

Reference number of working document: **ISO/IEC JTC1/SC22/WG20 N__**

Date: 2001-12-25

Reference number of document: **ISO/IEC DTR2 14652**

Committee identification: ISO/IEC JTC1/SC22

Secretariat: ANSI

Information technology —

Specification method for cultural conventions

Technologies de l'information —

Méthode de modélisation des conventions culturelles

		Page
2	Contents	
3		
4	1 SCOPE	1
5	2 NORMATIVE REFERENCES	1
6	3 TERMS, DEFINITIONS AND NOTATIONS	2
7	4 FDCC-set	6
8	4.1 FDCC-set description	7
9	4.2 LC_IDENTIFICATION	11
10	4.3 LC_CTYPE	12
11	4.4 LC_COLLATE	26
12	4.5 LC_MONETARY (controversial)	37
13	4.6 LC_NUMERIC	41
14	4.7 LC_TIME (controversial)	42
15	4.8 LC_MESSAGES	49
16	4.9 LC_XLITERATE (controversial)	50
17	4.10 LC_NAME	52
18	4.11 LC_ADDRESS	53
19	4.12 LC_TELEPHONE	55
20	5 CHARMAP	56
21	6 REPERTOIREMAP (controversial)	61
22	7 CONFORMANCE (controversial)	89
23		
24	Annex A (informative) DIFFERENCES FROM POSIX	90
25	Annex B (informative) RATIONALE	92
26	Annex C (informative) BNF GRAMMAR	105
27	Annex D (informative) OUTSTANDING ISSUES	111
28	Annex E (informative) INDEX	113
29	BIBLIOGRAPHY	116

30 Foreword

31
32 ISO (the International Organization for Standardization) and IEC (the International
33 Electrotechnical Commission) form the specialized system for worldwide standardization.
34 National bodies that are members of ISO or IEC participate in the development of
35 International Standards through technical committees established by the respective
36 organization to deal with particular fields of technical activity. ISO and IEC technical
37 committees collaborate in fields of mutual interest. Other international organizations,
38 governmental and non-governmental, in liaison with ISO and IEC, also take part in the
39 work. In the field of information technology, ISO and IEC have established a joint
40 technical committee, ISO/IEC JTC 1.

41
42 The main task of a technical committee is to prepare International Standards but in
43 exceptional circumstances, the publication of a Technical Report of one of the following
44 types may be proposed:

- 45 - type 1, when the required support cannot be obtained for the publication of an
46 International Standard, despite repeated efforts;
- 47
48 - type 2, when the subject is still under technical development or where for any
49 other reason there is the future but not immediate possibility of an agreement on an
50 International Standard;
- 51
52 - type 3, when a technical committee has collected data of a different kind from
53 that which is normally published as an International Standard ("state of the art", for
54 example).
- 55

56
57 Technical Reports are drafted in accordance with the rules given in the ISO/IEC
58 Directives, Part 3.

59
60 Technical Reports of types 1 and 2 are subject to review within three years of publication,
61 to decide whether they can be transformed into International Standards. Technical Report
62 of type 3 do not necessarily have to be reviewed until the date they provide are considered
63 to be no longer valid or useful.

64
65 ISO/IEC TR 14652 is a Technical Report type 1, and it was prepared by Joint Technical
66 Committee ISO/IEC JTC 1, *Information technology, Subcommittee 22, Programming*
67 *languages, their environments and system software interfaces.*

68
69 The Annexes A, B, C, D and E of this Technical Report are for information only.

70 Introduction

71

72 This Technical Report defines a general mechanism to specify cultural conventions, and it
73 defines formats for a number of specific cultural conventions in the areas of character
74 classification and conversion, sorting, number formatting, monetary formatting, date
75 formatting, message display, addressing of persons, postal address formatting, and
76 telephone number handling.

77

78 There are a number of benefits coming from this Technical Report:

79

80 Rigid specification Using this Technical Report, a user can rigidly specify a
81 number of the cultural conventions that apply to the
82 information technology environment of the user.

83

84 Cultural adaptability If an application has been designed and built in a
85 culturally neutral manner, the application may use the
86 specifications as data to its APIs, and thus the same
87 application may accommodate different users in a
88 culturally acceptable way to each of the users, without
89 change of the binary application.

90

91 Productivity This Technical Report specifies those cultural
92 conventions and how to specify data for them. With that
93 data an application developer is relieved from getting the
94 different information to support all the cultural
95 environments for the expected customers of the product.
96 The application developer is thus ensured of culturally
97 correct behaviour as specified by the customer, and
98 possibly more markets may be reached as customers may
99 have the possibility to provide the data themselves for
100 markets that were not targeted.

101

102 Uniform behaviour When a number of applications share one cultural
103 specification, which may be supplied from the user or
104 provided by the application or operating system, their
105 behaviour for cultural adaptation becomes uniform.

106

107 The specification format is independent of platforms and specific encoding, and targeted to
108 be usable from a wide range of programming languages.

109

110 A number of cultural conventions, such as spelling, hyphenation rules and terminology, are
111 not specifiable with this Technical Report, but it provides mechanisms to define new
112 categories and also new keywords within existing categories. An internationalized
113 application may take advantage of information provided with the FDCC-set (such as the
114 language) to provide further internationalized services to the user.

115

116 This Technical Report defines a format compatible with the one used in the International
117 string ordering standard, ISO/IEC 14651. This Technical Report is upward compatible
118 with the ISO/IEC 9945-2:1993 POSIX shell and utilities standard, particularly its clauses
119 2.4 and 2.5. The major extensions from that text are listed in annex A. This Technical
120 Report has enhanced functionality in a number of areas such as ISO/IEC 10646 support,
121 more classification of characters, transliteration, dual (multi) currency support, enhanced

122 date and time formatting, personal name writing, postal address formatting, telephone
123 number handling, and management of categories. There is enhanced support for character
124 sets including ISO/IEC 2022 handling and an enhanced method to separate the
125 specification of cultural conventions from an actual encoding via a description of the
126 character repertoire employed. A standard set of values for all the categories has been
127 defined covering the repertoire of ISO/IEC 10646-1, as referenced in the normative
128 references clause.

129
130 The Technical report was originally scheduled for adoption as an International Standard,
131 but a number of members of ISO and IEC found the specification problematical. It was
132 then decided to convert the specification into a Technical Report type I. Annex D lists a
133 number of issues that some members of ISO and IEC have with the specification.
134

135 **Information technology —**
136 **Specification method for cultural conventions**

137
138 **1 SCOPE**

139
140 This Technical Report specifies a description format for the specification of cultural
141 conventions, a description format for character sets, and a description format for binding
142 character names to ISO/IEC 10646, plus a set of default values for some of these items.
143

144 The specification is upward compatible with POSIX locale specifications - a locale
145 conformant to POSIX specifications will also be conformant to the specifications in this
146 Technical Report, while the reverse condition will not hold. The descriptions are intended
147 to be coded in text files to be used via Application Programming Interfaces, that are
148 expected to be developed for a number of systems which comply with ISO/IEC 9945. An
149 alignment effort has been undertaken for this specification to be aligned with the revision
150 of the ISO/IEC 9945 standard expected to be published in 2002.
151

152 **2 NORMATIVE REFERENCES**

153
154 The following normative documents contain provisions which, through reference in this
155 text, constitute provisions of this Technical Report. For dated references, subsequent
156 amendments to, or revisions of, any of these publications do not apply. However, parties
157 to agreements based on this Technical Report are encouraged to investigate the possibility
158 of applying the most recent editions of the normative documents indicated below. For
159 undated references, the latest edition of the normative document referred to applies.
160 Members of ISO and IEC maintain registers of currently valid Technical Reports.
161

162 ISO 639 (all parts), *Codes for the representation of names of languages*.

163
164 ISO/IEC 2022, *Information technology - Character code structure and extension tech-*
165 *niques*.

166
167 ISO 3166 (all parts), *Codes for the representation of names of countries and their*
168 *subdivisions*.

169
170 ISO 4217, *Codes for the representation of currencies and funds*.

171
172 ISO 8601, *Data elements and interchange formats - Information interchange - Represen-*
173 *tation of dates and times*.

174
175 ISO/IEC 9945:200x (to be published), *Information technology - Portable Operating System*
176 *Interface (POSIX)*.

177
178 ISO/IEC 9945-2:1993, *Information technology - Portable Operating System Interface*
179 *(POSIX) - Part 2: Shell and Utilities*.

180
181 ISO/IEC 10646-1:1993, *Information technology - Universal Multiple-Octet Coded Cha-*
182 *acter Set (UCS) - Part 1: Architecture and Basic Multilingual Plane, including Cor.1 and*
183 *AMD 1-9 plus AMD 18*. From AMD 18 only the characters U20AC EURO SIGN and
184 UFFFC OBJECT REPLACEMENT CHARACTER are accounted for in this TR.
185

186 ISO/IEC 14651:2000, *Information technology - International string ordering - Method for*

187 *comparing character strings and description of a default tailorable ordering.*

188

189 ISO/IEC 15897:1999, *Information technology - Procedures for registration of cultural*
190 *conventions.*

191

192 **3 TERMS, DEFINITIONS AND NOTATIONS**

193

194 **3.1 Terms and definitions**

195

196 For the purposes of this Technical Report, the terms and definitions given in the following
197 apply.

198

199 **3.1.1 Bytes and characters**

200

201 **3.1.1.1**

202 **byte:**

203 An individually addressable unit of data storage that is equal to or larger than an octet,
204 used to store a character or a portion of a character.

205

206 A byte is composed of a contiguous sequence of bits, the number of which is
207 implementation defined. The least significant bit is called the low-order bit; the most
208 significant bit is called the high-order bit.

