☞ **PLAIN**

TEX na indeksie

Bogusław Lichoński

Wprowadzenie

Na pewno każdy z nas korzystał z indeksu. Niekiedy daje on nieocenione usługi, szczególnie wtedy, gdy czytamy grube i ciekawe tomisko. Niemal niemożliwe wydaje się korzystanie z TEXbiblii, czyli dzieła Donalda E. Knutha „The TEXbook", bez obszernego skorowidzu.

Indeks nazwałbym tradycyjnym oszczędzaczem czasu. Jeżeli dzieło jest czytane przez wiele osób, to z pewnością czas potrzebny na stworzenie skorowidzu będzie o wiele krótszy od łącznego czasu, jaki muszą stracić czytelnicy na odszukanie interesującego zagadnienia. Nie muszę chyba przekonywać nikogo, kto często korzysta z indeksów o ich niezwykłej przydatności. Jeśli jednak ktoś wątpi, przytoczę przykład. Posiadana przeze mnie gruba książka kucharska nie zawiera prawie ilustracji. Znalezienie w niej przepisu na jajecznicę zajęło mi 22 sekundy. Przy braku indeksu sekundy zastąpiłyby z pewnością minuty – choć z punktu widzenia prawdopodobieństwa istnieje szansa odnalezienia żądanego przepisu przy pierwszym otworzeniu książki...

Jeśli więc indeks powinien być nieodzownym elementem każdej naukowej pozycji, to dlaczego tak często jest on pomijany?

Otóż proces tworzenia indeksu nigdy nie był i nie będzie łatwy. Bardzo trudno więc znaleźć osobę, która podejmie się tej pracy. Winne takiego stanu rzeczy są także wydawnictwa, które często nie podejmują wysiłków, by tworzyć indeksy.

Pierwszym problemem jest wybór koncepcji, czyli pomysłu na skorowidz. Indeks powinien znacznie ułatwiać czytelnikowi dostęp do treści dzieła, i to w sposób najprostszy. Trudną sprawą jest więc układ logiczny haseł w indeksie oraz ich dobór. Już w tym pierwszym etapie twórca skorowidzu musi kilka razy przeczytać dzieło i zastanowić się, jak wyglądać ma skorowidz, by był zgodny z pierwotną koncepcją.

Druga część pracy, to „wychwycenie" wszystkich haseł i stron, na których te hasła występują, oraz sortowanie – praca ta wykonywana bez pomocy komutera jest przysłowiowo benedyktyńska.

O ile pierwszy z etapów jest zawsze trudny i pracochłonny, to jednak przy dużym doświadczeniu redaktora może być sprawnie wykonany; drugi etap może wykonać za człowieka maszyna, czyli komputer wraz z oprogramowaniem. Nie można pomininąć faktu, iż do rozwiązania powyższego problemu idealnie nadaje się... TEX.

Użytkownicy LATEX-a już od dawna mogą tworzyć skorowidze i sortować je przy pomocy programu *MakeIndex*. Przygotowanie indeksu wygląda następująco:

- tworzymy dokument LATEX-owy, w którym etykietujemy słowa, wybrane do skorowidzu przy pomocy komendy `\index`
- kompilujemy i otrzymujemy nieposortowany plik `.idx`, zwierający hasła w kolejności występowania w tekście
- program *MakeIndex* sortuje wejściowy plik `.idx`, tworząc posortowany plik `.ind` będący złożonym skorowidzem
- kolejna kompilacja daje w efekcie publikację z indeksem

Powyższe cztery kroki opisane są szczegółowo w rozdziale trzecim – „Zrób skorowidz".

Oczywiście w tym miejscu nasunąć może się kilka pytań, jak choćby „czy można korzystać z `plain`-TEX-a?", czy też „jak radzić sobie z tekstami polskimi?". Odpowiedzi na te pytania oraz opis jak radzić sobie z powyższymi problemami w praktyce, udzielam w tym artykule.

Indeks po polsku

W rozdziale tym opisuję zasady, według których powinien być sortowany indeks zawierający polskie wyrazy.

Okazuje się, iż brak w Polsce stuprocentowo jasnej i wyczerpującej normy określającej sposób porządkowania haseł w skorowidzach. Istnieje – moim zdaniem nie do końca wyczerpująca temat – norma PN–80/N–01223 „Szeregowanie alfabetyczne", której przedmiotem są:

... zasady szeregowania w porządku alfabetycznym w katalogach, bibliografiach, opracowaniach dokumentacyjnych i innych wykazach dokumentów oraz indeksach ...

