1.12 — metody komputerowe,
języknawstwo,
leksykografia

A. Ziabicki

DWUSTOPNIOWE PORZĄDKOWANIE HASEŁ LEKSYKOGRAFICZNYCH W RÓŻNYCH JĘZYKACH

47/1990

WARSZAWA 1990

ISSN 0208-5658

Praca wpłynęła do Redakcji dnia 20 sierpnia 1990 r.

IPPT PAN

N a p r a w a c h r ę k o p i s u

Instytut Podstawowych Problemów Tecniki PAN
Nakład 100 egz. Ark.wyd.2,5 Ark.druk.3,5
Oddano do drukarni w grudniu 1990 r.
Nr zamówienia 13/91

Warszawska Drukarnia Naukowa, Warszawa,
ul. Śniadeckich 8

Andrzej Ziabicki
Instytut Podstawowych Problemów
Techniki PAN, Warszawa

DWUSTOPNIOWE PORZĄDKOWANIE HASEŁ LEKSYKOGRAFICZNYCH W RÓŻNYCH JĘZYKACH

W oparciu o teorię łańcuchów alfanumerycznych[1] opracowano standardowy program pozwalający na porządkowanie leksykograficzne wg wskazanego alfabetu i wtórne wg znaków diakrytycznych. Dla każdego języka, dla którego program ma być stosowany, należy określić zbiór i porządek samodzielnych jednostek alfabetycznych (liter, czasem dwuznaków traktowanych jak oddzielne litery), a także hierarchię znaków diakrytycznych (akcentów, znaków przegłosu, iloczasu itp). Przedyskutowano przykłady zasad porządkowania w kilku językach europejskich oraz realizację programu w języku FoxBase.

1. WSTĘP

W czasach komputerowego przetwarzania tekstów ustalenie ścisłych reguł porządkowania haseł (wyrazów, ciągów znaków graficznych) nabiera szczególnego znaczenia. Zarówno automatyczne sporządzanie słowników i indeksów, jak i komputerowa analiza tekstów połączona z wyszukiwaniem haseł w słowniku wymaga dokładnej i jednoznacznej zasady porządkowania, czyli kolejności w jakiej różne hasła powinny występować w uporządkowanych zbiorach.

Tradycyjna leksykografia oparta na pracy ręcznej w znacznej mierze polegała na doświadczeniu i "wyczuciu", co pozostawiało autorom słowników pewną swobodę w ustawianiu haseł. Brak sformalizowanych zasad porządkowania, zwłaszcza w odniesieniu do dwuznaków oraz liter ze znakami diakrytycznymi, prowadzi do

błędów w lokalizacji haseł. Przykłady takie podamy również w tej pracy. Zasady ortografii i porządkowania wyrazów w różnych językach ulegają zmianom i określenie aktualnych, jednoznacznych reguł zapisu i porządkowania wydaje się niezbędne.

W poprzedniej pracy[1] omówiliśmy teorię porządkowania łańcuchów alfanumerycznych, która może stanowić formalną podstawę porządkowania haseł leksykograficznych w słownikach lub skorowidzach. W językach nie stosujących znaków diakrytycznych (np. język polski) uporządkowanie haseł jest jednostopniowe, leksykograficzne, z uwzględnieniem właściwego alfabetu. W językach, w których występują akcenty, znaki iloczasu, przegłosu, dierezy, itp., uporządkowanie haseł jest dwustopniowe: pierwszy stopień to uporządkowanie leksykograficzne bez uwzględnienia znaków diakrytycznych, drugi stopień - wtórne uporządkowanie według znaków diakrytycznych. Podstawy takiego uporządkowania omówiliśmy poprzednio[1]. Obecnie dokonamy analizy systemów porządkowania w różnych językach europejskich i podamy schemat programu komputerowego realizującego dwustopniowe uporządkowanie leksykograficzne ze znakami diakrytycznymi w dowolnym języku posługującym się alfabetem nie przekraczającym 94 znaków.

2. UPORZĄDKOWANIE LEKSYKOGRAFICZNE

Podstawą uporządkowania leksykograficznego jest przypisanie każdej jednostce alfabetycznej (literze, kombinacji liter) określonego kodu liczbowego odpowiadającego położeniu tej jednostki w alfabecie. W przypadku standardowego alfabetu łacińskiego stosowanego w wielu językach europejskich (angielski,

francuski, włoski, portugalski) jednostkami alfabetycznymi są po prostu litery. W innych językach odrębnymi jednostkami alfabetycznymi o określonym położeniu w alfabecie mogą być również pary liter ('ch' i 'll' w hiszpańskim, 'gy', 'ny' itp. w węgierskim), modyfikacje liter alfabetu łacińskiego ('ą', 'ę', 'ć', 'ś' w polskim), a także odmienne znaki graficzne właściwe danemu językowi (grecki, rosyjski). W informatyce rozpowszechniony jest szeroko system siedmio-bitowych kodów ASCII (*American Standard Code for Information Interchange*) obejmujący 26 dużych i 26 małych liter standardowego alfabetu łacińskiego, cyfry, znaki arytmetyczne i przestankowe, kilka znaków graficznych ('@', '#', '$', '^', '&', '*', ':', '\', '~,'), a także 34 kody kontrolne do sterowania komputerem. Kody te pokazano w Tabeli I. Uporządkowanie leksykograficzne haseł (wyrazów) zapisanych standardowym alfabetem łacińskim polega na tym, że każdemu hasłu zbudowanemu z L liter przypisuje się liczbę charakterystyczną, V, sumując wartości kodów ASCII kolejnych liter pomnożonych przez potęgi ułamka (1/128):

$$(1) \qquad V = V_I = \sum_{k=1}^{L} ASC(Z_k) \cdot (128)^{1-k}$$

k oznacza położenie znaku w haśle, liczone od lewej. Liczba V jednoznacznie charakteryzuje hasło. Oznaczenie V_I odzwierciedla fakt, że uporządkowanie leksykograficzne dotyczy pierwszego stopnia uporządkowania.

Tabela I.

Standardowe "drukowalne" kody ASCII

Kod ASC(Z) hex	Kod ASC(Z) dec	Znak Z	Kod ASC(Z) hex	Kod ASC(Z) dec	Znak Z
20	32	SPC	3B	59	;
21	33	!	3C	60	<
22	34	"	3D	61	=
23	35	#	3E	62	>
24	36	$	3F	63	?
25	37	%	40	64	@
26	38	&	41	65	A
27	39	'	42	66	B
28	40	(	43	67	C
29	41	)	44	68	D
2A	42	*	45	69	E
2B	43	+	46	70	F
2C	44	,	47	71	G
2D	45	-	48	72	H
2E	46	.	49	73	I
2F	47	/	4A	74	J
30	48	0	4B	75	K
31	49	1	4C	76	L
32	50	2	4D	77	M
33	51	3	4E	78	N
34	52	4	4F	79	O
35	53	5	50	80	P
36	54	6	51	81	Q
37	55	7	52	82	R
38	56	8	53	83	S
39	57	9	54	84	T
3A	58	:	55	85	U

Tabela I. (c.d.)

Standardowe "drukowalne" kody ASCII

Kod ASC(Z) hex	Kod ASC(Z) dec	Znak Z	Kod ASC(Z) hex	Kod ASC(Z) dec	Znak Z
56	86	V	6B	107	k
57	87	W	6C	108	l
58	88	X	6D	109	m
59	89	Y	6E	110	n
5A	90	Z	6F	111	o
5B	91	[	70	112	p
5C	92	\	71	113	q
5D	93	]	72	114	r
5E	94	^	73	115	s
5F	95	_	74	116	t
60	96	`	75	117	u
61	97	a	76	118	v
62	98	b	77	119	w
63	99	c	78	120	x
64	100	d	79	121	y
65	101	e	7A	122	z
66	102	f	7B	123	{
67	103	g	7C	124	|
68	104	h	7D	125	}
69	105	i	7E	126	~
6A	106	j	7F	127	DEL

Relacje pomiędzy dwoma hasłami H_1 i H_2 są równoważne relacjom pomiędzy odpowiednimi liczbami charakterystycznymi[1]

$$(2) \qquad \begin{aligned} V(H_1) = V(H_2) &\iff H_1 = H_2 \\ V(H_1) < V(H_2) &\iff H_1 < H_2 \end{aligned}$$

Uporządkowanie haseł w porządku wzrastających liczb charakterystycznych V nazywa się uporządkowaniem leksykograficznym prostym.

Matematyczny sens uporządkowania leksykograficznego jest następujący: każdy znak ASCII (literę, cyfrę, znak przestankowy) traktuje się jak cyfrę w układzie pozycyjnym o podstawie 128. Ciąg znaków (hasło, wyraz) jest liczbą zapisaną w takim

układzie. Pierwszy znak (litera) ciągu określa część całkowitą, dalsze znaki - część ułamkową liczby charakteryzującej hasło. Traktując hasło jako "liczbę" musielibyśmy po pierwszym znaku (literze) postawić "kropkę dziesiętną" (ściślej: "kropkę 128-kową").

Porządkowanie leksykograficzne według innych alfabetów niż 26-literowy alfabet łaciński wymaga wprowadzenia dla niestandardowych jednostek alfabetycznych zastępczych znaków (kodów) ASCII. Jeżeli w rozpatrywanym alfabecie występuje więcej jednostek alfabetycznych niż 26, to niektóre z nich trzeba zastąpić nieliterowymi znakami ASCII. Często dla jednego systemu alfabetycznego tworzy się dwa systemy znaków łacińskich.

Pierwszy system, łatwy do rozpoznania i zapamiętania, służy do zapisywania tekstów w języku obcym standardowym alfabetem łacińskim. Transkrypcja taka najczęściej oparta jest na zasadach fonetycznych lub graficznym podobieństwie znaków obcego alfabetu i standardowych znaków ASCII. Transkrypcja jednostek alfabetycznych obcego języka (pojedynczych liter, dwuznaków stanowiących samodzielne jednostki alfabetyczne, polega na przyporządkowaniu im albo pojedynczych liter (znaków ASCII), albo kombinacji dwóch (rzadziej kilku) liter (znaków). W języku polskim np. pojedyncze litery 'ć, .'ń', 'ó', 'ś' i 'ź' wygodnie jest zapisywać jako dwuznaki: (c'), (n'), (o'), (s') i (z'). Czasem możliwa jest transliteracja, czyli przyporządkowanie każdej literze obcego alfabetu jednej (i tylko jednej) litery alfabetu łacińskiego. W rozdziałach 4 i 5 podamy przykłady transkrypcji dla różnych języków.