209

210 **3.1.1.2**

211 **character:**

212 A member of a set of elements used for the organization, control or representation of data.

213

214 **3.1.1.3**

215 **coded character:**

216 A sequence of one or more bytes representing a single character.

217

218 **3.1.1.4**

219 **text file:**

220 A file that contains characters organized into one or more lines.

221

222 **3.1.2 cultural and other major concepts**

223

224 **3.1.2.1**

225 **cultural convention:**

226 A data item for information technology that may vary dependent on language, territory, or
227 other cultural habits.

228

229 **3.1.2.2**

230 **FDCC**

231 A Formal Definition of a Cultural Convention, that is a cultural convention put into a
232 formal definition scheme.

233

234 **3.1.2.3**

235 **FDCC-set:**

236 A Set of Formal Definitions of Cultural Conventions (FDCC's). The definition of the
237 subset of a user's information technology environment that depends on language and
238 cultural conventions. Note: the FDCC-set is a superset of the "locale" term in C and POSIX.

- 239 **3.1.2.4**
240 **charmap:**
241 A definition of a mapping between symbolic character names and character codes, plus
242 related information.
243
- 244 **3.1.2.5**
245 **repertoiremap:**
246 A definition of a mapping between symbolic character names and characters for the
247 repertoire of characters used in a FDCC-set, further described in clause 6.
248
- 249 **3.1.3 FDCC categories related**
250
- 251 **3.1.3.1**
252 **character class:**
253 A named set of characters sharing an attribute associated with the name of the class.
254
- 255 **3.1.3.2**
256 **collation:**
257 The logical ordering of strings according to defined precedence rules.
258
- 259 **3.1.3.3**
260 **collating element:**
261 The smallest entity used to determine logical ordering.
262
263 See collating sequence. A collating element consists of either a single character, or two or
264 more characters collating as a single entity. The LC_COLLATE category in the associated
265 FDCC-set determines the set of collating elements.
266
- 267 **3.1.3.4**
268 **multicharacter collating element:**
269 A sequence of two or more characters that collate as an entity.
270
271 For example, in some languages two characters are sorted as one letter, as in the case for
272 Danish and Norwegian "aa".
273
- 274 **3.1.3.5**
275 **collating sequence:**
276 The relative order of collating elements as determined by the setting of the LC_COLLATE
277 category in the applied FDCC-set.
278
- 279 **3.1.3.6**
280 **equivalence class:**
281 A set of collating elements with the same primary collation weight.
282
283 Elements in an equivalence class are typically elements that naturally group together, such
284 as all accented letters based on the same letter.
285
286 The collation order of elements within an equivalence class is determined by the weights
287 assigned on any subsequent levels after the primary weight.
288
289
290

291 3.2 Notations

292
293 The following notations and common conventions for specifications apply to this
294 Technical Report:

296 3.2.1 Notation for defining syntax

297
298 In this Technical Report, the description of an individual record in a FDCC-set is done
299 using the syntax notation given in the following.

300 The syntax notation looks as follows:

301 " <format> ", [<arg1>, <arg2>, ..., <argn>]

302
303
304 The <format> is given in a format string enclosed in double quotes, followed by a number
305 of parameters, separated by commas. It is similar to the format specification defined in
306 clause 2.12 in the ISO/IEC 9945-2:1993 standard and the format specification used in C
307 language printf() function. The format of each parameter is given by an escape sequence
308 as follows:

309
310 % s specifies a string
311 % d specifies a decimal integer
312 % c specifies a character
313 % o specifies an octal integer
314 % x specifies a hexadecimal integer
315

316
317 A " " (an empty character position) in the syntax string represents one or more <blank>
318 characters.

319
320 All other characters in the format string except

321 %% specifies a single %
322 \n specifies an end-of-line
323

324
325 represent themselves.

326
327 The notation "... " is used to specify that repetition of the previous specification is optional,
328 and this is done in both the format string and in the parameter list.

330 3.2.3 Portable character set

331
332 A set of symbolic names for characters in Table 1, which is called the portable character
333 set, is used in character description text of this specification. The first eight entries in
334 Table 1 are defined in ISO/IEC 6429 and the rest is defined in ISO/IEC 9945-2 with some
335 definitions from ISO/IEC 10646-1.

336
337 **Table 1: Portable character set**

338 Symbolic name	339 Glyph	340 UCS	341 Description
342 <NUL>		<U0000>	NULL (NUL)
343 <alert>		<U0007>	BELL (BEL)
344 <backspace>		<U0008>	BACKSPACE (BS)
		<U0009>	CHARACTER TABULATION (HT)

345	<carriage-return>		<U000D>	CARRIAGE RETURN (CR)
346	<newline>		<U000A>	LINE FEED (LF)
347	<vertical-tab>		<U000B>	LINE TABULATION (VT)
348	<form-feed>		<U000C>	FORM FEED (FF)
349	<space>		<U0020>	SPACE
350	<exclamation-mark>	!	<U0021>	EXCLAMATION MARK
351	<quotation-mark>	"	<U0022>	QUOTATION MARK
352	<number-sign>	#	<U0023>	NUMBER SIGN
353	<dollar-sign>	\$	<U0024>	DOLLAR SIGN
354	<percent-sign>	%	<U0025>	PERCENT SIGN
355	<ampersand>	&	<U0026>	AMPERSAND
356	<apostrophe>	'	<U0027>	APOSTROPHE
357	<left-parenthesis>	(<U0028>	LEFT PARENTHESIS
358	<right-parenthesis>)	<U0029>	RIGHT PARENTHESIS
359	<asterisk>	*	<U002A>	ASTERISK
360	<plus-sign>	+	<U002B>	PLUS SIGN
361	<comma>	,	<U002C>	COMMA
362	<hyphen-minus>	-	<U002D>	HYPHEN-MINUS
363	<hyphen>	-	<U002D>	HYPHEN-MINUS
364	<full-stop>	.	<U002E>	FULL STOP
365	<period>	.	<U002E>	FULL STOP
366	<slash>	/	<U002F>	SOLIDUS
367	<solidus>	/	<U002F>	SOLIDUS
368	<zero>	0	<U0030>	DIGIT ZERO
369	<one>	1	<U0031>	DIGIT ONE
370	<two>	2	<U0032>	DIGIT TWO
371	<three>	3	<U0033>	DIGIT THREE
372	<four>	4	<U0034>	DIGIT FOUR
373	<five>	5	<U0035>	DIGIT FIVE
374	<six>	6	<U0036>	DIGIT SIX
375	<seven>	7	<U0037>	DIGIT SEVEN
376	<eight>	8	<U0038>	DIGIT EIGHT
377	<nine>	9	<U0039>	DIGIT NINE
378	<colon>	:	<U003A>	COLON
379	<semicolon>	;	<U003B>	SEMICOLON
380	<less-than-sign>	<	<U003C>	LESS-THAN SIGN
381	<equals-sign>	=	<U003D>	EQUALS SIGN
382	<greater-than-sign>	>	<U003E>	GREATER-THAN SIGN
383	<question-mark>	?	<U003F>	QUESTION MARK
384	<commercial-at>	@	<U0040>	COMMERCIAL AT
385	<A>	A	<U0041>	LATIN CAPITAL LETTER A
386		B	<U0042>	LATIN CAPITAL LETTER B
387	<C>	C	<U0043>	LATIN CAPITAL LETTER C
388	<D>	D	<U0044>	LATIN CAPITAL LETTER D
389	<E>	E	<U0045>	LATIN CAPITAL LETTER E
390	<F>	F	<U0046>	LATIN CAPITAL LETTER F
391	<G>	G	<U0047>	LATIN CAPITAL LETTER G
392	<H>	H	<U0048>	LATIN CAPITAL LETTER H
393	<I>	I	<U0049>	LATIN CAPITAL LETTER I
394	<J>	J	<U004A>	LATIN CAPITAL LETTER J
395	<K>	K	<U004B>	LATIN CAPITAL LETTER K
396	<L>	L	<U004C>	LATIN CAPITAL LETTER L
397	<M>	M	<U004D>	LATIN CAPITAL LETTER M
398	<N>	N	<U004E>	LATIN CAPITAL LETTER N
399	<O>	O	<U004F>	LATIN CAPITAL LETTER O
400	<P>	P	<U0050>	LATIN CAPITAL LETTER P
401	<Q>	Q	<U0051>	LATIN CAPITAL LETTER Q
402	<R>	R	<U0052>	LATIN CAPITAL LETTER R
403	<S>	S	<U0053>	LATIN CAPITAL LETTER S
404	<T>	T	<U0054>	LATIN CAPITAL LETTER T
405	<U>	U	<U0055>	LATIN CAPITAL LETTER U
406	<V>	V	<U0056>	LATIN CAPITAL LETTER V
407	<W>	W	<U0057>	LATIN CAPITAL LETTER W
408	<X>	X	<U0058>	LATIN CAPITAL LETTER X
409	<Y>	Y	<U0059>	LATIN CAPITAL LETTER Y
410	<Z>	Z	<U005A>	LATIN CAPITAL LETTER Z
411	<left-square-bracket>	[<U005B>	LEFT SQUARE BRACKET
412	<backslash>	\	<U005C>	REVERSE SOLIDUS
413	<reverse-solidus>	\	<U005C>	REVERSE SOLIDUS
414	<right-square-bracket>]	<U005D>	RIGHT SQUARE BRACKET