Język polski a komputery. Pierwszym problemem mariażu języka polskiego i komputera jest kodowanie polskich znaków diakrytycznych na komputerze PC. Jak powszechnie wiadomo PC został stworzony dla potrzeb języka angielskiego i dopiero w wersji 5.0 systemu operacyjnego DOS uwzględniono języki narodowe.

Gdy komputer PC znalazł się na polskich biurkach, zaistniała potrzeba wprowadzenia standardu kodowania polskich znaków diakrytycznych. Wymyślono więc standard Mazovia, który do dziś jest u nas powszechnie stosowany. Firmy IBM i Hewlett-Packard nie przejęły się tym faktem i stworzyły standard Latin2, zawierający także wszystkie „nasze” znaki.

Jeśli chodzi o polski format MeX, to przewiduje on użycie obu standardów w dokumentach TeX-owych, więc program powinien sortować polskie wyrazy w obu standardach. Ponieważ do pewnego czasu w Polsce królowała nieprogramowalna karta Hercules, więc w plikach źródłowych `.tex` używano powszechnie notacji prefiksowej do oznaczania polskich znaków (np. ‘ą’=‘/a’). Programy sortujące indeksy po polsku powinny więc przewidywać i tę możliwość z epoki przedMeXowej.

Reguły sortowania po polsku. Reguły sortowania po polsku ujęte są w normie PN–80/N–01223 [1], która mówi, iż porządek w indeksie powinien być zgodny z porządkiem w alfabecie polskim z dodaniem liter ‘q’, ‘v’, ‘x’.

Alfabet polski z literami q, v, x.

a ą b c ć d e ę f g h i j k l ł m n

ń o ó p q r s ś t u v w x y z ź ż

Obce znaki diakrytyczne nie powinny wpływać na porządek. Chociaż w poprzednim zdaniu możnaby słowo „obce” zamienić na „wszelkie”, gdyż jak mówi norma na stronie 2:

... dopuszcza się nieuwzględnienie w szeregowaniu polskich znaków diakrytycznych ...

Nie wydaje się jednak, by taka forma sortowania była popularna i nie jest uważana powszechnie za najbardziej wskazaną. O ile mi wiadomo, żaden program systemowy nie działa według powyższej normy [1].

Porządek w polskich skorowidzach, to tak zwany porządek słowowy, czyli spacje i znaki interpunkcyjne zostaną umieszczone przed literami alfabetu. Norma niezbyt jasno o tym mówi. Na stronie 1 czytamy:

... W szeregowaniu należy brać pod uwagę alfabetyczną kolejność wyrazów, a w ich obrębie – liter. Występujące między wyrazami znaki interpunkcyjne, spacje i łączniki należy traktować jako znaki rozdzielające poszczególne wyrazy ...

Mając na względzie powyższy cytat, przyjrzyjmy się przykładowi:

```
poprawnie              nie poprawnie
- - - - -              - - - - - - -
katalog informatyczny katalogi
katalog przedmiotowy  katalog informatyczny
katalogi              katalog przedmiotowy
```

Zauważmy, że polski alfabet zawiera kilka znaków, których nie znajdziemy w żadnym innym alfabecie narodowym. Ich wystąpienie w alfabecie jest naturalne, to znaczy występują one zawsze po literze, która w znacznej części ją tworzy.

```
l « ł        L « Ł
```

(znak « oznacza tu relację porządku w słowniku polskim)

Porządek wielkich i małych liter w języku polskim jest inny niż w angielskim. Mała litera wystąpi przed wielką w dwu identyczny słowach, na przykład:

```
polski « Polski   (po polsku)
Polish « polish   (po angielsku)
```

Poruszona w ostatnim akapicie kwestia jest jednak bardzo kontrowersyjna. Pomimo wielu poszukiwań ujętej w normie jasnej definicji, nic nie udało mi się znaleźć. W poradni językowej Uniwersytetu Gdańskiego uzyskałem informację, iż precyzyjnej normy nie znajdę, jednak istnieje

1: Mimo iż w dokumentacji systemu DOS znaleźć można informację o możliwości sortowania słów w językach narodowych – przy wykorzystaniu programu systemowego `sort` – to jednak opisana możliwość nie została chyba sumiennie przetestowana przez twórców systemu, gdyż... program nie działa poprawnie.

zasada, że małe litery powinny być przed wielkimi. Gdy podałem kilka przykładów nie stosowania tej zasady z dzieł niezłych przecież wydawnictw (jak choćby „Wiedza Powszechna"), otrzymałem odpowiedź, że tylko bardzo poważne wydawnictwa przywiązują wagę do tak istotnej reguły (sic!).

