbrace, 43
— \overleftarrow, 43
— \overline, 43
— \overrightarrow, 43
— \pagebreak, 16, 74
— \pageref, 29
— \pagestyle, 11
— \par, 70
— \paragraph, 27
— \parbox, 77, 78
— \parindent, 72
— \parskip, 72
— \part, 27, 28
— \pmb, 50
— \pmod, 44
— \polecenie, 65
— \printindex, 62
— \providecommand, 67
— \ProvidesPackage, 68
— \ps, 39
— \qquad, 41, 46
— \quad, 41, 46
— \raisebox, 79
— \ref, 29
— \renewcommand, 67
— \renewenvironment, 67
— \right, 45
— \rightmark, 62, 63
— \rule, 79
— \scriptscriptstyle, 48
— \scriptsize, 69
— \scriptstyle, 48
— \section, 27
— \sectionmark, 63
— \selecthyphenation, 26
— \selectlanguage, 23, 24, 26
— \setlength, 72, 76
— \settodepth, 76
— \settoheight, 76
— \settowidth, 76
— \signature, 38
— \sloppy, 17
— \small, 69
— \sqrt, 42
— \stretch, 73, 74
— \subparagraph, 27
— \subsection, 27
— \subsectionmark, 63
— \subsubsection, 27
— \sum, 44
— \table, 36
— \tableofcontents, 28
— \tan, 25
— \textbf, 69
— \textit, 69
— \textmd, 69
— \textnormal, 69
— \textrm, 48, 69
— \textsc, 69
— \textsf, 69
— \textsl, 69
— \textstyle, 48
— \texttt, 69
— \textup, 69
— \tg, 25
— \tgh, 25
— \thispagestyle, 11
— \tiny, 69
— \title, 29
— \totalheight, 78, 79
— \underbrace, 43
— \underline, 43
— \usepackage, 9, 22, 23, 67
— \vdots, 45
— \vec, 43
— \verb, 33
— \verbatiminput, 63
— \vspace, 74
— \vspace*, 74
— \widehat, 43
— \widetilde, 43
— \width, 78, 79

\int, 44
interlinia, 71
itemize, 31

ȷ („j” bez kropki), 21
jednostki miary, 73

klasa
— article, 8
— book, 8
— letter, 8, 38
— report, 8
— slides, 8
klej, 77
Knuth, Donald E., 1
komentarz, 7
kropka, 20

\label, 29
Lamport, Leslie, 1
\large, 69
\Large, 69
\LARGE, 69
LaTeX 2.09, 2
LaTeX 2_ε, 2
LaTeX3, 2, 5
latexsym, 10
layout, 74
\ldots, 20, 45
\left, 45
\leftmark, 62, 63
legalpaper, 9
\leq, 52
leqno, 9
letter, 38
letterpaper, 9
ligatura, 20
\linebreak, 16
\linespread, 71
linia podstawowa, 45, 77, 79
\listoffigures, 37
\listoftables, 37
litera grecka, 42

\mainmatter, 29
\makebox, 78
makeidx, 10, 61
makeindex, 61
\makeindex, 61
\maketitle, 29
math, 40
\mathbb, 42
\mathbf, 71
\mathcal, 71
\mathit, 71
\mathnormal, 71
\mathrm, 71
mathrsfs, 57
\mathsf, 71
\mathtt, 71
\mbox, 18, 20, 78
minipage, 77
Mittelbach, Frank, 2
modulo, 44
\multicolumn, 35

nawias, 44
\newcommand, 66
\newenvironment, 67
\newline, 15, 16
\newpage, 15
\newtheorem, 49
\noindent, 72
\nolinebreak, 16
\nopagebreak, 16
\normalsize, 69
\not, 52

odstęp, 5
— na początku wiersza, 5
— po instrukcji, 6
— poziomy, 72
— w trybie matematycznym, 45
œ, 21
ogranicznik, 44
opcje, 8
\opening, 39
\overbrace, 43
overfull hbox, 17
\overleftarrow, 43
\overline, 43