415	<circumflex-accent>	^	<U005E>	CIRCUMFLEX ACCENT
416	<circumflex>	^	<U005E>	CIRCUMFLEX ACCENT
417	<low-line>	—	<U005F>	LOW LINE
418	<underscore>	—	<U005F>	LOW LINE
419	<grave-accent>	`	<U0060>	GRAVE ACCENT
420	<a>	a	<U0061>	LATIN SMALL LETTER A
421		b	<U0062>	LATIN SMALL LETTER B
422	<c>	c	<U0063>	LATIN SMALL LETTER C
423	<d>	d	<U0064>	LATIN SMALL LETTER D
424	<e>	e	<U0065>	LATIN SMALL LETTER E
425	<f>	f	<U0066>	LATIN SMALL LETTER F
426	<g>	g	<U0067>	LATIN SMALL LETTER G
427	<h>	h	<U0068>	LATIN SMALL LETTER H
428	<i>	i	<U0069>	LATIN SMALL LETTER I
429	<j>	j	<U006A>	LATIN SMALL LETTER J
430	<k>	k	<U006B>	LATIN SMALL LETTER K
431	<l>	l	<U006C>	LATIN SMALL LETTER L
432	<m>	m	<U006D>	LATIN SMALL LETTER M
433	<n>	n	<U006E>	LATIN SMALL LETTER N
434	<o>	o	<U006F>	LATIN SMALL LETTER O
435	<p>	p	<U0070>	LATIN SMALL LETTER P
436	<q>	q	<U0071>	LATIN SMALL LETTER Q
437	<r>	r	<U0072>	LATIN SMALL LETTER R
438	<s>	s	<U0073>	LATIN SMALL LETTER S
439	<t>	t	<U0074>	LATIN SMALL LETTER T
440	<u>	u	<U0075>	LATIN SMALL LETTER U
441	<v>	v	<U0076>	LATIN SMALL LETTER V
442	<w>	w	<U0077>	LATIN SMALL LETTER W
443	<x>	x	<U0078>	LATIN SMALL LETTER X
444	<y>	y	<U0079>	LATIN SMALL LETTER Y
445	<z>	z	<U007A>	LATIN SMALL LETTER Z
446	<left-brace>	{	<U007B>	LEFT CURLY BRACKET
447	<left-curly-bracket>	{	<U007B>	LEFT CURLY BRACKET
448	<vertical-line>		<U007C>	VERTICAL LINE
449	<right-brace>	}	<U007D>	RIGHT CURLY BRACKET
450	<right-curly-bracket>	}	<U007D>	RIGHT CURLY BRACKET
451	<tilde>	~	<U007E>	TILDE

452

453 This Technical Report may use other symbolic character names than the above in
 454 examples, to illustrate the use of the range of symbols allowed by the syntax specified in
 455 4.1.1.

456

457 4 FDCC-set

458

459 A FDCC-set is the definition of the subset of a user's information technology environment
 460 that depends on language and cultural conventions. It is made up from one or more
 461 categories. Each category is identified by its name and controls specific aspects of the
 462 behaviour of components of the system. This Technical Report defines the following
 463 categories:

464

465	LC_IDENTIFICATION	Versions and status of categories
466	LC_CTYPE	Character classification, case conversion and code transformation.
467		
468	LC_COLLATE	Collation order.
469	LC_TIME	Date and time formats.
470	LC_NUMERIC	Numeric, non-monetary formatting.
471	LC_MONETARY	Monetary formatting.
472	LC_MESSAGES	Formats of informative and diagnostic messages and interactive responses.
473		
474	LC_XLITERATE	Character transliteration.
475	LC_NAME	Format of writing personal names.
476	LC_ADDRESS	Format of postal addresses.

477 LC_TELEPHONE Format for telephone numbers, and other telephone
478 information.
479

480 Note: In future editions of this Technical Report further categories may be added.
481

482 Other category names beginning with the 3 characters "LC_" are reserved for future
483 standardization, except for category names beginning with the five characters "LC_X_"
484 which is not used for future addition of categories specified in this Technical Report. An
485 application may thus use category names beginning with the five characters "LC_X_" for
486 application defined categories to avoid clashes with future standardized categories.
487

488 This Technical Report also defines an FDCC-set named "i18n" with values for some of
489 the above categories in order to simplify FDCC-set descriptions for a number of cultures.
490 The contents of "i18n" categories should not necessarily be considered as the most
491 commonly accepted values, while in many cases it could be the recommended values.
492

493 4.1 FDCC-set description 494

495 FDCC-sets are described with the syntax presented in this subclause. For the purposes of
496 this Technical Report, the text is referred to as the FDCC-set definition text or FDCC-set
497 source text.
498

499 The **FDCC-set definition text** contains one or more FDCC-set category source definitions,
500 and does not contain more than one definition for the same FDCC-set category. If the text
501 contains source definitions for more than one category, application-defined categories, if
502 present, appears after the categories defined by this clause. A category source definition
503 contains either the definition of a category or a copy directive. In the event that some of
504 the information for a FDCC-set category, as specified in this Technical Report, is missing
505 from the FDCC-set source definition, the behaviour of that category, if it is referenced, is
506 unspecified. A FDCC-set category is the normal way of specifying a single FDCC.
507

508 There are no **naming conventions** for FDCC-sets specified in this Technical Report, but
509 clause 6.8 in ISO/IEC 15897:1999 specifies naming rules for POSIX locales, charmaps
510 and repertoire maps, that may also be applied to FDCC-sets, charmaps and repertoire maps
511 specified according to this Technical Report.
512

513 A **category source definition** consists of a category header, a category body, and a
514 category trailer. A category header consists of the character string naming of the category,
515 beginning with the characters "LC_". The category trailer consists of the string "END",
516 followed by one or more "blank"s and the string used in the corresponding category
517 header.
518

519 The **category body** consists of one or more lines of text. Each line is one of the
520 following:
521

- 522 - a line containing an identifier, optionally followed by one or more operands. Identifiers
523 are either keywords, identifying a particular FDCC, or collating elements, or section
524 symbols,
- 525 - one of transliteration statements defined in 4.3.
526

527 In addition to the keywords defined in this Technical Report, the source can contain
528 application-defined keywords. Each **keyword** within a category has a unique name (i.e.,

two categories can have a commonly-named keyword); no keyword starts with the characters "LC_". Identifiers are separated from the operands by one or more "blank"s.

Operands are characters, collating elements, section symbols, or strings of characters. Strings are enclosed in double-quotes. Literal double-quotes within strings are preceded by the <escape character>, described below. When a keyword is followed by more than one operand, the operands are separated by semicolons; "blank"s are allowed before and/or after a semicolon.

4.1.1 Character representation

Individual characters, characters in strings, and collating elements are represented using symbolic names, UCS notation or characters themselves, or as octal, hexadecimal, or decimal constants as defined below. When constant notation is used, the resultant FDCC-set definitions need not be portable between systems.

(0) The left angle bracket (<) is a reserved symbol, denoting the start of a symbolic name; when used to represent itself outside a symbolic name it is preceded by the escape character.

(1) A character can be represented via a **symbolic name**, enclosed within angle brackets (< and >). The symbolic name, including the angle brackets, exactly matches a symbolic name defined in a charmap or a repertoiremap to be used, and is replaced by a character value determined from the value associated with the symbolic name in the charmap or a value associated via a repertoiremap. Repertoiremaps have predefined symbolic names for UCS characters, see clause 6. A FDCC-set may also use the UCS notation of clause 6 to represent characters, without a repertoiremap being defined for the FDCC-set. Use of the escape character or a right angle bracket within a symbolic name is invalid unless the character is preceded by the escape character.

Example: <c>;<c-cedilla> "<M><a><y>"

The items (2), (3), (4) and (5) are deprecated and are retained for compatibility with the POSIX standard. FDCC-sets should be specified in a coded character set independent way, using symbolic names. To make actual use of the FDCC-set, it is used together with charmaps and/or repertoiremaps, so that the symbolic character names can be resolved into the actual character encoding used.

(2) A character can be represented by the character itself, in which case the value of the character is application-defined. Within a string, the double-quote character, the escape character, and the right angle bracket character are escaped (preceded by the escape character) to be interpreted as the character itself. Outside strings, the characters

, ; < > escape_char

581 are escaped by the escape character to be interpreted as the
582 character itself.

583 Example: c ä "May"
584

- 585
586 (3) A character can be represented as an octal constant. An octal
587 constant is specified as the escape character followed by two
588 or more octal digits. Each constant represents a byte value.

589 Example: \143; \347; "\115"
590

- 591
592 (4) A character can be represented as a hexadecimal constant. A
593 hexadecimal constant is specified as the escape character
594 followed by an x followed by two or more hexadecimal
595 digits. Each constant represents a byte value.

596 Example: \x63;\xe7;
597

- 598
599 (5) A character can be represented as a decimal constant. A
600 decimal constant is specified as the escape character
601 followed by a d followed by two or more decimal digits.
602 Each constant represents a byte value.

603 Example: \d99; \d231;
604

- 605
606 (6) Multibyte characters can be represented by concatenated
607 constants specified in byte order with the last constant
608 specifying the least significant byte of the character.
609 Concatenated constants can include a mix of the above
610 character representations.

611 Example: \143\xe7; "\115\xe7\d171"
612

613
614 Only characters existing in the character set for which the FDCC-set definition is created
615 are specified, whether using symbolic names, the characters themselves, or octal, decimal,
616 or hexadecimal constants. If a charmap is present, only characters defined in the charmap
617 can be specified using octal, decimal, or hexadecimal constants. Symbolic names not
618 present in the charmap can be specified and are ignored, as specified under item (1)
619 above.

620
621 Note: The <character> symbolic character notation is recommended for use of specifying
622 all characters in a FDCC-set, to facilitate portability of the FDCC-sets, as the coded
623 character set of the application of the FDCC-set may be different from the coded character
624 set of the FDCC-set source. This is also recommended for format effectors in strings, such
625 as in LC_DATE or LC_ADDRESS, where the format effectors are allowed to be stored
626 together with the rest of the string, in a binary string with a different encoding from that
627 of the source FDCC-set.

628 4.1.2 Continuation of lines 629

630
631 A line in a specification can be continued by placing an escape character as the last visible
632 graphic character on the line; this continuation character is discarded from the input. The
633 line is continued to the next non-comment line.

634 4.1.3 Names for copy keyword

635

636 In most of the categories a "copy" keyword is allowed. The name specified with this copy
637 keyword is one of:

638

- 639 - "i18n" which indicate the "i18n" FDCC-set defined in this specification,
- 640 - the name of a FDCC-set or POSIX locale registered by the process defined in ISO/IEC
641 15897,
- 642 - any other name which may be recognized in some local context - not being
643 recommended as an international specification.