Słowniki niestety często nie przestrzegają tej zasady. Jedynie bardzo poważne publikacje jak na przykład w 4-o tomowej Encyklopedii powszechnej PWN wydanie I rok 1973 na stronie xi wyraźnie wspomina się iż:

...*Za podstawę układu haseł EP PWN przyjęto kolejność liter alfabetu polskiego – od małych liter do dużych* ...

Podobnie Encyklopedia popularna PWN wydanie VI rok 1984 wydana pod redakcją Rafała Łąkowskiego, mimo iż wszystkie hasła napisane są wersalikami, uwzględnia poprawny sposób sortowania.

Niestety, niektóre wydawnictwa zapominają o normach i w sposób dowolny umieszczają hasła w słownikach, stosując najczęściej angielskie normy do porządkowania układu haseł. Jeszcze dziwniejsze wydaje się jednak stosowanie polskich norm do sortowania słów angielskich, stało się tak na przykład w angielsko-polskim i polsko-angielskim Słowniku minimum Katarzyny Billip i Zofii Chociłowskiej, wydanym przez Wiedzę Powszechną w 1968 roku, wydanie II. Na stronie 224 można znaleźć hasła `polish` i `Polish`, napisane w wymienionej właśnie kolejności...

Natomiast zupełnie niezrozumiałe dla mnie jest stosowanie w tym samym słowniku i na tej samej stronie różnych zasad porządkowania! W „Słowniku ortograficznym języka polskiego" PWN, pod redakcją Mieczysława Szymczaka, z roku 1992, wydanie 15, na stronie 612 znajdują się hasła:

```
polonez
Polonez
 ...
Polka
polka
```

Po bliższym przyjrzeniu się temu zjawisku można zauważyć pewną własność chronologiczną. Otóż starsze wydania – mniej więcej sprzed 1970 roku – stosują zasadę porządkowania dwu identycznych haseł, rozpoczynających się od litery małej lub wielkiej (w kolejności: mała, wielka), zaś wydania nowsze nie stosują żadnych zasad. Sądzę, że wszystkiemu winne są komputery i oprogramowanie zapożyczone od Anglosasów.

Sortowanie cyfr. Liczebniki główne powinny być uporządkowane według ich dosłownego znaczenia, niezależnie od tego, czy liczba jest zapisana słownie, czy jako ciąg cyfr. W tym celu użyjemy następującej komendy:

```
\index{trzynaście@13}
```

W skorowidzu pojawi się więc hasło 'trzynaście', lecz algorytm sortowania potraktuje ten napis jako liczbę '13'.

Na stronach 3 i 8 norma [1] liczebniki porządkowe powinny być uporządkowane według ich wartości, niezależnie od dosłownego systemu zapisu (arabski, rzymski). Użytkownik znów może posłużyć się znaczkiem '`@`'.

```
\index{19.I.@I.}
\index{08.vii.@vii.}
\index{27.06.@06.}
```

W języku polskim po liczebnikach porządkowych piszemy kropkę.

Zrób skorowidz

Zaawansowani użytkownicy LaTeX-a na pewno tworzyli już skorowidze i proces ich powstawania nie jest dla nich żadną tajemnicą. Dla wszystkich, którzy nie mieli tej okazji, podaję krótki przepis jak tego dokonać:

- Dodaj styl `makeidx` jako parametr komendy `\documentstyle` (ujęty w nawiasy kwadratowe np. `\documentstyle[makeidx]{article}`)
- Umieść w preambule komendę `\makeindex` tak, aby znajdował się ona pomiędzy komendami `\documenstyle` a `\begin{document}`
- Umieść w tekście komendę `\printindex` w miejscu, w którym chcesz umiejscowić skorowidz (zwykle jest to koniec dokumentu – bezpośrednio przed komendą `\end{document}`)

Zastosowanie się do powyższych trzech punktów da oczywiście skorowidz pusty, nie wyznaczyliśmy bowiem haseł, które pojawią się w indeksie. Obok wybranego słowa wpisać należy komendę `\index{`⟨*argument*⟩`}`, której argument musi zawierać słowo pojawiające się potem w skorowidzu. Dla przykładu przyjrzyjmy się „wyrwanemu" z dokumentu fragmentowi:

```
... pusta linia oznacza dla \TeX-a
koniec akapitu\index{akapit}, więc ...
```

Po takim zabiegu w skorowidzu pojawi się słowo 'akapit' wraz z numerem strony, na której słowo 'akapitu' wystąpi.