\overrightarrow, 43

\pagebreak, 16, 74
\pageref, 29
\pagestyle, 11
pakiet, 9
— amsbsy, 50
— amsfonts, 42, 57
— amsmath, 45, 46, 48, 50
— amssymb, 42, 51
— babel, 21–26
— calc, 76
— dcolumn, 35
— doc, 10
— eucal, 57
— eufrak, 57
— exscale, 10, 45
— fancyhdr, 62, 63
— fontenc, 10, 23–25
— geometry, 76
— graphicx, 59
— ifthen, 10
— indentfirst, 72
— inputenc, 10, 18, 22, 24–26
— latexsym, 10
— layout, 74
— makeidx, 10, 61
— mathrsfs, 57
— platex, 19, 25, 44, 52
— polski, 25, 27
— showidx, 62
— syntonly, 10
— verbatim, 64
\par, 70
\paragraph, 27
\parbox, 77, 78
\parindent, 72
\parskip, 72
\part, 27, 28
pierwiastek kwadratowy, 42
platex, 19, 25, 44, 52
\pmb, 50
\pmod, 44
polecenia, 6
\polecenie, 65
polski, 25, 27
prim, 43
\printindex, 62
\providecommand, 67
\ProvidesPackage, 68
przecinek, 20
\ps, 39
pudełko, 77

\qquad, 41, 46
\quad, 41, 46
quotation, 32
quote, 32

\raisebox, 79
\ref, 29
\renewcommand, 67
\renewenvironment, 67
\right, 45
\rightmark, 62, 63
\rule, 79

\scriptscriptstyle, 48
\scriptsize, 69
\scriptstyle, 48
\section, 27
\sectionmark, 63
\selecthyphenation, 26
\selectlanguage, 23, 24, 26
\setlength, 72, 76
\settodepth, 76
\settoheight, 76
\settowidth, 76
showidx, 62
\signature, 38
\sloppy, 17
\small, 69
spójka, 20
\sqrt, 42
stopień pisma, 69
\stretch, 73, 74
\subparagraph, 27
\subsection, 27
\subsectionmark, 63

\subsubsection, 27
\sum, 44
surd, 43
syntonly, 10

środowisko, 31
— array, 46, 47
— center, 32
— description, 31
— displaymath, 40
— enumerate, 31
— eqnarray, 47
— equation, 40
— figure, 36
— flushleft, 32
— flushright, 32
— itemize, 31
— letter, 38
— math, 40
— minipage, 77
— quotation, 32
— quote, 32
— tabbing, 24
— table, 36
— tabular, 34
— thebibliography, 60
— verbatim, 33, 63
— verse, 32

tabbing, 24
\table, 36
table, 36
\tableofcontents, 28
tabular, 34
\tan, 25
\textbf, 69
\textit, 69
\textmd, 69
\textnormal, 69
\textrm, 48, 69
\textsc, 69
\textsf, 69
\textsl, 69
\textstyle, 48
\texttt, 69
\textup, 69
\tg, 25
\tgh, 25
thebibliography, 60
\thispagestyle, 11
\tiny, 69
\title, 29
\totalheight, 78, 79
tryb matematyczny, 40
twocolumn, 9

ułamek, 44
umlaut, 21
\underbrace, 43
underfull hbox, 17
\underline, 43
\usepackage, 9, 22, 23, 67

\vdots, 45
\vec, 43
\verb, 33
verbatim, 64
verbatim, 33, 63
\verbatiminput, 63
verse, 32
\vspace, 74
\vspace*, 74

w-tył-ciach, 6
wektor, 43
\widehat, 43
\widetilde, 43
\width, 78, 79
wielokropek, 45
wstawka, 35
WYSIWYG, 3, 4
wzorce podziału, 21

zalety LaTeX-a, 4
znak
— całki, 44
— sumowania, 44

żywa pagina, 62