644

645 4.1.4 Pre-category statements

646

647 In a FDCC-set the following statements can precede category specifications, and they
648 apply to all categories in the specified FDCC-set.

649

650 4.1.4.1 comment_char

651

652 The following line in a FDCC-set modifies the comment character. It has the following
653 syntax, starting in column 1:

654

```
655 "comment_char %c\n", <comment_character>
```

656

657 The comment character defaults to the number-sign (#). All examples in this Technical
658 Report use "%" as the <comment_character>, except where otherwise noted. Blank lines
659 and lines containing the <comment_character> in the first position are ignored. In collating
660 statements a <comment_character> occurring where the delimiter ";" may occur,
661 terminates the collating statement.

662

663 4.1.4.2 escape_char

664

665 The following line in a FDCC-set modifies the escape character to be used in the text. It
666 has the following syntax, starting in column 1:

667

```
668 "escape_char %c\n", <escape_character>
```

669

670 The escape character is used for representing characters in 4.1.1 and for continuing lines.
671 The escape character defaults to backslash "\". All examples in this Technical Report uses
672 "/" as the escape character, except where otherwise noted.

673

674 4.1.4.3 repertoiremap

675

676 The following line in a FDCC-set specifies the name of a repertoiremap used to define the
677 symbolic character names in the FDCC-set. There may be at most one "repertoiremap"
678 line. It has the following syntax, starting in column 1:

```
679 "repertoiremap %s\n", <repertoiremap>
```

680

681 The name is one of:

- 682 - "i18nrep" which indicates the "i18nrep" repertoiremap defined in this specification,
- 683 - the name of a <repertoiremap> registered by the process defined in ISO/IEC 15897,
- 684 - any other name which may be recognized in some local context - not being
685 recommended as an international specification.

686 **4.1.4.4 charmap**

687

688 The following line in a FDCC-set specifies the name of a charmap which may be used
689 with the FDCC-set. It has the following syntax, starting in column 1:

690

691 "charmap %s\n",<charmap>

692

693 This keyword gives a hint on which charmaps a FDCC-set is meant to be supported by.
694 There may be more than one charmap specification useful with a FDCC-set. It is an
695 application's responsibility to decide what charmap specification is to be used with that
696 application.

697

698 The name is one of:

699

- 700 - the name of a <charmap> registered by the process defined in ISO/IEC 15897,
- 701 - any other name which may be recognized in some local context - not being
702 recommended as an international specification.

702

703 **4.2 LC_IDENTIFICATION**

704

705 The LC_IDENTIFICATION category defines properties of the FDCC-set, and which
706 specification methods the FDCC-set is conforming to. Values must be supplied for all
707 unless otherwise noted, and the operands are strings. The following keywords are defined:

708

709	title	Title of the FDCC-set.
710	source	Organization name of provider of the source.
711	address	Organization postal address.
712	contact	Name of contact person. This keyword is optional.
713	email	Electronic mail address of the organization, or contact 714 person. This keyword is optional.
715	tel	Telephone number for the organization, in international 716 format. This keyword is optional.
717	fax	Fax number for the organization, in international format. 718 This keyword is optional.
719	language	Natural language to which the FDCC-set applies, as specified 720 in ISO 639. If a two-letter code exists for this language, it is 721 used, else the three-letter code is used. This keyword is 722 optional.
723	territory	The geographic extent where the FDCC-set applies (where 724 applicable), as two-letter form of ISO 3166. This keyword is 725 optional.
726	audience	If not for general use, an indication of the intended user 727 audience. This keyword is optional.
728	application	If for use of a special application, a description of the 729 application. This keyword is optional.
730	abbreviation	Short name for provider of the source. This keyword is 731 optional.
732	revision	Revision number consisting of digits and zero or more full 733 stops (".").
734	date	Revision date in the format according to this example: 735 "1995-02-05" meaning the 5th of February, 1995.

736

737

738 If required information is not present in ISO 639 or ISO 3166, the string should be given
 739 as empty, and the relevant Maintenance Authority should be approached to get the needed
 740 item registered.

741
 742 Note: Only one language per territory can be addressed with a single FDCC-set; an
 743 additional FDCC-set is required for each additional language for that territory.

744
 745 **category** Is used to define that a category is present and what
 746 specification the category is claiming conformance to. The
 747 first operand is a string in double-quotes that describes the
 748 specification that the category is claiming conformance to,
 749 and the following values are defined:
 750 "i18n:2002"
 751 "posix:1993"
 752 The second operand is a string with the category name,
 753 where the category names of clause 4 are defined. More than
 754 one "category" keyword may be given, but only one per
 755 category name.

756
 757 The "i18n" LC_IDENTIFICATION category is:

```

758 LC_IDENTIFICATION
759 % This is the ISO/IEC TR 14652 "i18n" definition for
760 % the LC_IDENTIFICATION category.
761 %
762 title "ISO/IEC TR 14652 i18n FDCC-set"
763 source "ISO/IEC Copyright Office"
764 address "Case postale 56, CH-1211 Geneve 20, Switzerland"
765 contact ""
766 email ""
767 tel ""
768 fax ""
769 language ""
770 territory ""
771 revision "1.0"
772 date "2001-12-08"
773 %
774 category "i18n:2002";LC_IDENTIFICATION
775 category "i18n:2002";LC_CTYPE
776 category "i18n:2002";LC_COLLATE
777 category "i18n:2002";LC_TIME
778 category "i18n:2002";LC_NUMERIC
779 category "i18n:2002";LC_MONETARY
780 category "i18n:2002";LC_MESSAGES
781 category "i18n:2002";LC_NAME
782 category "i18n:2002";LC_ADDRESS
783 category "i18n:2002";LC_TELEPHONE
784
785 END LC_IDENTIFICATION
  
```

787 788 789 **4.3 LC_CTYPE**

790
 791 The LC_CTYPE category defines character classification, case conversion, character
 792 transformation, and other character attribute mappings. Support for the portable character
 793 set is required.

794
 795 A series of characters in a specification can be represented by the hexadecimal symbolic
 796 ellipsis symbol ".." (two dots), the decimal symbolic ellipses symbols "...." (4 dots), or the
 797 absolute ellipses "... (3 dots).

798 The **hexadecimal symbolic ellipsis** ("..") specification is only valid between symbolic
 799 character names. The symbolic names consists of zero or more nonnumeric characters
 800 from the set shown with visible glyphs in Table 1, followed by an integer formed by one
 801 or more hexadecimal digits, using uppercase letters only for the range "A" to "F". The
 802 characters preceding the hexadecimal integer are identical in the two symbolic names, and
 803 the integer formed by the hexadecimal digits in the second symbolic name are identical to
 804 or greater than the integer formed by the hexadecimal digits in the first name. This is
 805 interpreted as a series of symbolic names formed from the common part and each of the
 806 integers in hexadecimal format using uppercase letters only between the first and the
 807 second integer, inclusive, and with a length of the symbolic names generated that is equal
 808 to the length of the first (and also the second) symbolic name. As an example,
 809 <U010E>..<U0111> is interpreted as the symbolic names <U010E>, <U010F>, <U0110>,
 810 and <U0111>, in that order.

811
 812 The **decimal symbolic ellipsis** ("...") specification is only valid between symbolic
 813 character names. The symbolic names consist of zero or more nonnumeric characters from
 814 the set shown with visible glyphs in Table 1, followed by an integer formed by one or
 815 more decimal digits. The characters preceding the decimal integer are identical in the two
 816 symbolic names, and the integer formed by the decimal digits in the second symbolic
 817 name is identical to or greater than the integer formed by the decimal digits in the first
 818 name. This is interpreted as a series of symbolic names formed from the common part and
 819 each of the integers in decimal format between the first and the second integer, inclusive,
 820 and with a length of the symbolic names generated that is equal to the length of the first
 821 (and also the second) symbolic name. As an example, <j0101>...<j0104> is interpreted as
 822 the symbolic names <j0101>, <j0102>, <j0103>, and <j0104>, in that order.

823
 824 The **absolute ellipsis** specification is only valid within a single encoded character set. An
 825 ellipsis is interpreted as including in the list all characters with an encoded value higher
 826 than the encoded value of the character preceding the ellipsis and lower than the encoded
 827 value of the character following the ellipsis. The absolute ellipsis specification is
 828 deprecated, as this is only relevant to FDCC-sets not using symbolic characters.
 829 As an example, \x30;...;\x39 includes in the character class all characters with encoded
 830 values between the endpoints.

831 832 **4.3.1 Character classification keywords**

833
 834 The following keywords are recognized. In the descriptions, the term "automatically
 835 included" means that it is not an error to either include the referenced characters or to
 836 omit them; the interpreting system provides them if missing and accept them silently if
 837 present.