3. UPORZĄDKOWANIE WEDŁUG ZNAKÓW DIAKRYTYCZNYCH

W poprzedniej pracy[1] omówiliśmy zasadę porządkowania odmienną od uporządkowania leksykograficznego. Występujące w wielu językach akcenty, znaki iloczasu, przegłosu lub dierezy, zwane znakami diakrytycznymi, wpływają na uporządkowanie haseł w inny sposób, niż odmienne jednostki alfabetyczne. Znakom diakrytycznym przypisuje się również pewne liczby charakterystyczne, lub kody, D, zwane rangami, określające ich wzajemne relacje oraz relacje względem liter niemodyfikowanych.

Jeżeli przy jednej literze może wystąpić tylko jeden rodzaj znaku diakrytycznego, to dla wszystkich znaków można przyjąć tę samą rangę, D, względem znaku 'pustego' (tzn. liter bez znaków diakrytycznych). Jeżeli litery ze znakiem diakrytycznym występują po literach bez znaku, to ranga znaku jest dodatnia względem znaku 'pustego' i można przyjąć np.

D('znak pusty') = '0'

D('dowolny znak diakrytyczny') = '1'

Gdy litery ze znakiem diakrytycznym poprzedzają litery bez znaku (jak to ma miejsce w akcentowanych i nieakcentowanych samogłoskach w języku nowogreckim, rozdział 4.3) ranga znaku jest ujemna. Aby uniknąć liczb ujemnych, zamiast D='0' dla znaku 'pustego' i D='-1' dla akcentu (´) wygodnie jest przyjąć

D('znak pusty') = '1'

D(´) = '0'

W przypadku wielu różnych znaków pojawiających się przy tych samych literach sprawa jest bardziej skomplikowana. Np. w

języku francuskim samogłoska 'e' pojawia się w czterech modyfikacjach: bez znaku diakrytycznego ('e'), oraz z akcentami aigu ('é'), grave ('è') i circonflexe ('ê'). W rozdziale 4.6 pokażemy, że znaki te mają następującą hierarchię

(3) (znak 'pusty') < (^) < (`) < (´)

Znakami '<' i '>' będziemy oznaczać porządek znaków i haseł. "a < ą" znaczy: "a poprzedza ą", a "ż > ź": "ż następuje po ź". W podobny sposób znakiem '<=>' będziemy oznaczać równoważność jednostek alfabetycznych pod względem leksykograficznym. Jednostki takie mogą się różnić postacią graficzną i obecnością znaków diakrytycznych. Znakiem '=>' będziemy oznaczać przekształcenie lub transkrypcję znaku lub grupy znaków.

Przy obecności wielu znaków diakrytycznych musimy określić ich hierarchię. Zgodnie z nierównościami (3) możemy np. przyjąć

(3a)
D('pusty') = '0' D(`) = '2'
D(^) = '1' D(´) = '3'

Znaki diakrytyczne określają uporządkowanie dodatkowe, wtórne, nakładające się na pierwotne uporządkowanie leksykograficzne. Aby zilustrować różnicę pomiędzy uporządkowaniem leksykograficznym i uporządkowaniem według znaków diakrytycznych wyobraźmy sobie hipotetyczną sytuację, w której modyfikacje litery 'e' stanowią odrębne jednostki alfabetyczne (litery), o takiej samej kolejności, co rangi odpowiednich znaków diakrytycznych (nierówności 3)

(3b) e < ê < è < é

Przy takiej kolejności "liter" 'e', ..., 'ó', hasła leksykograficzne uporządkowane byłyby jak następuje

(4) ea < ez < êa < êz < èa < èz < éa < éz

W szczególności, samogłoska 'ê' pojawiłaby się dopiero po wyczerpaniu wszystkich haseł z samogłoską nieakcentowaną 'e', a więc po 'ez', samogłoska 'è' po 'êz, 'é' po 'èz', itd. Poprawne uporządkowanie haseł francuskich uzyskuje się w inny sposób. Najpierw porządkuje się hasła leksykograficznie ignorując znaki diakrytyczne, czyli przyjmując równoważność różnie akcentowanych samogłosek:

(5) e <=> ê <=> è <=> é

Ostateczny porządek haseł uzyskuje się przez uporządkowanie wtórne według znaków diakrytycznych tylko haseł identycznych pod względem leksykograficznym.

Dwustopniowo uporządkowanemu hasłu o długości L ze znakami diakrytycznymi odpowiada liczba charakterystyczna

$$(6) \qquad V = \sum_{k=1}^{L} ASC(Z_k)\cdot(128)^{1-k} + 32\cdot(128)^{-L} + \sum_{k=L+1}^{2L} D(Z_{k-L})\cdot(128)^{-k}$$

Pierwszy wyraz we wzorze (6) odpowiada uporządkowaniu leksykograficznemu i zawiera kody ASCII poszczególnych liter. Drugi wyraz (spacja, ASCII = 32) stanowi separator oddzielający część leksykograficzną od diakrytycznej. Ostatni wyraz opisuje uporządkowanie wtórne według znaków diakrytycznych i zależy jedynie od rang, D, znaków pojawiających się przy kolejnych

znakach, Z_k. Pełna liczba charakterystyczna V stanowi podstawę uporządkowania haseł.

Uporządkowanie dwustopniowe można zrealizować tworząc tzw. rozszerzone łańcuchy alfanumeryczne[1]. Dowolne L-literowe hasło ze znakami diakrytycznymi można przedstawić w postaci rozszerzonego łańcucha alfanumerycznego o długości 2L+1:

$$Z_1 Z_2 Z_3 \ldots Z_L \_ D_1 D_2 \ldots D_L$$

Pierwsze L znaków tego łańcucha stanowią kolejne litery hasła

$$Z_1, Z_2, \ldots Z_L.$$

Końcowa część łańcucha rozszerzonego (również o długości L), zbudowana jest z kodów rangi znaków diakrytycznych

$$D(Z_1), D(Z_2), \ldots D(Z_L)$$

Spacja oznaczona kreską (_) jest separatorem oddzielającym część alfabetyczną od diakrytycznej. Oddzielenie części alfabetycznej od diakrytycznej separatorem o najniższej wartości ASCII (ASC=32) jest konieczne, jeżeli kody rangi znaków diakrytycznych (np. cyfry) pojawiają się w samym haśle. W poprzedniej pracy[1] na temat wielostopniowego porządkowania łańcuchów alfanumerycznych pomijaliśmy separatory, co może prowadzić do błędów. Dwustopniowe uporządkowanie ze znakami diakrytycznymi będące podstawą uporządkowania w różnych językach realizuje się automatycznie przez leksykograficzne uporządkowanie łańcuchów rozszerzonych.

Rozważmy dla przykładu 3 hasła francuskie: "élève" (uczeń),

"élève1" (gdzie cyfra '1' stanowi integralną część hasła), oraz "élevé" (podniesiony, wychowany). Wykorzystując rangi znaków diakrytycznych ze wzoru (3a), otrzymujemy łańcuchy rozszerzone

```
élève   =>  eleve_30200
élève1  =>  eleve1_302000
élevé   =>  eleve_30300
```

wyznaczający porządek:

élève < élevé < élève1

Pominięcie w łańcuchach rozszerzonych separatora (_)

```
élève   =>  eleve30200
élève1  =>  eleve1302000
élevé   =>  eleve30300
```

prowadzi do niejednoznaczności hasła "élève1" i wyznacza nieprawidłowy porządek:

élève1 < élève < élevé

Tabela II podaje przykłady dwustopniowego uporządkowania haseł francuskich. W I kolumnie podano hipotetyczne uporządkowanie jednostopniowe, jakie miałoby miejsce gdyby różnie akcentowane litery stanowiły odrębne jednostki alfabetyczne (nierówności 3b). Kolejne etapy uporządkowania dwustopniowego pokazują kolumny II i III, a ostatnia kolumna zawiera "łańcuchy rozszerzone". Leksykograficzne uporządkowanie takich łańcuchów określa poprawny porządek haseł ze znakami diakrytycznymi.

Łańcuchy rozszerzone haseł nie zawierających żadnych znaków diakrytycznych - w wyrazach *elle, emballage, exulter* - różnią się od postaci wyjściowej tylko ciągiem zer. W wyrazach

akcentowanych relacje uporządkowania wyznaczają zarówno rangi znaków diakrytycznych, jak i ich położenia w wyrazie.

Tabela II.

Uporządkowanie haseł francuskich ze znakami diakrytycznymi[3]

Hipotetyczne uporządkowanie jednostopniowe	Uporządkowanie dwustopniowe I etap	II etap	Łańcuchy rozszerzone
elle (ona)	elan	élan	elan_3000
emballage (opakowanie)	eleve	élevé	eleve_30003
exulter (nie posiadać się z radości)	eleve	élève	eleve_30200
être (być)	elite	élite	elite_30000
ère (era)	elle	elle	elle_0000
élan (rozpęd)	elude	élude	elude_30000
élevé (wychowany)	elude	éludé	elude_30003
élève (uczeń)	emballage	emballage	emballage_000000000
élite (elita)	ere	ère	ere_200
élude (omijam)	etre	être	etre_1000
éludé (ominięty)	exulter	exulter	exulter_0000000

4. ZASADY PORZĄDKOWANIA HASEŁ W RÓŻNYCH JĘZYKACH

Opierając się na wyżej omówionych zasadach porównamy systemy porządkowania haseł w kilku językach europejskich. W tym celu należy najpierw zidentyfikować wszystkie jednostki alfabetyczne oraz znaki diakrytyczne i ich rangi. Jednostkom alfabetycznym

należy przyporządkować kody ASCII, czyli 'cyfry' w 128-kowym układzie pozycyjnym, w takiej kolejności, w jakiej występują w alfabecie danego języka. Jednostki alfabetyczne nie muszą być odrębnymi znakami graficznymi. Mogą to być np. dwuznaki ('ll' i 'ch' w języku hiszpańskim, 'cs', 'gy', w węgierskim, itd.) lub modyfikacje liter, stanowiących w postaci niezmodyfikowanej odrębne jednostki ('ō' i 'ū' obok 'o' i 'u' w języku węgierskim, 'ş' i 'ţ' obok 's' i 't' w rumuńskim, ć, ń, ś, ó, ź w polskim itd). Różne znaki graficzne mogą stanowić tę samą jednostkę alfabetyczną (*sigma*, σ, i *stigma*, ς, w języku greckim, 'ȳ' i 'ij', w niderlandzkim, ligatury 'æ' i 'ae' oraz 'œ' i 'oe' w angielskim i francuskim, 'ß' i 'ss' w niemieckim). Ten sam znak graficzny może mieć różny charakter leksykograficzny w różnych językach: ō stanowi odrębną literę w węgierskim i szwedzkim, lecz jedynie modyfikację samogłoski 'o' znakiem diakrytycznym przegłosu w języku niemieckim. Autorzy słowników czasem nie zdają sobie sprawy z różnic pomiędzy znakiem graficznym i jednostką alfabetyczną, co prowadzi do niekonsekwentnego uporządkowania haseł.