Oto krótki przykład LaTeX-owego dokumentu[2] do którego tworzy się mini-indeks o nazwie `przyklad.tex`:

```
\documentstyle[makeidx]{article}
\makeindex
\begin{document}
 Znaki końca linii traktowane są przez
 \TeX-a tak samo jak spacja\index{spacja}.
 Dowolny ciąg spacji jest traktowany
 przez \TeX-a tak samo jak jedna
 spacja\index{spacja}. Zatem nie jest
 ważne, czy odstęp pomiędzy słowami
 jest wprowadzony jako jedna, czy
 więcej spacji\index{spacja}.
 Jednakże pusta linia oznacza dla
 \TeX-a koniec akapitu\index{akapit}.
\printindex
\end{document}
```

Uznać można w tym miejscu, iż wykonano pierwszy z czterech kroków opisanych w rozdziale pierwszym.

Drugi krok, czyli kompilacja LaTeX-em, może wydać się łatwy. Jest tak w istocie, o ile posiadamy plik stylu `makeidx.sty`. Plik ten zawiera definicje makr do tworzenia indeksów i dystrybuowany jest wraz z programem *MakeIndex*. `makeidx.sty` umieścić należy w przeszukiwanym przez TeX-a katalogu[3] `...\texinput` lub katalogu bieżącym; jeśli plik ten istnieje kompilacja powinna „przejść" bez kłopotu.

```
>tex &lplain przyklad.tex
```

(znak '>' symbolizuje prompt)

Krok trzeci to sortowanie powstałego z powyższej kompilacji pliku `przyklad.idx`, który zawiera linia po linii hasła wraz z numerami stron, na których występują w kolejności występowania ich w dokumencie. Oto postać pliku `przyklad.idx`:

```
\indexentry{spacja}{1}
\indexentry{spacja}{1}
\indexentry{spacja}{1}
\indexentry{akapit}{1}
```

Hasła w plikach `.idx` o powyższej strukturze można sortować na przykład programem *MakeIndex* w następujący sposób:

```
>makeindex przyklad.idx
```

Powstanie plik o nazwie `przyklad.ind`, który jest gotowym do kompilacji uporządkowanym skorowidzem. Wprowadzany jest on do składu przez komendę `\printindex`. Oto jego postać:

```
\begin{theindex}
  \item akapit, 1
  \indexspace
  \item spacja, 1
\end{theindex}
```

Krok ostatni to kolejna kompilacja LaTeX-em. Oprócz zapisanego dokumentu pojawi się nowa strona ze skorowidzem.

Programy i makra użytkowe

W pierwszym rozdziale postawiłem sobie dwa nieco kłopotliwe pytania, na które obiecałem odpowiedzieć – co niniejszym czynię.

***MakeIndex* i *PLindex*.** Doświadczony użytkownik zauważył z pewnością, że w powyższym przykładzie kompilacja pliku `przyklad.tex` nie przebiegnie poprawnie! Stanie się tak, gdyż LaTeX nie „zna" polskich znaków i w miejscach ich wystąpienia wykaże błąd...

No właśnie – co dzieje się z indeksem, gdy zaczynamy używać LAMEX-a do składania polskich publikacji? W skorowidzu pojawić się mogą oczywiście polskie wyrazy zawierające polskie litery oraz podlegające polskim normom sortowania. W takiej sytuacji program *MakeIndex* nie wystarcza.

Ponieważ źródła programu *MakeIndex* dostępne są jako *free software*, więc spróbowałem zaimplementować *MakeIndex* dla potrzeb polskiego użytkownika. Powstał program *PLindex*, który sortuje według zasad opisanych w rozdziale drugim – „Indeks po polsku".

Program *PLindex* posiada wszystkie możliwości programu *MakeIndex* (ang. upward compatible). Szczegółowe informacje o możliwościach obu programów znaleźć można w pracach [3] oraz [4].