838
 839 **copy** Specify the name of an existing FDCC-set to be used as the source for the
 840 definition of this category. If this keyword is specified, no other keyword is
 841 specified.
 842 **upper** Define characters to be classified as uppercase letters. No character
 843 specified for the keywords "cntrl", "digit", "punct", or "space" is specified.
 844 The uppercase letters A through Z of the portable character set,
 845 automatically belong to this class, with application-defined character values.
 846 The keyword may be omitted.
 847 **lower** Define characters to be classified as lowercase letters. No character
 848 specified for the keywords "cntrl", "digit", "punct", or "space" is specified.
 849 The lowercase letters a through z of the portable character set, automatically

850		belong to this class, with application-defined character values. The keyword
851		may be omitted.
852	alpha	Define characters to be classified as used to spell out the words for natural
853		languages; such as letters, syllabic or ideographic characters. No character
854		specified for the keywords "cntrl", "digit", "punct", or "space" is specified.
855		In addition, characters classified as either "upper" or "lower" automatically
856		belong to this class. The keyword may be omitted.
857	digit	Define the characters to be classified as decimal digits. Digits corresponding
858		to the values 0, 1, 2, 3, 4, 5, 6, 7, 8, and 9 can be specified in groups of 10
859		digits, and in ascending order of the values they represent. The digits of the
860		portable character set are automatically included. If this keyword is not
861		specified, the digits 0 through 9 of the portable character set automatically
862		belong to this class, with application-defined character values. The "digit"
863		keyword is used to specify which characters are accepted as digits in input
864		to an application, such as characters typed in or scanned in from an input
865		text file, and should list digits used with all the scripts supported by the
866		FDCC-set. The keyword may be omitted.
867	alnum	Define the characters to be classified as used to spell out the words for
868		natural languages, and numeric digits. The characters of the "alpha" and
869		"digit" classes are automatically included in this class. The keyword may be
870		omitted.
871	outdigit	Define the characters to be classified as decimal digits for output from an
872		application, such as to a printer or a display or a output text file. Decimal
873		digits corresponding to the values <0>, <1>, <2>, <3>, <4>, <5>, <6>, <7>
874		<8>, and <9> can be specified, and in ascending order of the values they
875		represent. The intended use is for all places where decimal digits are used
876		for output, including numeric and monetary formatting, and date and time
877		formatting. Only one set of 10 decimal digits may be specified. If this
878		keyword is not specified, the decimal digits 0 through 9 of the portable
879		character set automatically belong to this class, with application-defined
880		character values. The keyword may be omitted.
881	blank	Define characters to be classified as "blank" characters. If this keyword is
882		unspecified, the characters <space> and <tab>, with application-defined
883		character values, belong to this character class.
884	space	Define characters to be classified as white-space characters, to find
885		syntactical boundaries. No character specified for the keywords "upper",
886		"lower", "alpha", "digit", "graph", or "xdigit" is specified. If this keyword is
887		not specified, the characters <space>, <form-feed>, <newline>, <carriage-
888		return>, <tab>, and <vertical-tab>, automatically belong to this class, with
889		application-defined character values. Any characters included in the class
890		"blank" are automatically included. The class should not include the NO-
891		BREAK spaces characters <U00A0>, <U2007>, <UFEFF>, as these
892		characters should not be used for word boundaries. The keyword may be
893		omitted.
894	cntrl	Define characters to be classified as control characters. No character
895		specified for the keywords "upper", "lower", "alpha", "digit", "punct",
896		"graph", "print", or "xdigit" is specified. The keyword is specified.
897	punct	Define characters to be classified as punctuation characters. No character
898		specified for the keywords "upper", "lower", "alpha", "digit", "cntrl",
899		"xdigit", or as the <space> character is specified. The keyword is specified.
900	xdigit	Define the characters to be classified as hexadecimal digits. Only the
901		characters defined for the class "digit" are specified, in ascending sequence

902		by numerical value, followed by sets of six characters representing the
903		hexadecimal digits 10 through 15 in ascending order (for example <A>, , <C>, <D>, <E>, <F>, <a>, , <c>, <d>, <e>, <f>). The digits <0> through <9>, the uppercase letters <A> through <F>, and the lowercase letters <a> through <f>, automatically belong to this class, with application-defined character values.
906		
907		
908	graph	Define characters to be classified as printable characters, not including the <space> character. If this keyword is not specified, characters specified for the keywords "upper", "lower", "alpha", "digit", "xdigit", and "punct" belong to this character class. No character specified for the keyword "cntrl" is specified.
909		
910		
911		
912	print	Define characters to be classified as printable characters, including the <space> character. If this keyword is not provided, characters specified for the keywords upper, lower, alpha, digit, xdigit, punct, graph, and the <space> character belong to this character class. No character specified for the keyword "cntrl" is specified.
913		
914		
915		
916		
917	toupper	Define the mapping of lowercase letters to uppercase letters. The operand consists of character pairs, separated by semicolons. The characters in each character pair are separated by a comma and the pair enclosed by parentheses. The first character in each pair is the lowercase letter, the second the corresponding uppercase letter. Only characters specified for the keywords "lower" and "upper" are specified. If this keyword is not specified, the lowercase letters <a> through <z>, and their corresponding uppercase letters <A> through <Z>, are automatically included, with application-defined character values.
918		
919		
920		
921		
922		
923		
924		
925		
926	tolower	Define the mapping of uppercase letters to lowercase letters. The operand consists of character pairs, separated by semicolons. The characters in each character pair are separated by a comma and the pair enclosed by parentheses. The first character in each pair is the uppercase letter, the second the corresponding lowercase letter. Only characters specified for the keywords "lower" and "upper" are specified. If this keyword is specified, the uppercase letters <A> through <Z>, and their corresponding lowercase letter, are specified. If this keyword is not specified, the mapping is the reverse mapping of the one specified for toupper.
927		
928		
929		
930		
931		
932		
933		
934		
935	class	(Controversial) Define characters to be classified in the class with the name given in the first operand, which is a string. This string only contains characters of the portable character set that either has the string "LETTER" in its description, or is a digit or <hyphen-minus> or <low-line>. The following operands are characters. This keyword is optional. The keyword can only be specified once per named class. The following two names are recognized:
936		
937		
938		
939		
940		
941		
942		
943		
944		
945		
946		
947		
948		
949		
950	width	(Controversial) Define the column width of characters, for example for use of the C function wwidth(). The operands are first a list for characters, possibly using various ellipses, and semicolon separated, then a <colon>, and then the width of these characters given as an unsigned positive integer.
951		
952		
953		

954 Such width-lists separated by <semicolon> may be given for the various
 955 widths. The default value of width of characters in class "cntrl" and class
 956 "combining" is 0, else the default value of width is 1. A width for a
 957 character may be overridden by a WIDTH specification in a charmap. This
 958 keyword is optional.

959 **map** (Controversial) Define the mapping of characters to other characters. The
 960 first operand is a string, defining the name of the mapping. The string only
 961 contains letters, digits and <hyphen-minus> and <low-line> from the
 962 portable character set. The following operands consist of character pairs,
 963 separated by semicolons. The characters in each character pair are separated
 964 by a comma and the pair enclosed by parentheses. The first character in
 965 each pair is the character to map from, the second the corresponding
 966 character to map to. This keyword is optional. The keyword can only be
 967 specified once per named mapping.

968
 969 The mapping names "toupper", and "tolower" are taken to mean the
 970 mapping defined by the respective keywords.

971
 972 Example of use of the "map" keyword:

```
973 map "kana",(<U30AB>,<U304B>);(<U30AC>,<U304C>);(<U30AD>,<U304D>)
```

974
 975 This example introduces a new mapping "kana" that maps three Katakana characters to corresponding Hiragana
 976 characters.
 977

978
 979 Table 2 shows the allowed character class combinations.

980
 981 **Table 2: Valid Character Class Combinations**

982 Class	upper	lower	alpha	digit	space	cntrl	punct	graph	print	xdigit	blank
983 upper		+	A	x	x	x	x	A	A	+	x
984 lower	+		A	x	x	x	x	A	A	+	x
985 alpha	+	+		x	x	x	x	A	A	+	x
986 digit	x	x	x		x	x	x	A	A	A	x
987 space	x	x	x	x		+	*	*	*	x	+
988 cntrl	x	x	x	x	+		x	x	x	x	+
989 punct	x	x	x	x	+	x		A	A	x	+
990 graph	+	+	+	+	+	x	+		A	+	+
991 print	+	+	+	+	+	x	+	+		+	+
992 xdigit	+	+	+	+	x	x	x	A	A		x
993 blank	x	x	x	x	A	+	*	*	*	x	

994
 995 Note 1: Explanation of codes:

996 A Automatically included; see text

997 + Permitted

998 x Mutually exclusive

999 * See note 2

1000
 1001 Note 2: The <space> character, which is part of the "space" and "blank" class, cannot belong to "punct" or
 1002 "graph", but automatically belong to the "print" class. Other "space" or "blank" characters can be classified
 1003 as "punct", "graph", and/or "print".
 1004
 1005
 1006
 1007
 1008

1009 4.3.2 "i18n" LC_CTYPE category

1010

1011 The "i18n" FDCC-set for the LC_CTYPE is defined as follows:

1012

1013

LC_CTYPE

1014

% The following is the ISO/IEC TR 14652 i18n fdcc-set LC_CTYPE category.

1015

% It covers ISO/IEC 10646-1 including Cor.1 and AMD 1 thru 9

1016

% COLLECTION numbers and names are from ISO/IEC 10646-1 Annex A

1017

%

1018

% The "upper" class reflects the uppercase characters of class "alpha"