W językach posługujących się zmodyfikowanym alfabetem łacińskim zachowano pełny zestaw 26 liter, nawet jeśli zapis wyrazów rodzimego pochodzenia tego nie wymaga.

4.1 Język angielski[4]

Współczesny język angielski posługuje się 26-literowym alfabetem łacińskim

abcdefghijklmnopqrstuvwxyz

W wyrazach pochodzenia francuskiego sporadycznie pojawiają się akcenty (np. étagère, passé, congé, rôle), a także znak dierezy (np. naïve, coöpt).

W dawniejszych tekstach angielskich szereg wyrazów pochodzenia łacińskiego lub greckiego zapisywano ligaturami 'æ' i 'œ' stanowiącymi skróty dwuznaków 'ae' i 'oe'. Współcześnie znak 'æ' zastępuje się dwuznakiem 'ae' lub samogłoską 'e', a znak 'œ' dwuznakiem 'oe' lub samogłoską 'e':

anæmia (*anaemia*)	mediæval (*mediaeval*)
æsthetic (*aesetetic*)	œcology (*ecology*)
ætiology (*aetiology, etiology*)	fœtus (*foetus, fetus*)
cænozoic (*cainozoic*)	manœuvre (*manoeuvre*)
cæcum (*caecum, cecum*)	subpœna (*subpoena*).

W słownikach, w których zachowano pisownię z æ i œ (np. OED[4]) ligatury æ i œ są porządkowane jak dwuznaki 'ae' i 'oe'. Tak więc mamy:

'cadmium' < 'cæcum' < 'cage'

'Odyssey'< 'œcology' < 'of'

W wielu słownikach (np. Wylda[4]) ligatury już się nie pojawiają. Podstawą uporządkowania haseł angielskich jest uporządkowanie leksykograficzne.

4.2. Język polski[5]

Alfabet polski obejmuje 35 liter. Zachowujemy 26 liter alfabetu łacińskiego (łącznie z 'q', 'v' i 'x', służącymi do zapisu wyrazów obcojęzycznych) i wprowadzamy 9 dodatkowych liter otrzymując alfabet

aąbcćdeęfghijklłmnńoópqrsśtuvwxyzźż

We współczesnej ortografii polskiej wszystkie litery (35) stanowią równoprawne jednostki alfabetyczne, którym należy przyporządkować znaki zastępcze (kody) ASCII. Do zapisu tekstów standardowym alfabetem łacińskim, dodatkowe litery polskie można zamienić na nieliterowe znaki ASCII, lub zastosować transkrypcją dwuliterową, np.

ę => (e'), ć => (c'), ... itd.

W uporządkowanym zbiorze haseł każda z 35 liter pojawia się dopiero po wyczerpaniu wszystkich możliwych kombinacji poprzedniego znaku. Litera 'ć' następuje po 'c' a przed 'd', zatem poprawny jest następujący porządek haseł:

(7) cap < cofać < czyn < ćma < ćpać < ćwikła < dal

We współczesnej pisowni polskiej nie występują znaki diakrytyczne, choć jeszcze w końcu XIX w. stosowano znak iloczasu (np. *daléj, przytém*). Uporządkowanie polskich haseł jest jednostopniowe, choć znaki diakrytyczne mogą pojawić się w wyrazach obcych.

W wielu słownikach wydanych na przełomie wieku XX, a nawet w latach międzywojennych stosowano odmienne zasady. W słowniku

Brücknera z r. 1927[5] dwuznak 'ch' traktowany był jako odrębna jednostka alfabetyczna i porządkowany po 'h'; 'ą', 'ę', 'ł', 'ń', 'ś i 'ż' traktowane były, jak dziś, jako odrębne jednostki alfabetyczne, natomiast nie odróżniano 'ć' od 'c', 'ó' od 'o' i 'ź' od 'z'.

Tak np. w słowniku Brücknera:

'cwany' < 'ćwikła' < 'cybuch'

'wojłok' < 'wójt' < 'wola'

'zdun' < 'źdźbło' < 'zegar'

W wydanym w r. 1866 Słowniku Polsko-Łacińskim Bielikiewicza[6] z 9 polskich liter wyróżnione jest jedynie 'ł'. Pozostałe litery: 'ą', 'ć', 'ę', 'ń', 'ó', 'ś', 'ź' i 'ż' traktowane są, odpowiednio, jak 'a', 'c', 'e', 'n', 'o', 's', 'z'. Np.:

'mąka' < 'makówka'

'ćwiartka' < 'cyranka'

'mężczyzna' < 'męzki'

'ósemka' < 'oset'

W Słowniku Karłowicza, Kryńskiego i Niedźwiedzkiego[5] z r. 1900, wszystkie litery polskie są traktowane, jak w słownikach współczesnych, jako odrębne jednostki alfabetyczne.

4.3. Język nowogrecki[7]

Alfabet grecki zawiera 24 litery, przy czym jedna z nich występuje w dwóch równoważnych formach graficznych: 'σ' (*sigma*) i (na końcu wyrazów) 'ς'(*stigma*). Niewielka liczba liter umożliwia prostą transliterację wszystkich znaków. System zalecany przez *American Philological Association*[2] oparty jest w znacznym stopniu na zasadach fonetycznych:

alfabet grecki: αβγδεζηϑικλμνξοπρσςτυφχψω

transliteracja łacińska: abgdezhqiklmncoprsjtufxyw

W języku starogreckim stosowano trzy akcenty (*acutus*, *gravis*, *circumflexus*), *iota subscriptum*, oraz *przydechy* (słaby i mocny). Przydechy pojawiały się też w różnych kombinacjach z akcentami. W procesie kształtowania się współczesnego języka nowogreckiego (greka klasyczna => koiné => katharewusa => dimotiki) liczba znaków diakrytycznych stopniowo zmniejszała się. Od r. 1982 oficjalna pisownia przewiduje jeden akcent (´) i znak dierezy (¨).

Ranga akcentu w języku greckim jest ujemna: hasła akcentowane występują przed hasłami nieakcentowanymi, np.

ώς (*praep* aż do) < ως (podczas, gdy)

ή (czy - czy, niż) < η (*rodzajnik*)

νόμος (prawo) < νομός (gmina)

ώμος (bark) < ωμός (surowy)

τζάμι (szkło) < τζαμί (meczet)

w związku z czym przyjmiemy

D(´) = '0'

D('pusty') = '1'

4.4. Język rosyjski[8]

We wstępie do niektórych słowników (np. Wielki Słownik Rosyjsko - Polski, WSRP[8]) opisuje się alfabet rosyjski jako system 33 liter (wraz z *jerami* - znakiem twardym i miękkim). Takie założenie przyjęliśmy również w poprzedniej pracy[1]. Analiza porządku haseł w słownikach (w tym również WSRP[8]) wykazuje jednak, że z leksykograficznego punktu widzenia tylko 32 znaki stanowią odrębne jednostki alfabetyczne, a 'ē' stanowi modyfikację samogłoski 'e' znakiem diakrytycznym przegłosu (¨). 'ē' jest odrębnym fonemem, ale nie stanowi jednostki alfabetycznej. W piśmie często znak przegłosu pomija się. Transliterację znaków alfabetu rosyjskiego oparliśmy (tam gdzie to było możliwe) na zasadach fonetycznych. Nie jest to jednak system standardowy.

alfabet rosyjski: абвгдежзийклмнопрстуфхцчшщъыьэюя

transliteracja łacińska: abvgde\zijklmnoprstufhc'[]^y*qwx

W Tabeli III. podano hasła rosyjskie zawierające 'e' i 'ē'. Gdyby 'ē' stanowiło odrębną jednostkę 33-literowego alfabetu następującą po 'e'

e < ē

to porządek haseł przybrałby postać taką, jak pokazano w pierwszej kolumnie Tabeli III. Poprawne uporządkowanie haseł rosyjskich jest dwustopniowe: leksykograficzne na zasadzie alfabetu

32-literowego, a następnie uporządkowanie wtórne według znaku przegłosu (¨)

'e' <=> 'ë'

D('e') < D('ë')

Tabela III.

Uporządkowanie wyrazów rosyjskich z 'e' i 'ë'[8]

Hipotetyczne uporządkowanie jednostopniowe		Uporządkowanie dwustopniowe I etap	II etap	Łańcuch rozszerzony
едок	(smakosz)	едок	едок	едок_0000
её	(*acc.* ją)	ее	её	ее_01
ежа	(rżniączka)	еж	ёж	еж_10
елей	(olej święty)	ежа	ежа	ежа_000
ер	(jer, twardy znak)	елей	елей	елей_0000
ерь	(jer', miękki znak)	елка	ёлка	елка_1000
ёж	(jeż)	ер	ер	ер_00
ёлка	(choinka)	ерш	ерш	ерш_100
ёрш	(jazgarz)	ерь	ерь	ерь_000

Przebieg uporządkowania dwustopniowego pokazują dalsze kolumny Tabeli III. W pierwszej kolumnie (hipotetyczny alfabet 33-literowy z odrębnymi jednostkami 'e' i 'ë') hasło 'ёж' (jeż) pojawia się po haśle 'ежа' (rżniączka, *Dactylis L.*), podczas,

gdy poprawne dwustopniowe uporządkowanie zakładające równoważność 'e' i 'ë' i efekt diakrytyczny prowadzi do pojawienia się hasła 'ёж' przed hasłem 'ежа', zgodnie z inwentarzem słowników[8].