2: Tekst jest fragmentem książki [2].

3: Zwykle jest to katalog określony zmienną TEXINPUT, na przykład poleceniem:
`SET TEXINPUT=c:\tex\texinput`

Jedyną nową opcją dodaną do programu *PLindex* jest '-z'.

Jak używać *PLindex*. Oto przepis jak używać *PLindex* do polskich skorowidzów:

```
>plindex -z [mazovia|latin2|prefixing]
                                ... foo.idx
```

Po parametrze '-z' wpisać należy jeden z trzech podanych napisów, określających sposób kodowania polskich znaków diakrytycznych. Trzy kropki oznaczają, że istnieją inne parametry (pokrywające się znaczeniem z parametrami programu *MakeIndex*).

Oczywiste wydaje się, iż nie można sortować zbioru wyrazów, używając zasad dwóch różnych języków narodowych. Na przykład nie można sortować jednocześnie w języku polskim (opcja '-z') oraz niemieckim (opcja '-g'). Jeżeli użytkownik poda obie opcje, ostatnia z nich będzie brana pod uwagę przez program. Także opcja '-c'[4] nie jest uwzględniana w trybie sortowania po polsku.

Domyślnie (ang. default) czyli bez jakiejkolwiek opcji *PLindex* działa dokładnie tak samo jak *MakeIndex*, to znaczy sortuje zgodnie z angielskimi normami i angielskim alfabetem.

***PLindex* w biegu.** Pozwólmy teraz popracować programowi *PLindex*. Weźmy na przykład taki oto plik .idx z polskimi znakami kodowanymi w standardzie Mazovii i nazwijmy go testpl.idx

```
\indexentry{KĄT}{1}
\indexentry{kĄT}{1}
\indexentry{kąt}{1}
\indexentry{kąt}{100}
\indexentry{Kąt}{1}
\indexentry{kąT}{1}
\indexentry{KAT}{1}
\indexentry{kAT}{1}
\indexentry{kat}{1}
\indexentry{Kat}{1}
\indexentry{kaT}{1}
\indexentry{kółko}{1}
\indexentry{kolka}{1}
```

Wpiszmy teraz w linii komend polecenie:

```
>plindex -z mazovia test.idx
```

4: Użycie tej opcji powoduje znikanie pośrednich (zbytecznych) spacji. Domyślnie wszystkie spacje są pamiętane.

Po krótkiej chwili powinniśmy uzyskać rezultat w postaci pliku .ind o nazwie testpl.ind w postaci źródłowej:

\begin{theindex}	Wygląd w składzie
\item kat, 1	kat, 1
\item kaT, 1	kaT, 1
\item kAT, 1	kAT, 1
\item Kat, 1	Kat, 1
\item KAT, 1	KAT, 1
\item kąt, 1, 100	kąt, 1, 100
\item kąT, 1	kąT, 1
\item kĄT, 1	kĄT, 1
\item Kąt, 1	Kąt, 1
\item KĄT, 1	KĄT, 1
\item kolka, 1	kolka, 1
\item kółko, 1	kółko, 1
\end{theindex}	

oraz pliku-ilg[5] o nazwie testpl.ilg.

Zachęcam do przetestowania programu przy użyciu wszelkich sposobów. Osobiście testowałem program wielokrotnie, między innymi posłużyłem się zbiorem testów, jakie znalazły się w pakiecie programu *MakeIndex* w wersji 2.12. *PLindex* spisał się bez zarzutu.

IDXMAC i PLIDXMAC. Można powiedzieć, że dotychczas artykuł przeznaczony był tylko dla użytkowników LaTeX-a i nie zdziwię się, jeśli kogoś zdenerwuje ta jednostronność.

W pakiecie programu *MakeIndex* w wersji 2.12 znaleźć można plik idxmac.tex, który zawiera makra do składania skorowidzów w systemie plain-TeX. Ponieważ makra te po prostu nie działały, spróbowałem więc je poprawić i nieco zmodyfikować dla użytkownika formatu MeX.

Po przetestowaniu makr IDXMAC znalazłem w nich kilka błędów oraz niezgodności z systemem LaTeX, które usunąłem; zmiany te szczegółowo opisałem w [5].

IDXMAC z naniesionymi poprawkami i drobnymi dodatkami dołączam do pakietu *PLindex* pod nazwą PLIDXMAC w nadziei, że okażą się przydatne dla użytkowników formatu plain.