1019

upper /

1020

% COLLECTION 1 BASIC LATIN/

1021

<U0041>..<U005A>;/

1022

% COLLECTION 2 LATIN-1 SUPPLEMENT/

1023

<U00C0>..<U00D6>;<U00D8>..<U00DE>;/

1024

% COLLECTION 3 LATIN EXTENDED-A/

1025

<U0100>;<U0102>;<U0104>;<U0106>;<U0108>;<U010A>;<U010C>;<U010E>;/

1026

<U0110>;<U0112>;<U0114>;<U0116>;<U0118>;<U011A>;<U011C>;<U011E>;/

1027

<U0120>;<U0122>;<U0124>;<U0126>;<U0128>;<U012A>;<U012C>;<U012E>;/

1028

<U0130>;<U0132>;<U0134>;<U0136>;/

1029

<U0139>;<U013B>;<U013D>;<U013F>;/

1030

<U0141>;<U0143>;<U0145>;<U0147>;/

1031

<U014A>;<U014C>;<U014E>;/

1032

<U0150>;<U0152>;<U0154>;<U0156>;<U0158>;<U015A>;<U015C>;<U015E>;/

1033

<U0160>;<U0162>;<U0164>;<U0166>;<U0168>;<U016A>;<U016C>;<U016E>;/

1034

<U0170>;<U0172>;<U0174>;<U0178>;/

1035

<U0179>;<U017B>;<U017D>;/

1036

% COLLECTION 4 LATIN EXTENDED-B/

1037

<U0181>;<U0182>;<U0184>;<U0186>;<U0187>;/

1038

<U0189>..<U018B>;<U018E>..<U0191>;<U0193>;<U0194>;/

1039

<U0196>..<U0198>;<U019C>;<U019D>;<U019F>;/

1040

<U01A0>;<U01A2>;<U01A4>;<U01A6>;/

1041

<U01A7>;<U01A9>;<U01AC>;<U01AE>;<U01AF>;<U01B1>..<U01B3>;/

1042

<U01B5>;<U01B7>;<U01B8>;<U01BC>;<U01C4>;<U01C5>;<U01C7>;<U01C8>;/

1043

<U01CA>;<U01CB>;/

1044

<U01CD>;<U01CF>;<U01D1>;<U01D3>;<U01D5>;<U01D7>;<U01D9>;<U01DB>;/

1045

<U01DE>;<U01E0>;<U01E2>;<U01E4>;<U01E6>;<U01E8>;<U01EA>;<U01EC>;<U01EE>;/

1046

<U01F1>;<U01F2>;<U01F4>;<U01FA>;<U01FC>;<U01FE>;/

1047

<U0200>;<U0202>;<U0204>;<U0206>;<U0208>;<U020A>;<U020C>;<U020E>;/

1048

<U0210>;<U0212>;<U0214>;<U0216>;/

1049

% COLLECTION 8 BASIC GREEK/

1050

<U0386>;<U0388>..<U038A>;<U038C>;<U038E>;<U038F>;<U0391>..<U03A1>;/

1051

<U03A3>..<U03AB>;<U03D2>..<U03D4>;/

1052

% COLLECTION 9 GREEK SYMBOLS AND COPTIC/

1053

<U03E2>;<U03E4>;<U03E6>;<U03E8>;<U03EA>;<U03EC>;<U03EE>;/

1054

% COLLECTION 10 CYRILLIC/

1055

<U0401>..<U040C>;<U040E>..<U042F>;/

1056

<U0460>;<U0462>;<U0464>;<U0466>;<U0468>;<U046A>;<U046C>;<U046E>;/

1057

<U0470>;<U0472>;<U0474>;<U0476>;<U0478>;<U047A>;<U047C>;<U047E>;/

1058

<U0480>;/

1059

<U0490>;<U0492>;<U0494>;<U0496>;<U0498>;<U049A>;<U049C>;<U049E>;/

1060

<U04A0>;<U04A2>;<U04A4>;<U04A6>;<U04A8>;<U04AA>;<U04AC>;<U04AE>;/

1061

<U04B0>;<U04B2>;<U04B4>;<U04B6>;<U04B8>;<U04BA>;<U04BC>;<U04BE>;/

1062

<U04C1>;<U04C3>;<U04C7>;<U04CB>;/

1063

<U04D0>;<U04D2>;<U04D4>;<U04D6>;<U04D8>;<U04DA>;<U04DC>;<U04DE>;/

1064

<U04E0>;<U04E2>;<U04E4>;<U04E6>;<U04E8>;<U04EA>;<U04EE>;/

1065

<U04F0>;<U04F2>;<U04F4>;<U04F8>;/

1066

% COLLECTION 11 ARMENIAN/

1067

<U0531>..<U0556>;/

1068

% COLLECTION 28 GEORGIAN EXTENDED/

1069

<U10A0>..<U10C5>;/

1070

% COLLECTION 30 LATIN EXTENDED ADDITIONAL/

1071

<U1E00>;<U1E02>;<U1E04>;<U1E06>;<U1E08>;<U1E0A>;<U1E0C>;<U1E0E>;/

1072

<U1E10>;<U1E12>;<U1E14>;<U1E16>;<U1E18>;<U1E1A>;<U1E1C>;<U1E1E>;/

1073

<U1E20>;<U1E22>;<U1E24>;<U1E26>;<U1E28>;<U1E2A>;<U1E2C>;<U1E2E>;/

1074

<U1E30>;<U1E32>;<U1E34>;<U1E36>;<U1E38>;<U1E3A>;<U1E3C>;<U1E3E>;/

1075

<U1E40>;<U1E42>;<U1E44>;<U1E46>;<U1E48>;<U1E4A>;<U1E4C>;<U1E4E>;/

1076

<U1E50>;<U1E52>;<U1E54>;<U1E56>;<U1E58>;<U1E5A>;<U1E5C>;<U1E5E>;/

1077

<U1E60>;<U1E62>;<U1E64>;<U1E66>;<U1E68>;<U1E6A>;<U1E6C>;<U1E6E>;/

1078

<U1E70>;<U1E72>;<U1E74>;<U1E76>;<U1E78>;<U1E7A>;<U1E7C>;<U1E7E>;/

1079

<U1E80>;<U1E82>;<U1E84>;<U1E86>;<U1E88>;<U1E8A>;<U1E8C>;<U1E8E>;/

1080

<U1E90>;<U1E92>;<U1E94>;/

1081

<U1EA0>;<U1EA2>;<U1EA4>;<U1EA6>;<U1EA8>;<U1EAA>;<U1EAC>;<U1EAE>;/

1082

<U1EB0>;<U1EB2>;<U1EB4>;<U1EB6>;<U1EB8>;<U1EBA>;<U1EBC>;<U1EBE>;/

1083

<U1EC0>;<U1EC2>;<U1EC4>;<U1EC6>;<U1EC8>;<U1ECA>;<U1ECC>;<U1ECE>;/

1084

<U1ED0>;<U1ED2>;<U1ED4>;<U1ED6>;<U1ED8>;<U1EDA>;<U1EDC>;<U1EDE>;/

```

1085     <U1EE0>;<U1EE2>;<U1EE4>;<U1EE6>;<U1EE8>;<U1EEA>;<U1EEC>;<U1EEE>;/
1086     <U1EF0>;<U1EF2>;<U1EF4>;<U1EF6>;<U1EF8>;/
1087 % COLLECTION 31 GREEK EXTENDED/
1088     <U1F08>..<U1F0F>;<U1F18>..<U1F1D>;<U1F28>..<U1F2F>;<U1F38>..<U1F3F>;/
1089     <U1F48>..<U1F4D>;<U1F59>;<U1F5B>;<U1F5D>;<U1F5F>;<U1F68>..<U1F6F>;/
1090     <U1F88>..<U1F8F>;<U1F98>..<U1F9F>;<U1FA8>..<U1FAF>;<U1FB8>..<U1FBC>;/
1091     <U1FC8>..<U1FCC>;<U1FD8>..<U1FDB>;<U1FE8>..<U1FEC>;<U1FF8>..<U1FFC>
1092 % COLLECTION 28 GEORGIAN EXTENDED is not addressed as the letters does not
1093 %
1094     have a uppercase/lowercase relation
1095 %
1096 % The "lower" class reflects the lowercase characters of class "alpha"
1097 lower /
1098 % COLLECTION 1 BASIC LATIN/
1099     <U0061>..<U007A>;/
1100 % COLLECTION 2 LATIN-1 SUPPLEMENT/
1101     <U00DF>..<U00F6>;<U00F8>..<U00FF>;/
1102 % COLLECTION 3 LATIN EXTENDED-A/
1103     <U0101>;<U0103>;<U0105>;<U0107>;<U0109>;<U010B>;<U010D>;<U010F>;/
1104     <U0111>;<U0113>;<U0115>;<U0117>;<U0119>;<U011B>;<U011D>;<U011F>;/
1105     <U0121>;<U0123>;<U0125>;<U0127>;<U0129>;<U012B>;<U012D>;<U012F>;/
1106     <U0131>;<U0133>;<U0135>;<U0137>;/
1107     <U0138>;<U013A>;<U013C>;<U013E>;<U0140>;<U0142>;<U0144>;<U0146>;<U0148>;/
1108     <U0149>;<U014B>;<U014D>;<U014F>;/
1109     <U0151>;<U0153>;<U0155>;<U0157>;<U0159>;<U015B>;<U015D>;<U015F>;/
1110     <U0161>;<U0163>;<U0165>;<U0167>;<U0169>;<U016B>;<U016D>;<U016F>;/
1111     <U0171>;<U0173>;<U0175>;/
1112     <U0177>;<U017A>;<U017C>;<U017E>;<U017F>;/
1113 % COLLECTION 4 LATIN EXTENDED-B/
1114     <U0180>;<U0183>;<U0185>;<U0188>;<U018C>;<U018D>;<U0192>;<U0195>;/
1115     <U0199>..<U019B>;<U019E>;<U01A1>;<U01A3>;<U01A5>;<U01A8>;<U01AB>;<U01AD>;/
1116     <U01B0>;<U01B4>;<U01B6>;<U01B9>;<U01BA>;<U01BD>;<U01C5>;<U01C6>;/
1117     <U01C8>;<U01C9>;<U01CB>;/
1118     <U01CC>;<U01CE>;<U01D0>;<U01D2>;<U01D4>;<U01D6>;<U01D8>;<U01DA>;<U01DC>;/
1119     <U01DD>;<U01DF>;/
1120     <U01E1>;<U01E3>;<U01E5>;<U01E7>;<U01E9>;<U01EB>;<U01ED>;<U01EF>;/
1121     <U01F1>;<U01F2>;<U01F3>;<U01F5>;<U01FB>;<U01FD>;<U01FF>;/
1122     <U0201>;<U0203>;<U0205>;<U0207>;<U0209>;<U020B>;<U020D>;<U020F>;/
1123     <U0211>;<U0213>;<U0215>;<U0217>;/
1124 % COLLECTION 5 IPA EXTENSIONS/
1125     <U0250>..<U0293>;<U0299>..<U02A0>;<U02A3>..<U02A8>;/
1126 % COLLECTION 8 BASIC GREEK/
1127     <U0390>;<U03AC>..<U03CE>;/
1128 % COLLECTION 9 GREEK SYMBOLS AND COPTIC/
1129     <U03E2>;<U03E4>;<U03E6>;<U03E8>;<U03EA>;<U03EC>;<U03EE>;/
1130 % COLLECTION 10 CYRILLIC/
1131     <U0430>..<U044F>;<U0451>..<U045C>;<U045E>;<U045F>;/
1132     <U0461>;<U0463>;<U0465>;<U0467>;<U0469>;<U046B>;<U046D>;<U046F>;/
1133     <U0471>;<U0473>;<U0475>;<U0477>;<U0479>;<U047B>;<U047D>;<U047F>;/
1134     <U0481>;/
1135     <U0491>;<U0493>;<U0495>;<U0497>;<U0499>;<U049B>;<U049D>;<U049F>;/
1136     <U04A1>;<U04A3>;<U04A5>;<U04A7>;<U04A9>;<U04AB>;<U04AD>;<U04AF>;/
1137     <U04B1>;<U04B3>;<U04B5>;<U04B7>;<U04B9>;<U04BB>;<U04BD>;<U04BF>;/
1138     <U04C2>;<U04C4>;<U04C8>;<U04CC>;/
1139     <U04D1>;<U04D3>;<U04D5>;<U04D7>;<U04D9>;<U04DB>;<U04DD>;<U04DF>;/
1140     <U04E1>;<U04E3>;<U04E5>;<U04E7>;<U04E9>;<U04EB>;/
1141     <U04EF>;<U04F1>;<U04F3>;<U04F5>;/
1142     <U04F9>;/
1143 % COLLECTION 11 ARMENIAN/
1144     <U0561>..<U0587>;/
1145 % COLLECTION 28 GEORGIAN EXTENDED/
1146     <U10D0>..<U10F6>;/
1147 % COLLECTION 30 LATIN EXTENDED ADDITIONAL/
1148     <U1E01>;<U1E03>;<U1E05>;<U1E07>;<U1E09>;<U1E0B>;<U1E0D>;<U1E0F>;/
1149     <U1E11>;<U1E13>;<U1E15>;<U1E17>;<U1E19>;<U1E1B>;<U1E1D>;<U1E1F>;/
1150     <U1E21>;<U1E23>;<U1E25>;<U1E27>;<U1E29>;<U1E2B>;<U1E2D>;<U1E2F>;/
1151     <U1E31>;<U1E33>;<U1E35>;<U1E37>;<U1E39>;<U1E3B>;<U1E3D>;<U1E3F>;/
1152     <U1E41>;<U1E43>;<U1E45>;<U1E47>;<U1E49>;<U1E4B>;<U1E4D>;<U1E4F>;/
1153     <U1E51>;<U1E53>;<U1E55>;<U1E57>;<U1E59>;<U1E5B>;<U1E5D>;<U1E5F>;/
1154     <U1E61>;<U1E63>;<U1E65>;<U1E67>;<U1E69>;<U1E6B>;<U1E6D>;<U1E6F>;/
1155     <U1E71>;<U1E73>;<U1E75>;<U1E77>;<U1E79>;<U1E7B>;<U1E7D>;<U1E7F>;/
1156     <U1E81>;<U1E83>;<U1E85>;<U1E87>;<U1E89>;<U1E8B>;<U1E8D>;<U1E8F>;/
1157     <U1E91>;<U1E93>;<U1E95>;/
1158     <U1EA1>;<U1EA3>;<U1EA5>;<U1EA7>;<U1EA9>;<U1EAB>;<U1EAD>;<U1EAF>;/
1159     <U1EB1>;<U1EB3>;<U1EB5>;<U1EB7>;<U1EB9>;<U1EBB>;<U1EBD>;<U1EBF>;/
1160     <U1EC1>;<U1EC3>;<U1EC5>;<U1EC7>;<U1EC9>;<U1ECB>;<U1ECD>;<U1ECF>;/
1161     <U1ED1>;<U1ED3>;<U1ED5>;<U1ED7>;<U1ED9>;<U1EDB>;<U1EDD>;<U1EDF>;/
1162     <U1EE1>;<U1EE3>;<U1EE5>;<U1EE7>;<U1EE9>;<U1EEB>;<U1EED>;<U1EEF>;/
1163     <U1EF1>;<U1EF3>;<U1EF5>;<U1EF7>;<U1EF9>;/