4.5. Język niemiecki

W piśmie gotyckim występują, obok odpowiedników wszystkich liter standardowego alfabetu łacińskiego, trzy odmiany litery 's': 'krótkie', 'długie', i 'ostre' (scharfe s) stanowiące ligaturę 'długie s + z'.

We współczesnym języku niemieckim posługującym się alfabetem łacińskim, nie odróżnia się 'krótkiego' i 'długiego' s, występuje natomiast ligatura 'ß' ('ostre s'), obecnie rozwiązywana na dwuznak 'ss'. Ponadto, występują 3 samogłoski z przegłosem (umgelautete Vokale), 'ä', 'ö', 'ü', a w wyrazach obcego pochodzenia pojawia się akcent (´) i sporadycznie znak dierezy ('Aleüten', Aleuty).

Przegłos w samogłoskach 'a', 'o', 'u', wprowadzany jest za pomocą znaku diakrytycznego (¨) jako efekt modyfikujący, jak akcenty w języku francuskim. Nie jest to wybór oczywisty, gdyż w innych językach (np. szwedzki, fiński, węgierski) samogłoska 'ö' traktowana jest jako niezależna jednostka alfabetyczna (litera). Zasady porządkowania haseł z 'ä', 'ö', 'ü', 'ß' ulegały zmianom. W niedawnej pracy Möcker[9] dyskutuje 5 możliwości uszeregowania 'ä' (i innych samogłosek z przegłosem):

i. jako odrębnej jednostki alfabetycznej pomiędzy 'a' i 'b' (jak 'ą' w języku polskim)

aa < az < äa < äz < ba

ii. jako odrębnej jednostki alfabetycznej na końcu alfabetu; podobnie traktowane są inne samogłoski z przegłosem i 'ß'

x < y < z < ä < ö < ü < ß

iii. jako znaku podlegającego zamianie na dwuznak 'ae', porządkowany pomiędzy 'ad' i 'af'.

iv. jako równoważnika dwuznaku 'ae' ze znakiem diakrytycznym o dodatniej randze

'ä' <=> 'ae'

D('ä') > D('ae')

v. jako równoważnika samogłoski 'a' ze znakiem diakrytycznym o dodatniej randze

'ä' <=> 'a'

D('ä') > D('a')

Wszystkie te zasady dotyczą również 'ö' i 'ü'. W tekstach obcojęzycznych 'ä' jest niejednokrotnie zamieniane na dwuznak 'ae' i porządkowane zgodnie z zasadą iii. Obecne zasady ortografii niemieckiej i austriackiej[10] zgodnie przyjmują zasadę v. Przykłady podaje Tabela IV.

Tabela IV.

Uporządkowanie wyrazów niemieckich ze znakiem przegłosu (¨)[11]

Hasło	I etap porządkowania	Łańcuch rozszerzony
Lahme (paralityk)	lahme	lahme_11111
Lähme porażenie)	lahme	lahme_12111
lahmen (kuleć)	lahmen	lahmen_111111
lähmen (sparaliżować)	lahmen	lahmen_121111
losen (słuchać)	losen	losen_11111
lösen (podsłuchiwać)	losen	losen_12111
Ode (oda)	ode	ode_111
Öde (pustkowie)	ode	ode_211
Bluse (bluzka)	bluse	bluse_11111
Blüse (światło latarni morskiej)	bluse	bluse_11211
Schussel (narwaniec)	schussel	schussel_11111111
Schüssel (miska)	schussel	schussel_11121111

Zasady uporządkowania wyrazów z 'ostrym s' ('ß') są bardziej kontrowersyjne. Najczęściej 'ß' traktuje się jako znak 'ss' ze znakiem diakrytycznym, jednakże do dziś pojawiają się propozycje innego uporządkowania opartego na pracach Wittgensteina[12]. Möcker[9] zestawia 5 sposobów traktowania znaku 'ß':

vi. jako odrębnej jednostki alfabetycznej pomiędzy s i t (jak ś w języku polskim)

sz < ß < t

Taka identyfikacja zdaje się nawiązywać do pierwotnego sensu ligatury 'ß', która powstała z dwuznaku 'sz'.

vii. jako odrębnej litery na końcu alfabetu (por. ii.)

viii. jako równoważnika dwuznaku 'ss' ze znakiem diakrytycznym o ujemnej randze[13]

'ß' <=> 'ss'

D('ß') < D('ss')

ix. jako równoważnika dwuznaku 'ss' ze znakiem diakrytycznym o dodatniej randze

'ß' <=> 'ss'

D('ß') > D('ss')

x. jako równoważnika pojedynczego 's' ze znakiem diakrytycznym o dodatniej randze[12]

'ß' <=> 's'

D('ß') > D('s')

W tym ostatnim wypadku hasło 'Muse' występuje bezpośrednio po 'Muße'.

System viii. przyjęty był przez Dudena[13] i normy niemieckie z r. 1901, system ix. odpowiada normom austriackim Ökonorm[10], a system x. proponował Wittgenstein[12]. Möcker[9] przytacza przykłady różnego uporządkowania tych samych haseł według różnych prawideł. We współczesnych słownikach niemieckich uporządkowanie większości haseł odpowiada zasadzie viii, t.zn. 'ß' traktowane jest jak 'ss' ze znakiem diakrytycznym o ujemnej randze. Wykorzystując omówione wyżej zasady v. i viii. można zaproponować następującą hierarchię znaków diakrytycznych:

D('ß') = '0' D(¨) = '2'

D('pusty') = '1' D(´) = '3'

"Wirtualnemu" znakowi diakrytycznemu odróżniającemu 'ß' od 'ss' przypisujemy najniższą rangę, a rangi wyższe kolejno znakowi 'pustemu' (niemodyfikowane 'ss' lub samogłoska bez przegłosu lub akcentu) znakowi przegłosu (¨) i akcentu (´). Leksykograficzną naturę znaku 'ß' ('ss' z wirtualnym znakiem diakrytycznym, a nie odrębna jednostka alfabetyczna), a także ujemną rangę wirtualnego znaku diakrytycznego przy 'ß' potwierdzają przykłady zestawione w Tabeli V.

Tabela V.

Uporządkowanie haseł niemieckich z 'ostrym s'[11]

Hasło	I etap porządkowania	Łańcuch rozszerzony
Maß (miara)	mass	mass_1100
Massage (masaż)	massage	massage_1111111
Maße (w miarę)	masse	masse_11001
Masse (masa)	masse	masse_11111
Maßeinheit (jednostka miary)	masseinheit	masseinheit_110011111111
maßen (ponieważ)	massen	massen_110011
Massen (*pl. od Masse*)	massen	massen_111111
massig (masywny)	massig	massig_111111
mäßig (powściągliwy)	massig	massig_120011

W Tabeli podano hasła z 'ß' oraz odpowiadające im łańcuchy rozszerzone. Ponieważ ligaturę 'ß' rozwija się na dwuznak 'ss', to odpowiedni znak rangi (D='0') pojawia się dwukrotnie.

Niektóre hasła zamieszczone w Wielkim Słowniku Niemiecko-Polskim (WSNP[11]) uporządkowane są w sposób niekonsekwentny. Hasło 'Reisschnaps' (*wódka ryżowa*, łańcuch rozszerzony 'reisschnaps_1111111111') powinno następować po, a nie przed hasłem 'Reißaus' (*ucieczka*, łańcuch rozszerzony 'reissaus_11100111'). Podobnie 'Reisspeise' (*potrawa z ryżu*, 'reisspeise_1111111111') powinno poprzedzać hasło 'Reißstift' (*pluskiewka*, łańcuch rozszerzony 'reissstift_1110011111') . Jest to zapewne błąd przy sporządzaniu słownika, gdyż inne słowniki (np. Chodera - Kubicy[11], lub Wahriga[11]) podają poprawną kolejność podobnych haseł.

4.6. Język francuski[3]

Uporządkowanie leksykograficzne opiera się na standardowym 26-literowym alfabecie łacińskim. W większości słowników (zwłaszcza drukowanych we Francji) występuje ligatura 'œ' porządkowana jak dwuznak 'oe'

'bock' < 'bœuf' < 'boggie'

'odorat' < 'œil' < 'œuvre' < 'offensant'

Ligatura 'œ' występuje w Wielkim Słowniku Francusko - Polskim (WSFP)[3], a także w słowniku Hamela[3], natomiast w wydanym w Holandii słowniku naukowym Doriana[3] zastąpiona jest dwuznakiem:

'bœuf' => 'boeuf'; 'œuvre' => 'oeuvre'

W niektórych słownikach (np. WSFP, Hamel[3]) sporadycznie pojawia się też ligatura 'æ' równoważna dwuznakowi 'ae' (np. 'cæcum', kątnica).

Uporządkowanie II stopnia dopuszcza rozbudowany system znaków diakrytycznych. Występują tu 3 akcenty nad samogłoskami (´), (`), (^), znak dierezy (¨), oraz t.zw. *cedilla* (cedille) modyfikująca spółgłoskę 'c' (ç). Przykłady dwustopniowego uporządkowania według znaków diakrytycznych pokazuje Tabela II w rozdziale 2 i Tabela VI. Przykłady zestawione w tych tabelach ilustrują naturę liter modyfikowanych znakami diakrytycznymi i hierarchię znaków.

Analizowane hasła pozwalają ustalić następujące relacje:

('pusty') < (`) < (´)
('pusty') < (^) < (´)
('pusty') < (¨)

Wszystkie znaki (z wyjątkiem *cedille*, dla której nie udało się znaleźć odpowiednich przykładów) mają rangę dodatnią, więc znakowi 'pustemu' przypiszemy rangę '0'. Biorąc pod uwagę, że cedille występuje wyłącznie przy jednej spółgłosce ('c') i nie odgrywa większej roli w porządkowaniu haseł, przyporządkujemy jej rangę '1'. Położenie znaków (¨) i (^) względem akcentu *grave* nie wynika z dostępnych przykładów.