PLIDXMAC po raz pierwszy. W pakiecie programu *PLindex* znajduje się plik plidxmac.tex,

5: Plik-ilg jest odpowiednikiem pliku-log, produkowanego przez TeX-a, czyli jest zbiorem komunikatów wysyłanych przez *PLindex*.

który zawiera wszystkie definicje potrzebne do tworzenia skorowidzu makr.

Plik ten należy dołączyć do tworzonego dokumentu TEX-owego komendą:

```
\input plidxmac.tex
```

którą należy umieścić przed wywołaniem jakiejkolwiek nowej komendy, służącej do tworzenia indeksu. Może to być na przykład pierwsza linia tego dokumentu.

W tym momencie mamy do dyspozycji trzy [6] nowe komendy i mimo iż nie jest ich zbyt wiele, w zupełności wystarczą one do wygenerowania skorowidzu. Kolejność wywołania komend jest istotna; oto ich układ:

```
 ...
\makeindex
 ...
\index{⟨argument⟩}
\index{⟨argument⟩}
 ...
\index{⟨argument⟩}
 ...
\printindex
 ...
```

Aby stworzyć najprostszy skorowidz, wystarczy w pliku źródłowym (rozszerzenie .tex) umieścić komendę \makeindex. Następnie obok wybranych do indeksu słów należy wpisać komendę \index{⟨*argument*⟩}, której dokładny opis znajdziesz poniżej. Makro \index służy do etykietowania słów, które mają znaleźć się w skorowidzu.

Miejsce publikacji, w którym życzymy sobie umieścić indeks, oznaczyć należy komendą \printindex, rzecz jasna skorowidz powinien być wcześniej posortowany i złożony [4].

Jeśli weźmiemy znany już plik przyklad.tex w postaci:

```
\input plidxmac.tex
\makeindex
 Znaki końca linii traktowane są
 przez \TeX-a tak samo jak
 spacja\index{spacja}. Dowolny ciąg
 spacji jest traktowany przez \TeX-a
 tak samo jak jedna
 spacja\index{spacja}. Zatem nie jest
 ważne, czy odstęp pomiędzy
 słowami jest wprowadzony jako jedna,
 czy więcej spacji\index{spacja}.
 Jednakże pusta linia oznacza dla
 \TeX-a koniec akapitu\index{akapit}.
\printindex
```

to plik przyklad.idx należy posortować wydając polecenie:

```
>plindex -z Mazovia -s plaintex.ist
                            przyklad.idx
```

Zagadkowe może tu być pojawienie się pliku plaintex.ist. Dokładnie opisałem tę systuację w dokumentacji do programu *PLindex* [4] zaznaczę w tym miejscu, iż plik ten, to tak zwany plik stylu i powinien znajdować się w kartotece bieżącej lub kartotece określonej przez zmienną systemową INDEXSTYLE (przykładowo SET INDEXSTYLE=c:\tex\ist), oraz przedstawię jego zawartość:

```
preamble
"\\beginindex\n"
postamble
"\n\n\\endindex\n"
```

Posortowany plik przyklad.ind jest wspominanym już i gotowym do kompilacji uporządkowanym skorowidzem, który wprowadzany jest do składu komendą \printindex. Oto jego postać:

```
\beginindex
\item akapit, 1
\indexspace
\item spacja, 1
\endindex
```

Indeks z pauzami. Ponieważ niektóre dzieła posiadają indeksy, w których zamiast wcięć pojawiają się pauzy, dlatego zdefiniowałem makro \indexemdashes, które wykonuje tę czynność. Makro to powinno pojawić się w pliku źródłowym przed makrem \printindex.

Co jeszcze potrafi PLIDXMAC. W rozdziale „PLIDXMAC po raz pierwszy” wspomniałem o innych możliwościach, jakie niosą z sobą makra PLIDXMAC.