```

```

1163 % COLLECTION 31 GREEK EXTENDED/
1164 <U1F08>..<U1F0F>;<U1F18>..<U1F1D>;<U1F28>..<U1F2F>;<U1F38>..<U1F3F>;/
1165 <U1F48>..<U1F4D>;<U1F59>;<U1F5B>;<U1F5D>;<U1F5F>;<U1F68>..<U1F6F>;/
1166 <U1F00>..<U1F07>;<U1F10>..<U1F15>;<U1F20>..<U1F27>;<U1F30>..<U1F37>;/
1167 <U1F40>..<U1F45>;<U1F50>..<U1F57>;<U1F60>..<U1F67>;<U1F70>..<U1F7D>;/
1168 <U1F80>..<U1F87>;<U1F90>..<U1F97>;<U1FA0>..<U1FA7>;<U1FB0>..<U1FB4>;/
1169 <U1FB6>;<U1FB7>;<U1FC2>..<U1FC4>;<U1FC6>;<U1FC7>;<U1FD0>..<U1FD3>;/
1170 <U1FD6>;<U1FD7>;<U1FE0>..<U1FE7>;<U1FF2>..<U1FF4>;<U1FF6>;<U1FF7>;/
1171 % COLLECTION 33 SUPERSCRIPTS AND SUBSCRIPTS/
1172 <U207F>
1173 %
1174 % The "alpha" class of the "i18n" FDCC-set is reflecting
1175 % the recommendations in TR 10176 annex A
1176 alpha /
1177 % COLLECTION 1 BASIC LATIN/
1178 <U0041>..<U005A>;<U0061>..<U007A>;/
1179 % COLLECTION 2 LATIN-1 SUPPLEMENT/
1180 <U00AA>;<U00BA>;<U00C0>..<U00D6>;<U00D8>..<U00F6>;<U00F8>..<U00FF>;/
1181 % COLLECTION 3 LATIN EXTENDED-A/
1182 <U0100>..<U017F>;/
1183 % COLLECTION 4 LATIN EXTENDED-B/
1184 <U0180>..<U01F5>;<U01FA>..<U0217>;/
1185 % COLLECTION 5 IPA EXTENSIONS/
1186 <U0250>..<U02A8>;/
1187 % COLLECTION 30 LATIN EXTENDED ADDITIONAL/
1188 <U1E00>..<U1E9B>;<U1EA0>..<U1EF9>;/
1189 % COLLECTION 33 SUPERSCRIPTS AND SUBSCRIPTS/
1190 <U207F>;/
1191 % COLLECTION 8 BASIC GREEK/
1192 <U0386>;<U0388>..<U038A>;<U038C>;<U038E>..<U03A1>;<U03A3>..<U03CE>;/
1193 % COLLECTION 9 GREEK SYMBOLS AND COPTIC/
1194 <U03D0>..<U03D6>;<U03DA>;<U03DC>;<U03DE>;<U03E0>;<U03E2>..<U03F3>;/
1195 % COLLECTION 31 GREEK EXTENDED/
1196 <U1F00>..<U1F15>;<U1F18>..<U1F1D>;<U1F20>..<U1F45>;<U1F48>..<U1F4D>;/
1197 <U1F50>..<U1F57>;<U1F59>;<U1F5B>;<U1F5D>;<U1F5F>..<U1F7D>;/
1198 <U1F80>..<U1FB4>;<U1FB6>..<U1FBC>;<U1FC2>..<U1FC4>;<U1FC6>..<U1FCC>;/
1199 <U1FD0>..<U1FD3>;<U1FD6>..<U1FDB>;<U1FE0>..<U1FEC>;<U1FF2>..<U1FF4>;/
1200 <U1FF6>..<U1FFC>;/
1201 % COLLECTION 10 CYRILLIC/
1202 <U0401>..<U040C>;<U040E>..<U044F>;<U0451>..<U045C>;<U045E>..<U0481>;/
1203 <U0490>..<U04C4>;<U04C7>..<U04C8>;<U04CB>..<U04CC>;<U04D0>..<U04EB>;/
1204 <U04EE>..<U04F5>;<U04F8>..<U04F9>;/
1205 % COLLECTION 11 ARMENIAN/
1206 <U0531>..<U0556>;<U0561>..<U0587>;/
1207 % COLLECTION 13 HEBREW EXTENDED/
1208 <U05B0>..<U05B9>;<U05BB>..<U05BD>;<U05BF>;<U05C1>..<U05C2>;/
1209 <U05D0>..<U05EA>;<U05F0>..<U05F2>;/
1210 % COLLECTION 15 ARABIC EXTENDED/
1211 <U0621>..<U063A>;<U0641>..<U064A>;<U0670>..<U06B7>;<U06BA>..<U06BE>;/
1212 <U06C0>..<U06CE>;<U06D0>..<U06D3>;<U06D5>..<U06DC>;<U06E5>..<U06E8>;/
1213 % COLLECTION 16 DEVANAGARI/
1214 <U0901>..<U0903>;<U0905>..<U0939>;<U093E>..<U094D>;<U0950>..<U0952>;/
1215 <U0958>..<U0963>;/
1216 % COLLECTION 17 BENGALI/
1217 <U0981>..<U0983>;<U0985>..<U098C>;<U098F>..<U0990>;/
1218 <U0993>..<U09A8>;<U09AA>..<U09B0>;<U09B2>;<U09B6>..<U09B9>;/
1219 <U09BE>..<U09C4>;<U09C7>..<U09C8>;<U09CB>..<U09CD>;<U09DC>..<U09DD>;/
1220 <U09DF>..<U09E3>;<U09F0>..<U09F1>;/
1221 % COLLECTION 18 GURMUKHI/
1222 <U0A02>;<U0A05>..<U0A0A>;<U0A0F>..<U0A10>;<U0A13>..<U0A28>;/
1223 <U0A2A>..<U0A30>;<U0A32>..<U0A33>;<U0A35>..<U0A36>;<U0A38>..<U0A39>;/
1224 <U0A3E>..<U0A42>;<U0A47>..<U0A48>;<U0A4B>..<U0A4D>;<U0A59>..<U0A5C>;/
1225 <U0A5E>;<U0A74>;/
1226 % COLLECTION 19 GUJARATI/
1227 <U0A81>..<U0A83>;<U0A85>..<U0A8B>;<U0A8D>;<U0A8F>..<U0A91>;/
1228 <U0A93>..<U0AA8>;<U0AAA>..<U0AB0>;<U0AB2>..<U0AB3>;<U0AB5>..<U0AB9>;/
1229 <U0ABD>..<U0AC5>;<U0AC7>..<U0AC9>;<U0ACB>..<U0ACD>;<U0AD0>;<U0AE0>;/
1230 % COLLECTION 20 ORIYA/
1231 <U0B01>..<U0B03>;<U0B05>..<U0B0C>;<U0B0F>..<U0B10>;<U0B13>..<U0B28>;/
1232 <U0B2A>..<U0B30>;<U0B32>..<U0B33>;<U0B36>..<U0B39>;<U0B3E>..<U0B43>;/
1233 <U0B47>..<U0B48>;<U0B4B>..<U0B4D>;<U0B5C>..<U0B5D>;<U0B5F>..<U0B61>;/
1234 % COLLECTION 21 TAMIL/
1235 <U0B82>..<U0B83>;<U0B85>..<U0B8A>;<U0B8E>..<U0B90>;<U0B92>..<U0B95>;/
1236 <U0B99>..<U0B9A>;<U0B9C>;<U0B9E>..<U0B9F>;<U0BA3>..<U0BA4>;/
1237 <U0BA8>..<U0BAA>;<U0BAE>..<U0BB5>;<U0BB7>..<U0BB9>;<U0BBE>..<U0BC2>;/
1238 <U0BC6>..<U0BC8>;<U0BCA>..<U0BCD>;/
1239 % COLLECTION 22 TELUGU/
1240 <U0C01>..<U0C03>;<U0C05>..<U0C0C>;<U0C0E>..<U0C10>;<U0C12>..<U0C28>;/