Proponowana hierarchia znaków diakrytycznych:

D('pusty') = 0 D(`) = '2'
D(¨) = D(ç) = D(^) = '1' D(´) = '3'

Tabela VI.

Uporządkowanie haseł francuskich[3]

Sekwencja haseł	Wnioski
me (*acc.* mnie) méandre (meander) mèche (knot, lont) mèdaille (medal) membre (członek) même (ten sam) mental (umysłowy)	e <=> ê <=> è <=> é
masure (rudera) mat (matowy) mât (maszt) matador (matador)	â <=> a; D(^) > D('pusty')
foret (świder) forêt (bór)	D(^) > D('pusty')
pêche (brzoskwinia) péché (grzech pierworodny)	D(^) < D(´)
la (*rodzajnik*) là (tam)	D(`) > D('pusty')
ou (lub) où (gdzie)	D(`) > D('pusty')
zèbre (zebra) zébré (pręgowany)	D(`) < D(´)
zèle (gorliwość) zélé (gorliwy)	D(`) < D(´)
cale (klin) calé (bogaty)	D(´) > D('pusty')
de (*praep.*, z, od) dé (kostka do gry)	D(´) > D('pusty')
caille (przepiórka) caïman (kajman) caïque (kajak) caisse (kasa)	'ï' <=> 'i'
mais (lecz) maïs (kukurydza)	D(¨) > D('pusty')
perçage (wiercenie) percale (perkal) perçant (ostry, przenikliwy) percer (przebijać)	'ç' <=> 'c'

jest zgodna ze wszystkimi analizowanymi przykładami, lecz może ulec modyfikacji w przypadku znalezienia nowych danych.

4.7. Język włoski[14]

Uporządkowanie leksykograficzne oparte jest na standardowym alfabecie łacińskim. W wyrazach rodzimego pochodzenia nie występują litery 'k', 'w', 'x' i 'y'. Trzy znaki diakrytyczne: akcenty (´), (`) i znak dierezy (¨) stanowią podstawę uporządkowania II stopnia. Dodatnią rangę znaków diakrytycznych udowadnia porządek haseł[14]:

'di' (*praep.* od) < 'dí' (dzień)
'li' (*acc. pl.* ich) < 'lí' (tam)
'la' (*rodzajnik*) < 'là' (tam)
'se' (jeśli) < 'sè' (*acc.* siebie)

sugerujący następującą hierarchię akcentów:

D(´) > D('pusty')
D(`) > D('pusty')

Przyjęcie takiej hierarchii implikuje porządek haseł, które nie pojawiają się w słownikach[14] obok siebie:

'e' (i, a) < 'è' (jest)
'faro' (latarnia morska) < 'farò' (*fut.* zrobię)
'giro' (krążenie) < 'girò' (*fut.* pójdę)

Należy zauważyć, że w niektórych różnie akcentowanych ale identycznie zapisywanych wyrazach (homografach) znak akcentu nie występuje. Przykładem jest 'ancora' (*ancora*, kotwica) z

akcentem na pierwszej sylabie, porządkowana przed hasłem 'ancora' (*ancora*, wciąż jeszcze) wymawiana z akcentem na drugiej sylabie. Zapis graficzny nie odróżnia obu haseł. Możliwym sposobem zaznaczenia kolejności haseł

'ancora' (kotwica) < 'ancora' (wciąż jeszcze)

jest uporządkowanie trzystopniowe. Atrybutem odróżniającym te hasła jest znaczenie, lub wymowa. Zasady wielostopniowego porządkowania haseł z atrybutami omówiliśmy w pracy[1].

4.8. Język hiszpański[15]

Alfabet łaciński uzupełniają 3 dodatkowe jednostki alfabetyczne: litera 'ñ' oraz (traktowane jako odrębne litery) dwuznaki 'ch' i 'll'. Podstawę uporządkowania leksykograficznego stanowi 29-elementowy alfabet:

abc(ch)defghijkl(ll)mnñopqrstuvwxyz

O tym, że dwuznaki (ch) i (ll) i litera 'ñ' stanowią odrębne jednostki alfabetyczne świadczą następujące relacje:

'czar' (car) < 'chacal' (szakal)
'luz' (światło) < 'llama' (płomień)
'nutrir' (żywić) < 'ñaque' (kupa gratów)

Znaki diakrytyczne: akcent (´) i znak dierezy (¨) uwzględnia się w II stopniu uporządkowania. Dowodem na to, że ani akcentowane samogłoski (np. 'á', 'é) ani samogłoski ze znakiem dierezy (np. 'ü') nie stanowią odrębnych jednostek alfabetycz-

nych są sekwencje:

'aledo' (skrzydlaty)

'álamo' (topola)

'alargar' (przedłużać)

a także:

'aguinaldo' (prezent)

'agüista' (kuracjusz)

'aguja' (igła)

Dodatnia ranga akcentu, D(´) > D('pusty') wynika z relacji:

'arteria' (arteria) < 'artería' (podstęp, przebiegłość)

'asa' (ucho, rączka) < 'asá' (tak, w ten sposób)

'pelicano' (szpakowaty) < 'pelícano' (pelikan).

4.9. Język czeski[16]

Język czeski ma dość nieprzejrzysty sposób zapisywania haseł. Wynika to z faktu, że dwa znaki specjalne, (ˇ) (*hačeк*) i akcent (´) odgrywają różne role w kombinacji z różnymi spółgłoskami i samogłoskami. Obok standardowych liter alfabetu łacińskiego występuje 5 dodatkowych spółgłosek stanowiących odrębne jednostki alfabetyczne: 'č', 'ř', 'š', 'ž', oraz dwuznak 'ch'. Trzy zmiękczone spółgłoski (ň, d´ i t´) nie stanowią odrębnych jednostek i porządkuje się je jak 'n', 'd' i 't' ze znakami diakrytycznymi. Alfabet czeski obejmuje więc 31 jednostek alfabetycznych:

abcčdefgh(ch)ijklmnopqrřsštuvwxyzž

Występują 3 znaki diakrytyczne: (´) przy samogłoskach 'a', 'e', 'i', 'o', 'u', 'y' oznacza akcent; ten sam znak następujący po spółgłoskach 'd' lub 't' oznacza zmiękczenie. Haczyk (ˇ), nad spółgłoskami 'c', 'r', 's' i 'z' definiuje nowe litery, nad samogłoską 'e' oznacza przegłos ('ě'), a w połączeniu ze spółgłoską 'n' - zmiękczenie (ň). Przegłos samogłoski 'u' oznacza się znakiem diakrytycznym (°).

Uporządkowanie haseł w języku czeskim jest dwustopniowe: leksykograficzne według 31-elementowego alfabetu i wtórne według znaków diakrytycznych.

Przykłady pokazane w Tabeli VII. wykazują, że zmiękczone spółgłoski d´, ň i t´ stanowią tylko modyfikacje odpowiednich spółgłosek twardych. Odrębną jednostką alfabetyczną jest natomiast dwuznak 'ch' zlokalizowany po 'h', oraz spółgłoski z haczykiem 'č', 'ř', 'š' i 'ž'. Z Tabeli VII. wynika sens modyfikowanych samogłosek 'é', 'ě' i 'ů' i hierarchia znaków diakrytycznych.

Dodatnia ranga znaku (ˇ) przy 'e' wynika z relacji:

'bel' < 'běl'

a znaku (´) z relacji:

'tyl' < 'týl'

oraz:

'sitový' < 'sít´ový'

Tabela VII.

Uporządkowanie haseł czeskich wg[16]

Sekwencje haseł		Wnioski
med (miód)	[000]	
měd´ (miedź)	[0**]	
medovina (miód pitny)		d <=> d´; D('pusty') < D(˘)
zad´ (rufa statku)		
zadní (tylny)		
zad´ový (rufowy)		d <=> d´
báň (kopuła)		
bandáž (bandaż)		
baňka (bańka)		n <=> ň
konsistentni (gęsty)		
koňský (koński)		
konsola (konsola)		n <=> ň
lat´ (żerdź)		
látka (materiał)	[0*000]	
lat´ka (listwa)	[00*00]	t <=> t´; D('pusty') > D(´) !
sitový (sitowy)	[00000*]	
sít´ový (sieciowy)	[0**00*]	t <=> t´; D('pusty') < D(´)
cukr (cukier)		
čas (czas)		č > c
hygienický (higieniczny)		
charakter (charakter)		
identický (identyczny)		h < (ch) < i
rychlý (szybki)		
řad (*mat.* rząd)		ř > r
sytiti (nasycać)		
šablona (szablon)		š > s
zvuk (dźwięk)		
žar (żar)		ž > z
bel (*fiz.* bel)	[000]	
běl (biel)	[0*0]	
benzin (benzyna)		e <=> ě; D(˘) > D('pusty')
tyl (*włok.* tiul)		
týl (*wojsk.* zaplecze, tył)		D('pusty') < D(´)
průbeh (przebieg)		
pruh (smuga)		
průhyb (wygięcie)		u <=> ů

Występujące w analizowanych słownikach[16] uporządkowanie haseł

'látka' < 'lat´ka'

jest sprzeczne z wnioskami o randze znaku (´) i charakterze zmiękczonej spółgłoski (t´). Kolejność taka występuje w kilku słownikach wydanych w latach 1953-1978 przez praskie wydawnictwa SPN i SNTL[16]; można sądzić, że ta grupa haseł była w kolejnych wydaniach kopiowana. Ponadto w słowniku Muszkatowej i Valouška[16] występuje błędna kolejność:

'hnědý' < 'hněd´'

a w słowniku Vydry[16]:

látkový < lat´ka

Pozostałe słowniki podają poprawną kolejność.