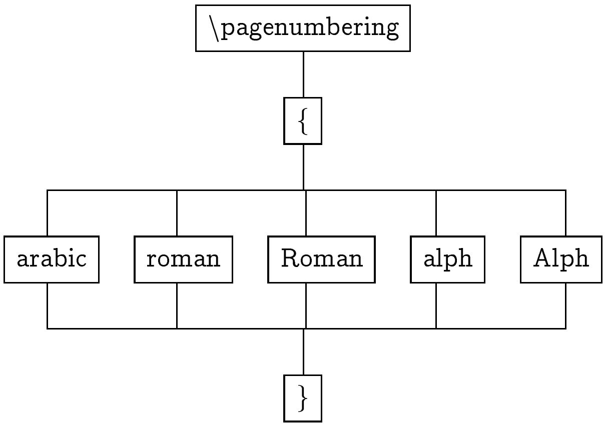
Pierwszą z nich jest wybór konwencji zapisu numerów stron. Domyślnie przy hasłach pojawiają się cyfry arabskie, można także numerować wielkimi i małymi cyframi rzymskimi,

6: W rzeczywistości jest ich więcej; patrz rozdział „Co jeszcze potrafi PLIDXMAC”.

a także wielkimi i małymi literami. Aby uzyskać któryś z opisanych efektów, wystarczy komendę `\makeindex` poprzedzić komendą:

`\pagenumbering{`⟨*argument*⟩`}`, gdzie

⟨*argument*⟩ ⟶ `arabic` | `roman` | `Roman` | `alph` | `Alph`



Każdy, kto obok skorowidzu chciałby stworzyć glosarium[7], może dokonać tego w identyczny sposób, jak tworzy się indeks. Wystarczy użyć komend `\makeglossary` oraz `\glossary`, które odpowiadają makrom `\makeindex` i `\index`. Powstaje wówczas plik z rozszerzeniem `.glo`.

Umieszczenie w dokumencie sekwencji:

```
\begindoublecolumns
\printindex
\enddoublecolumns
```

spowoduje złożenie skorowidzu w dwu szpaltach wyrównanych do góry, zaś umieściwszy po komendzie `\begindoublecolumns`; `\balancecolumns` wyrównanych do dołu.

Zakończenie

Tworzenie skorowidzów na pewno nie jest i nie będzie sprawą łatwą. Niektóre systemy – na przykład system WEB – wspomagają ten proces, tworząc automatycznie część indeksu. Jest to jeden z przykładów, gdzie komputer wykonuje cząstkę pracy koncepcyjnej.

Rzeczywistą trudnością jest dobór kryteriów tworzenia skorowidzu dla danego typu dzieła. Można zdecydować się na stworzenie kilku indeksów w obrębie jednej publikacji, tworzyć indeksy po każdym rozdziale czy tomie. Możliwości jest na prawdę wiele. Można śmiało stwierdzić, że stworzenie odpowiedniego skorowidzu „dopasowanego” do publikacji jest swoistą sztuką.

Wiele osób w naszym kraju, posługujących się TeX-em jako systemem składu, tworzy mniej lub bardziej obszerne publikacje naukowe, zapominając o skorowidzach lub nie tworząc ich z braku odpowiednich narzędzi. Mam nadzieję, że *PLindex* i zbiór makr PLIDXMAC obudzą w nas potrzebę ich tworzenia przydając się wielu użytkownikom.

Dostęp

Program *MakeIndex* dostępny jest np. w zasobach komputerów CTAN. Program *PLindex* i makra PLIDXMAC dostępne są na komputerze `ftp.pg.gda.pl` z konta ‘anonymous’.

Bibliografia

[1] Norma PN–80/N–01223 „Szeregowanie alfabetyczne”, Maj 1980. Norma zgłoszona przez Ministerstwo Kultury i Sztuki. Ustanowiona przez Polski Komitet Miar i Jakości.

[2] Leslie Lamport. *LaTeX: System przygotowywania dokumentów*. Ariel, 1992. Tłumaczenie polskie: Piotr Wyrostek.

[3] Leslie Lamport. *MakeIndex*: An Index Processor For LaTeX. Documentation for general *MakeIndex* users, Luty 1987.

[3] Bogusław Lichoński. *PLindex* polska implementacja programu *MakeIndex*. Dokumentacja dla użytkowników i programistów, Wrzesień 1993.

[5] Bogusław Lichoński. PLIDXMAC makra do tworzenia skorowidzów w systemie plain-TeX. Dokumentacja dla użytkowników i programistów, Październik 1993.

[6] Pehong Chen i Michael A. Harrison. Index Preparation and Processing. Computer Science Division, Marzec 1987.

7: glosarium – od łac. *glossarium* zbiór glos od łac. *glossa* (stary, obcy) wyraz wymagający wyjaśnienia; słowni(cze)k wyrazów trudnich, rzadkich, przestarzałych, wymagających objaśnienia.