Ponieważ znaki (ˇ) i (°) nigdy nie występują przy tych samych literach, natomiast każdy z nich może pojawić się zamiast akcentu (´), przyjmiemy następujące rangi:

D('pusty') = '0'

D(´) = '1'

D(°) = D(ˇ) = '2'

a w przypadku włączenia do porządkowanych haseł cyfr:

D('pusty') = '#'

D(´) = '$'

D(°) = D(ˇ) = '&'

4.10 Język węgierski[17]

W języku węgierskim obok 26 znaków alfabetu łacińskiego występują dwie samogłoski z przegłosem: 'ö', 'ü' oraz szereg dwuznaków stanowiących odrębne fonemy, a równocześnie niezależne jednostki alfabetyczne: (cs), (gy), (ly), (ny), (sz), (ty) i (zs). W odróżnieniu od języka niemieckiego, samogłoski 'ö', i 'ü' stanowią odrębne litery usytuowane w alfabecie po tych samych samogłoskach bez przegłosu. Tak więc alfabet węgierski obejmuje 35 elementów:

abc(cs)defg(gy)hijkl(ly)mn(ny)oöpqrs(sz)t(ty)uüvwxyz(zs)

Samogłoski 'a', 'e', 'i', 'o', 'u' oraz 'ö' i 'ü' mogą występować w formie przedłużonej, co zapisuje się za pomocą znaku (´) nad samogłoskami bez przegłosu ('á', 'é', 'í', 'ó', 'ú') oraz znaku (˝) nad samogłoskami z przegłosem ('ő', 'ű').

Ortografia węgierska przewiduje skrócenie podwojonych dwuznaków na trójznaki:

'cscs' => 'ccs'; 'gygy' => 'ggy'; 'zszs' => 'zzs', itd.

O ile dwuznaki ('cs', 'gy', itd) oznaczają pojedyncze jednostki alfabetyczne (i pojedyncze fonemy), to trójznaki pochodzące ze skrócenia par dwuznaków intepretuje się jako dwie (identyczne) jednostki alfabetyczne i tak też porządkuje.

Przykłady pokazane w Tabeli VIII. potwierdzają odrębność dwuznaków 'cs', 'gy', ..., 'zs' i samogłosek z przegłosem ('ö', 'ü'), a także ilustrują zasadę porządkowania trójznaków.

Uporządkowanie haseł węgierskich jest dwustopniowe: leksykograficzne według 35-znakowego alfabetu oraz wtórne według

znaku przedłużenia (´ lub ˝).

Przykłady zestawione w Tabeli VIII. potwierdzają równoważność krótkich i przedłużonych samogłosek, a także dodatnią rangę znaków (´), lub (˝). Przy hasłach różniących się tylko znakiem diakrytycznym, w nawiasach kwadratowych podano dodatkową część łańcucha rozszerzonego zawierającą kody rangi, D. Gwiazdką (*) zaznaczono położenie znaku diakrytycznego.

Wnioski z przykładów podanych w Tabeli VIII. sugerują następującą hierarchię znaków diakrytycznych:

D('pusty') = '0'

D('á', 'é', 'í', 'ó', 'ú') = '1'

a dla samogłosek z przegłosem:

D('ő', 'ű') = '0'

D('ö', 'ü') = '1'

Jeden z przykładów wskazuje na osobliwość porządkowania wyrazów obcego pochodzenia. Hasło 'nylon', wymawiane jako dwie sylaby: *ni-lon*, występuje w słowniku pod literą 'n', a nie 'ny', w związku z czym poprzedza ono hasło 'nyáj'. Formalne potraktowanie tekstu zawierającego hasło 'nylon' spowodowałoby automatyczną zamianę dwuznaku (ny) na miękką spółgłoskę (ny). Normalnie, spółgłoska (ny) występuje albo na końcu wyrazu (np. 'igény'= pretensja), albo przed samogłoskami ('nyár' = lato, 'nyel' = łykać, 'nyil' = strzała, 'nyomban' = natychmiast).

Tabela VIII.

Uporządkowanie haseł węgierskich wg[17]

Sekwencje haseł	Wnioski
cukor (cukier) csabít (nęcić)	c < (cs)
zűrzavar (bałagan, zamęt) zsák (worek)	z < (zs)
egzotikus (egzotyczny) egyéb (inny)	g < (gy)
szenzációs (sensacyjny) szennyes (brudny)	n < (ny)
pulzus (puls) pulya (indyk)	l < (ly)
fagy (mróz) faggyu (łój) fagylalt (lody)	ggy <=> (gy)(gy)
enyhe (łagodny) ennyi (tyle)	nny <=> (ny)(ny)
nulla (zero) nylon (nylon) nyáj (stado)	ny ≠ (ny) n < (ny)
ũrge (suseł) ũrhajó (statek kosmiczny) ũrit (opróżniać)	ũ <=> ű
alkusz (makler) áll (podbródek) alma (jabłko)	a <=> á
lap (karta, gazeta) [000] láp (bagno) [0*0]	D(´) > D('pusty')
ver (uderzać) [000] vér (krew) [0*0]	D(´) > D('pusty')
kor (wiek, era) [000] kór (choroba) [0*0]	D(´) > D('pusty')
kõr (koło) [000] kőr (kier) [0*0]	D(õ) > D(õ)

5. PROGRAM REALIZUJĄCY UPORZĄDKOWANIE HASEŁ W RÓŻNYCH JĘZYKACH

5.1. Podstawowe elementy programu

Program *Transorder* analizuje wyjściowe hasła obcojęzyczne (zmienna **ENTRY**) poddane umownej transkrypcji znakami alfabetu łacińskiego i tworzy równoważne tym hasłom łańcuchy "zastępczych znaków ASCII" (w przypadku dwustopniowego porządkowania ze znakami diakrytycznymi: łańcuchy rozszerzone). Wynikiem tej operacji jest zmienna **TRANSENTRY** stanowiąca podstawę sortowania haseł. Postępowanie takie jest uogólnieniem programów "Transpol", "Transrus" i "Transgrek" opisanych w pracy[1]; w odróżnieniu od wcześniejszych programów, program *Transorder* dopuszcza dwustopniowe uporządkowanie z udziałem znaków diakrytycznych oraz przetwarza ligatury, dwuznaki i trójznaki.

System porządkowania haseł właściwy dla każdego języka określają zmienne logiczne (przyjmujące wartości '0' lub '1'). Charakteryzują one zarówno sam język, jak i przyjęty sposób transkrypcji. Występowanie *ligatur*, które należy zamienić na pary znaków (æ, œ, we francuskim i angielskim, ß w niemieckim), charakteryzuje zmienna **LIG**. Występowanie dwuznaków reprezentujących niezależne jednostki alfabetyczne ('ch' w hiszpańskim i czeskim, 'cs', 'zs' w węgierskim, a także (c´), (n´) i (s´) zastępujących w transkrypcji pojedyncze litery polskie 'ć', 'ń' i 'ś', określa zmienna **DIPH**. Zmienna **TRIP** wskazuje na występowanie trójznaków, np. pochodzących ze skrócenia par dwuznaków (język węgierski), a zmienna **DIA** - występowanie znaków diakrytycznych. Zmienna **MOD** przybiera wartość '1' jeżeli rozpatrywany

język posługuje się alfabetem zmodyfikowanym, zawierającym inne znaki niż standardowy alfabet łaciński (np. grecki, rosyjski), te same znaki w innej kolejności (np. 'y' po 'i' w litewskim), a także jeśli występują w nim samodzielne dwuznaki traktowane jako litery ('ll' i ch' w hiszpańskim, 'cs', 'gy' w węgierskim, itp). Obecność standardowych liter łacińskich ze znakami diakrytycznymi nie wpływa na wartość zmiennej **MOD**.

Szczegółowych informacji o alfabecie, dwuznakach, itp. zawierają zmienne tekstowe (lub dwuwymiarowe tablice):

- łańcuch (tablica) znaków zastrzeżonych, **RESERVED**. Znakami zastrzeżonymi, które nie mogą wystąpić w przetwarzanych tekstach (tzn. w zmiennej **ENTRY**) i są z nich usuwane procedurą *ClearRes*, są znaki zastępcze reprezentujące dodatkowe jednostki alfabetyczne (litery, dwuznaki) analizowanego języka.

- łańcuch (tablica) ligatur, **LIGATURES** i odpowiadających im par znaków, **LIGCODES**,

- łańcuch (tablica) jednoznakowych jednostek alfabetycznych (w odpowiedniej transliteracji lub w postaci znaków zastępczych), **ALPHABET** i odpowiadających im znaków ASCII, **ASCII**. Łańcuchy te stają się identyczne (i nie są stosowane) gdy **MOD** = '0'.

- łańcuch (tablica) dwuznaków traktowanych jako niezależne jednostki alfabetyczne, **DIPHTONGS** i odpowiadających im zastępczych znaków ASCII, **DIPHCODES**,

- łańcuch (tablica) trójznaków, **TRIPTONGS**;

- łańcuchy znaków diakrytycznych, **DIACRITS**, i ich rang **DIARANKS**.

Analizy i przekształcenia haseł (**ENTRY** => **TRANSENTRY**) doko-

nuje się za pomocą procedur.

- Procedura *Decompress* zamienia ligatury na właściwe kombinacje znaków, uwzględniając ewentualnie również znaki diakrytyczne. Korzysta ona przy tym ze zmiennych **LIGATURES** i **LIGCODES**.

Procedura *ClearRes* usuwa ze zmiennej **ENTRY** znaki zastrzeżone używane do transkrypcji dodatkowych liter i jednostek alfabetycznych, które nie mogą występować w przetwarzanym tekście. Listę takich znaków zawiera zmienna **RESERVED**.

- Procedura *DiaMarks* usuwa ze zmiennej **ENTRY** znaki diakrytyczne i buduje dodatkową część łańcucha rozszerzonego w oparciu o zmienne **DIACRITS** i **DIARANKS**.

- Procedura *Combine* usuwa dwuznaki i łączy je w jednoznakowe symbole pośrednie, korzystając ze zmiennych **DIPHTONGS** i **DIPHCODES**.

- Procedura *Explode* wyszukuje trójznaki powstałe ze skrócenia pary identycznych dwuznaków (w języku węgierskim: 'ggy' <=> 'gygy') i zamienia je na pary dwuznaków. Trójznaki identyfikuje się przez porównanie z łańcuchem **TRIPTONGS**.

- Procedurę *Trans* stosuje się po "oczyszczeniu" zmiennej **ENTRY** procedurami *Combine*, *Decompress*, *Explode* i *DiaMarks*, gdy **MOD** = '1'. Zamienia ona pojedyncze znaki łańcucha **ENTRY** na znaki zastępcze ASCII korzystając ze zmiennych **ALPHABET** i **ASCII**. Procedura ta, podobnie jak *Decompress* analizuje pojedyncze znaki łańcucha, podczas gdy procedury *Combine* i *DiaMarks* analizują pary znaków.

5.2 Schemat blokowy programu

CZYTAJ: **LIG, TRIP, DIPH, DIA, MOD, LIGATURES** i **LIGCODES, DIPHTONGS, TRIPTONGS** i **DIPHCODES, DIACRITS** i **DIARANKS, ALPHABET** i **ASCII.**

OTWÓRZ BAZĘ DANYCH <Nazwa> I USTAW LICZNIK REKORDÓW W POZYCJI '1'

CZYTAJ **ENTRY**

JEŚLI (**RESERVED** ≠ '') WYKONAJ *ClearRes*

LIG = ?

JEŚLI (**LIG** = '1') WYKONAJ *Decompress*

TRIP = ?

JEŚLI (**TRIP** = '1') WYKONAJ *Explode*

DIPH = ?

JEŚLI (**DIPH** = '1') WYKONAJ *Combine*

DIA = ?

JEŚLI (**DIA** = '1') WYKONAJ *DiaMarks*

MOD = ?

JEŚLI (**MOD** = '1') WYKONAJ *Trans*

EOF() = ?

JEŚLI (EOF() = '0') DODAJ 1 DO LICZNIKA REKORDÓW I WRÓĆ DO ROZKAZU "CZYTAJ **ENTRY**"

ZAMKNIJ BAZĘ DANYCH

KONIEC

Pierwsza część programu obejmuje operacje globalne - wprowadzenie do programu podstawowych informacji o języku, systemie transkrypcji i porządkowania (zmienne logiczne) oraz łańcuchy danych szczegółowych. Otwiera się analizowaną bazę danych i ustawia licznik rekordów na pierwszej pozycji.

Druga część zawiera operacje na rekordach. Kolejno wczytuje

się z bazy danych zawartość pola ***Name*** do zmiennej **ENTRY**, analizuje i przekształca na zmienną **TRANSENTRY**. Jeśli **DIA** = '1', to **TRANSENTRY** stanowi łańcuch rozszerzony, a jego długość jest równa podwójnej długości łańcucha **ENTRY**. Jeśli porządkowanie nie uwzględnia znaków diakrytycznych (**DIA** = '0'), łańcuch **TRANSENTRY** stanowi transkrypcję hasła **ENTRY** o długości zbliżonej do hasła wyjściowego. W procesie analizy i przekształcania zmiennej **ENTRY** wykorzystuje się procedury *ClearRes*, *Decompress*, *Combine*, *Explode*, *DiaMarks* oraz *Trans*. Przekształcone hasło (łańcuch rozszerzony) **TRANSENTRY** zapisuje się na nowym polu każdego rekordu o nazwie ***TransName***.

Teksty programu głównego i procedur napisane w języku *FoxBase* podano w załączniku A.

Omówimy dokładniej dwa przykłady zastosowania ogólnego programu. Wybór czeskiego wynika z tego, że język ten zawiera wszystkie, z wyjątkiem ligatur i trójznaków, cechy porządkowania (**DIPH** = **DIA** = **MOD** = '1'). Język niemiecki ilustruje przetwarzanie ligatur i porządkowanie według znaków diakrytycznych z ujemną rangą.

5.3. Przykład 1: język czeski

W języku czeskim nie występują ligatury i trójznaki, pojawia się natomiast dwuznak 'ch' traktowany jako jedna litera porządkowana po 'h', dodatkowe litery alfabetu ('č', 'ř', 'š', 'ž') oraz znaki diakrytyczne (ˇ), (´) i (°). Przyjmiemy najpierw dwuznakową transkrypcję dodatkowych jednostek alfabetycznych znakami ASCII:

'ch' => 'ch' (bez zmian)

'č' => (c^)

'ř' => (r^)

'š' => (s^)

'ž' => (z^)

Nie występujące w zbiorze standardowych znaków ASCII znaki diakrytyczne (˘) i (°) zastąpimy gwiazdką (*) i tildą (~); znaki te, podobnie, jak akcent ('), będziemy umieszczać po odpowiednich samogłoskach lub spółgłoskach tworząc dwuznaki:

'á' => (a')

'é' => (e'), itd.

'ě' => (e*)

'ň' => (n*)

'ů' => (u~)

Wygodnie jest zaznaczyć w transkrypcji różne funkcje znaku (˘). Gdy tworzy on nowe litery ('č', 'ř', 'š', 'ž'), reprezentujemy go "daszkiem" (^), a gdy stanowi znak diakrytyczny modyfikujący samogłoskę 'e' lub zmiękczający spółgłoskę 'n' - stosujemy gwiazdkę (*).

Zmienne logiczne przybierają postać:

LIG = '0'; **TRIP** = '0'; **DIPH** = '1'

DIA = '1'; **MOD** = '1'.

Łańcuchy przyporządkowania dwuznaków stanowiących niezależne jednostki alfabetyczne, zarówno tych, które występują w oryginalnym zapisie standardowymi literami alfabetu łacińskiego

('ch'), jak i będących wynikiem transkrypcji liter 'č', ..., 'ž' zbudujemy ze znaków ASCII nie występujących w normalnych tekstach.

DIPHTONGS = 'C^CHChR^S^Z^c^chr^s^z^'

DIPHCODES = '[[]]]]{{}}::@@&&##$$\\'

Operacje zamiany dwuznaków zawartych w łańcuchu **DIPHTONGS** na pojedyncze znaki z łańcucha **DIPHCODES** nie dotyczą dwuznaków powstałych w wyniku transkrypcji liter modyfikowanych znakami diakrytycznymi. Łańcuchy opisujące znaki diakrytyczne (po transkrypcji) mają postać:

DIACRITS = ''*~'; **DIARANKS** = '$%%#'

Łańcuch **DIARANKS** jest o 1 znak dłuższy od łańcucha **DIACRITS**: pierwszy znak z prawej (tu: '#') oznacza rangę znaku 'pustego'. Ponieważ nie występują znaki diakrytyczne o randze ujemnej, znakowi 'pustemu' przypisujemy najniższą rangę ('#', ASC=35); kolejne rangi charakteryzują kody '$' (ASC=36) i '%' (ASC=37). Taki wybór kodów znaków diakrytycznych (zamiast: '0', '1', '2') dopuszcza do porządkowania wszystkie cyfry i znaki przestankowe (por. zmienną **ALPHABET**). Znaki ASCII zawarte w łańcuchu **DIARANKS** nie mogą występować w łańcuchu **ASCII** i ograniczają z dołu zakres dopuszczalnych znaków w porządkowanych hasłach.

Pełny zestaw porządkowanych znaków zawiera wszystkie samodzielne jednostki 31-znakowego alfabetu (duże i małe litery), cyfry i podstawowe znaki przestankowe:

```
!()*+,-./0123456789:;<=>?ABCČDEFGH(CH)IJKLMNOPQR
ŘSŠTUVWXYZŽabcčdefgh(ch)ijklmnopqrřsštuvwxyzž'
```

zmienna **ALPHABET** zawiera zamiast liter czeskich i dwuznaku (ch) odpowiednie znaki zastępcze (por. **DIPHCODES**)

```
ALPHABET =  '!()*+,-./0123456789:;<=>?ABC[DEFGH]IJKLMNOP
            QR{S}TUVWXYZ|abc@defgh&ijklmnopqr#s$tuvwxyz\'
```

Odpowiadające tym znakom znaki ASCII zawarte są w łańcuchu:

```
ASCII =     '()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQR
            STUVWXYZ[\]^_`abcdefghijklmnopqrstuvwxyz{|}~'
```

Znaki zastrzeżone do transkrypcji dwuznaków i znaków diakrytycznych zawarte są w łańcuchu: **RESERVED** = '#$&@[\]{|}~'

5.4. Przykład 2: język niemiecki

Język niemiecki nie zawiera samodzielnych jednostek alfabetycznych nieobecnych w standardowym alfabecie łacińskim. Występuje natomiast ligatura 'ß' odpowiadająca dwuznakowi 'ss' z wirtualnym znakiem diakrytycznym o ujemnej randze. W transkrypcji ASCII będziemy 'ß' zastępować znakiem '$'

```
'ß' => '$'
```

Znaki diakrytyczne, to przede wszystkim znak przegłosu (¨), sporadycznie pojawia się akcent (´) oraz "wirtualny znak diakrytyczny" przy 'ß'.

Zmienne logiczne przybierają postać:

LIG = '1'; **TRIP** = '0'; **DIPH** = '0';

DIA = '1'; **MOD** = '0'

Łańcuchy przyporządkowania ligaturom kombinacji znaków mają postać:

LIGATURES ='$'; **LIGCODES** = 'ss``

Transkrypcję samogłosek z przegłosem ('ä', 'ö', itd) oprzemy na znaku (") umieszczonym po modyfikowanych samogłoskach:

'ä' => (a")

'ö' => (o")

'ü' => (u")

Zmienne opisujące znaki diakrytyczne mają postać:

DIACRITS = '`"´'; **DIARANKS** = '#%&$'

Ponieważ ranga 'wirtualnego znaku diakrytycznego 'ß'= 'ss`' jest ujemna względem znaku 'pustego', jego ranga ('#', ASC=35) jest niższa niż ranga znaku pustego ('$', ASC=36).

Alfabet niemiecki dopuszcza wszystkie litery 26-znakowego alfabetu łacińskiego i prawie wszystkie znaki ASCII. Nie ma znaków zarezerwowanych, **RESERVED** = '' i zbędne jest stosowanie procedury *ClearRes*. Ponieważ nie występują żadne niestandardowe jednostki alfabetyczne, **MOD** = '0' i nie ma potrzeby stosowania procedury *Trans*.

6. LITERATURA

1. A. Ziabicki, Automatyczne porządkowanie łańcuchów alfanumerycznych według różnych alfabetów i zasad porządkowania. *Prace IPPT PAN*, #40/1988.

2. Computing in the Humanities (P.C. Patton and R.A. Holoien, Eds.), Lexington Books, Lexington-Toronto, 1988, p.51.

3. Wielki Słownik Francusko-Polski, Wiedza Powszechna, Warszawa 1982; Bernard Hamel, Dictionnaire Français - Polonais, Librairie Polonaise a Paris, 1990; Dorian's Dictionary of Science and Technology. French - English. Elsevier 1980.

4. The Compact Edition of the Oxford English Dictionary, Oxford 1971; H.C. Wyld, The Universal Dictionary of the English Language, Routledge & Kegan Paul Ltd., London 1956.

5. Aleksander Brückner, Słownik etymologiczny języka polskiego, Kraków 1927 (Wiedza Powszechna, Warszawa 1974); J. Karłowicz, A. Kryński, W. Niedźwiedzki, Słownik Języka Polskiego, Warszawa 1900 (PIW 1952).

6. Ks. Antoni Bielikiewicz, Słownik Polsko-Łaciński, Kraków 1866.

7. Α. Γεωργοπαπαδάκου, Το Μεγάλο Λεξικό της Νεοελληνικής Γλώσσας, Θεσσαλονίκη 1980

8. A. Mirowicz, I. Dulewicz, I. Grek-Pabis, I. Maryniak, Wielki Słownik Rosyjsko-Polski, Wiedza Powszechna - Sowietskaja Encikłopedija, Warszawa-Moskwa 1970; AN SSSR, Inst. Russkogo Jazyka, Słowar' Russkogo Jazyka, Russkij Jazyk, Moskwa 1981.

9. H. Möcker, Wittgensteins Beitrag zu einer Hierarchie der Buchstaben, *Muttersprache*, XCVI (1986), str. 181.

10. Regeln für die deutsche Rechtschreibung nebst Wörterverzeichnis, Wien 1925.

11. Jan Piprek, Juliusz Ippoldt, Wielki Słownik Niemiecko-Polski, Wiedza Powszechna, Warszawa 1970; J. Choder, S. Kubica, Podręczny Słownik Niemiecko-Polski, Wiedza Powszechna, Warszawa 1966; G. Wahrig, Deutsches Wörterbuch, Bertelsmann Lexicon Verlag, Berlin-Wien 1973.

12. Ludwig Wittgenstein, Wörterbuch für Volksschulen, Wstęp, str. XXX.

13. Konrad Duden, Vollständiges Ortographisches Wörterbuch der deutschen Sprache, Leipzig 1880.

14. Grande Dizionario Garzanti de la Lingua Italiana, Garzanti, 1988; Wojciech Meisels, Podręczny Słownik Włosko-Polski, Wiedza Powszechna, Warszawa, 1964.

15. Stanisław Wawrzkowicz, Kazimierz Hiszpański, Podręczny Słownik Hiszpańsko-Polski, Wiedza Powszechna, Warszawa, 1983.

16. Halina Muszkatowa i Rudolf Valoušek, Polsko-Český a Česko-Polský Technický Slovník, SNTL, Praha, 1958; Bohumil Vydrá, Česko-Polský Slovník, SPN Praha 1958; A. Murawska, E. Tabaczkiewicz, Słownik Techniczny Czesko-Polski, SNTL-WNT, Warszawa 1975.

17. Jan Reychman, Wielki Słownik Węgiersko-Polski, Akademiai Kiadó, Budapest, Wiedza Powszechna, Warszawa, 1980; Livia Havas, Sándor Skripecz, István Varsányi, Kieszonkowy Słownik Węgiersko-Polski, Wiedza Powszechna, Warszawa, 1965;

7. SPIS TREŚCI

DODATEK A

TEKSTY PROGRAMÓW W JĘZYKU FoxBase

Program *Transorder.PRG*

```
SET TALK OFF
SET ECHO OFF
RESTORE FROM <CZECH = Nazwa zbioru danych języka>
SET PROCEDURE TO PROCESSING
USE <CZESKA = Nazwa sortowanej bazy danych>
GO TOP
DO WHILE .NOT. EOF()
STORE TRIM(NAME) TO ENTRY
   IF .NOT. RESERVED = ''
   DO CLEARRES
   ELSE
   ENDIF
REPLACE NAME WITH ENTRY
STORE ENTRY TO TRANSENTRY
STORE '' TO EXTENDED
  IF LIG='1'
  DO DECOMPRESS
  ELSE
  ENDIF
    IF TRIP='1'
    DO EXPLODE
    ELSE
    ENDIF
      IF DIPH='1'
      DO COMBINE
      ELSE
      ENDIF
        IF DIA='1'
        DO DIAMARKS
        ELSE
        ENDIF
          IF MOD='1'
          DO TRANS
          ELSE
          ENDIF
REPLACE TRANSNAME WITH TRANSENTRY + ' ' + EXTENDED
SKIP
ENDDO
CLOSE ALL
```

Zbiór Procedur *Processing.PRG*

```
PROCEDURE CLEARRES
L=LEN(ENTRY)
I=1
SS=''
DO WHILE I<L+1
Q=SUBSTR(ENTRY,I,1)
P=AT(Q,RESERVED)
   IF P<1
   SS=SS+Q
   ELSE
   ENDIF
I=I+1
ENDDO
STORE SS TO ENTRY
RETURN

PROCEDURE TRANS
L=LEN(TRANSENTRY)
SS=''
I=1
DO WHILE I<L+1
Q=SUBSTR(TRANSENTRY,I,1)
P=AT(Q,ALPHABET)
   IF P>0
   Q=SUBSTR(ASCII,P,1)
   ELSE
   ENDIF
   SS=SS+Q
   I=I+1
ENDDO
STORE SS TO TRANSENTRY
RETURN

PROCEDURE DECOMPRESS.
L=LEN(TRANSENTRY)
I=1
SS=''
DO WHILE I<L+1
Q=SUBSTR(TRANSENTRY,I,1)
P=AT(Q,LIGATURES)
   IF P>0
   SS=SS+SUBSTR(LIGCODES, 2*P-1, 2)
   ELSE
   SS=SS+Q
   ENDIF
I=I+1
ENDDO
STORE SS TO TRANSENTRY
RETURN
```

```
PROCEDURE COMBINE
L=LEN(TRANSENTRY)
I=1
SS=''
DO WHILE I<L
Q1=SUBSTR(TRANSENTRY,I,1)
Q2=SUBSTR(TRANSENTRY,I,2)
Q3=SUBSTR(TRANSENTRY,I+1,1)
P=AT(Q2,DIPHTONGS)
   IF P>0
   SS=SS+SUBSTR(DIPHCODES, P,1)
   I=I+2
   ELSE
   SS=SS+Q1
   I=I+1
   ENDIF
ENDDO
  IF I=L
  SS=SS+SUBSTR(TRANSENTRY,L,1)
  ELSE
  ENDIF
STORE SS TO TRANSENTRY
RETURN

PROCEDURE EXPLODE
L=LEN(TRANSENTRY)
I=1
SS=''
DO WHILE I<L-1
Q1=SUBSTR(TRANSENTRY,I,1)
Q2=SUBSTR(TRANSENTRY,I+1,1)
Q3=SUBSTR(TRANSENTRY,I+2,1)
QR=SUBSTR(TRANSENTRY,I+1,2)
QT=SUBSTR(TRANSENTRY,I,3)
P=AT(QT,TRIPTONGS)
   IF P>0
      IF Q1=Q2
      SS=SS+QR+QR
      I=I+3
      ELSE
      SS=SS+Q1+CHR(ASC(Q3)-32)+QR
      I=I+3
      ENDIF
    ELSE
      SS=SS+Q1
      I=I+1
     ENDIF
ENDDO
IF (I=L-1 .OR. I=L)
SS=SS+SUBSTR(TRANSENTRY,I,L-I+1)
ELSE
ENDIF
STORE SS TO TRANSENTRY
RETURN
```

```
PROCEDURE DIAMARKS
L=LEN(TRANSENTRY)
I=1
SS=''
EXT=''
VOID=SUBSTR(DIARANKS, LEN(DIARANKS), I)
DO WHILE I<L
Q1=SUBSTR(TRANSENTRY,I,1)
Q2=SUBSTR(TRANSENTRY,I+1,1)
P=AT(Q2,DIACRITS)
   IF P>0
   SS=SS+Q1
   EXT=EXT+SUBSTR(DIARANKS,P,1)
   I=I+2
   ELSE
   SS=SS+Q1
   EXT=EXT+VOID
   I=I+1
   ENDIF
     IF I=L
     SS=SS+SUBSTR(TRANSENTRY,L,1)
     EXT=EXT+VOID
     ELSE
     ENDIF
ENDDO
STORE SS TO TRANSENTRY
STORE EXT TO EXTENDED
RETURN
```

Zbiór danych języka czeskiego *Czech.MEM*

```
LIG='0'
TRIP='0'
DIPH='1'
DIA='1'
MOD='1'
DIPHTONGS='C^CHChR^S^Z^c^chr^s^z^'
DIPHCODES='[[]]]]{{}}::@@&&###$$\\'
DIACRITS=''*~'
DIARANKS='$%%#'
ALPHABET='!()*+,-./0123456789:;<=>?ABC[DE
    FGH]IJKLMNOPQR{S}TUVWXYZ:abc@defgh&ij
    klmnopqr#s$tuvwxyz\'
ASCII  = '()*+,-./0123456789:;<=>?@ABCDEF
    GHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijk
    lmnopqrstuvwxyz{:}~'
RESERVED='#$&@[\]{:}'
```

TWO-STEP ORDERING OF LEXICOGRAPHICAL ENTRIES IN VARIOUS LANGUAGES

Andrzej Ziabicki

A b s t r a c t

Basing on the theory of ordering alphanumeric strings[1], computer techniques for ordering lexicographical entries (words, abbreviations, sequences of characters) in various languages have been developed. Two-step ordering is considered: primary, lexicographical order according to the appropriate alphabet, and secondary order based on diacritical marks (accents, diaeresis, vocation, etc.). For any language ordering system is characterised by several logical variables describing existence of ligatures, **LIG**, diphtongs, considered as independent alphabetic units, **DIPH**, triple characters representing two contracted diphtongs, **TRIP**, diacritical marks, **DIA**, and modification of the standard roman alphabet, **MOD**. Detailed information is included in the variables **ALPHABET** and **DIACRITS**.

A master programme written in the *FoxBase* database language has been described (Chapter 5. and Appendix A) and two examples of application: *Czech* (Chapter 5.3) and *German* (Chapter 5.4) have been discussed.

Table of Contents