```



```

1241     <U0C2A>..<U0C33>;<U0C35>..<U0C39>;<U0C3E>..<U0C44>;<U0C46>..<U0C48>;/
1242     <U0C4A>..<U0C4D>;<U0C60>..<U0C61>;/
1243 % COLLECTION 23 KANNADA/
1244     <U0C82>..<U0C83>;<U0C85>..<U0C8C>;<U0C8E>..<U0C90>;<U0C92>..<U0CA8>;/
1245     <U0CAA>..<U0CB3>;<U0CB5>..<U0CB9>;<U0CBE>..<U0CC4>;<U0CC6>..<U0CC8>;/
1246     <U0CCA>..<U0CCD>;<U0CDE>;<U0CE0>..<U0CE1>;/
1247 % COLLECTION 24 MALAYALAM/
1248     <U0D02>..<U0D03>;<U0D05>..<U0D0C>;<U0D0E>..<U0D10>;<U0D12>..<U0D28>;/
1249     <U0D2A>..<U0D39>;<U0D3E>..<U0D43>;<U0D46>..<U0D48>;<U0D4A>..<U0D4D>;/
1250     <U0D60>..<U0D61>;/
1251 % COLLECTION 25 THAI/
1252     <U0E01>..<U0E3A>;<U0E40>..<U0E4E>;/
1253 % COLLECTION 26 LAO/
1254     <U0E81>..<U0E82>;<U0E84>;<U0E87>..<U0E88>;<U0E8A>;<U0E8D>;/
1255     <U0E94>..<U0E97>;<U0E99>..<U0E9F>;<U0EA1>..<U0EA3>;<U0EA5>;<U0EA7>;/
1256     <U0EAA>..<U0EAB>;<U0EAD>..<U0EAE>;<U0EB0>..<U0EB9>;<U0EBB>..<U0EBD>;/
1257     <U0EC0>..<U0EC4>;<U0EC6>;<U0EC8>..<U0ECD>;<U0EDC>..<U0EDD>;/
1258 % TIBETAN Amendment 6/
1259     <U0F00>;<U0F18>..<U0F19>;<U0F35>;<U0F37>;<U0F39>;<U0F40>..<U0F47>;/
1260     <U0F49>..<U0F69>;/
1261     <U0F71>..<U0F84>;<U0F86>..<U0F8B>;<U0F90>..<U0F95>;<U0F97>;/
1262     <U0F99>..<U0FAD>;<U0FB1>..<U0FB7>;<U0FB9>;/
1263 % COLLECTION 28 GEORGIAN EXTENDED/
1264     <U10A0>..<U10C5>;<U10D0>..<U10F6>;/
1265 % COLLECTION 50 HIRAGANA/
1266     <U3041>..<U3093>;<U309B>..<U309C>;/
1267 % COLLECTION 51 KATAKANA/
1268     <U30A1>..<U30F6>;<U30FB>..<U30FC>;/
1269 % COLLECTION 52 BOPOMOFO/
1270     <U3105>..<U312C>;/
1271 % CJK unified ideographs/
1272     <U4E00>..<U9FA5>;/
1273 % HANGUL amendment 5/
1274     <UAC00>..<UD7A3>;/
1275 % Miscellaneous/
1276     <U00B5>;<U02B0>..<U02B8>;<U02BB>;<U02BD>..<U02C1>;/
1277     <U02D0>..<U02D1>;<U02E0>..<U02E4>;<U037A>;<U0559>;<U093D>;<U0B3D>;/
1278     <U1FBE>;<U2160>..<U2182>;<U3021>..<U3029>
1279 %
1280 % The "digit" class of the "i18n" FDCC-set is reflecting
1281 % the recommendations in TR 10176 annex A
1282 digit /
1283 % COLLECTION 1 BASIC LATIN/
1284     <U0030>..<U0039>;/
1285 % COLLECTION 15 ARABIC EXTENDED/
1286     <U0660>..<U0669>;<U06F0>..<U06F9>;/
1287 % COLLECTION 16 DEVANAGARI/
1288     <U0966>..<U096F>;/
1289 % COLLECTION 18 BENGALI/
1290     <U09E6>..<U09EF>;/
1291 % COLLECTION 18 GURMUKHI/
1292     <U0A66>..<U0A6F>;/
1293 % COLLECTION 19 GUJARATI/
1294     <U0AE6>..<U0AEF>;/
1295 % COLLECTION 20 ORIYA/
1296     <U0B66>..<U0B6F>;/
1297 % COLLECTION 21 TAMIL/
1298     <0>;<U0BE7>..<U0BEF>;/
1299 % COLLECTION 22 TELUGU/
1300     <U0C66>..<U0C6F>;/
1301 % COLLECTION 23 KANNADA/
1302     <U0CE6>..<U0CEF>;/
1303 % COLLECTION 24 MALAYALAM/
1304     <U0D66>..<U0D6F>;/
1305 % COLLECTION 25 THAI/
1306     <U0E50>..<U0E59>;/
1307 % COLLECTION 26 LAO/
1308     <U0ED0>..<U0ED9>;/
1309 % TIBETAN Amendment 6/
1310     <U0F20>..<U0F29>;/
1311 % FULLWIDTH /
1312     <UFF10>..<UFF19>
1313 %
1314 outdigit <U0030>..<U0039>
1315 %
1316 space /
1317 % ISO/IEC 6429/
1318     <U0008>;<U000A>..<U000D>;/

```

```

1319 % COLLECTION 1 BASIC LATIN/
1320 <U0020>;/
1321 % COLLECTION 35 GENERAL PUNCTUATION/
1322 <U2000>..<U2006>;<U2008>..<U200B>;/
1323 % COLLECTION 50 CJK SYMBOLS AND PUNCTUATION, HIRAGANA/
1324 <U3000>
1325 %
1326 cntrl <U0000>..<U001F>;<U007F>..<U009F>
1327 %
1328 punct /
1329 <U0021>..<U002F>;<U003A>..<U0040>;<U005B>..<U0060>;<U007B>..<U007E>;/
1330 <U00A0>..<U00A9>;<U00AB>..<U00B4>;<U00B6>..<U00B9>;<U00BB>..<U00BF>;/
1331 <U00D7>;<U00F7>;/
1332 <U037E>;<U0482>;<U055A>..<U055F>;<U0589>;<U05BE>;<U05C0>;<U05C3>;/
1333 <U05F3>;<U05F4>;<U060C>;<U061B>;<U061F>;<U0640>;<U064B>..<U0652>;/
1334 <U066A>..<U066D>;<U06D4>;<U06DD>..<U06E1>;<U06E9>..<U06EC>;<U10FB>;/
1335 <U2010>..<U2029>;<U2030>..<U2046>;<U20A0>..<U20AC>;<U2100>..<U210B>;/
1336 <U210D>..<U2110>;<U2112>..<U211B>;<U211D>..<U2127>;<U212A>..<U212C>;/
1337 <U212E>..<U2138>;<U2200>..<U22F1>;<U2300>;<U2302>..<U237A>;<U2400>..<U2424>;/
1338 <U2440>..<U244A>;<U2580>..<U2595>;<U25A0>..<U25EF>;<U2600>..<U2613>;/
1339 <U261A>..<U266F>;<U2701>..<U2704>;<U2706>..<U2709>;<U270C>..<U2727>;/
1340 <U2729>..<U274B>;<U274D>;<U274F>..<U2752>;<U2756>;<U2758>..<U275E>;/
1341 <U2761>..<U2767>;<U3000>..<U3020>;<U3030>;<U3036>;<U3037>;<U303F>;<U3164>;/
1342 <U3190>..<U319F>;<U3200>..<U321C>;<U3220>..<U3243>;<U3260>..<U327B>;/
1343 <U327F>..<U32B0>;<U32C0>..<U32CB>;<U32D0>..<U32FE>;<U3300>..<U3376>;/
1344 <U337B>..<U33DD>;<U33E0>..<U33FE>;<UFD3E>;<UFD3F>;<UFE49>..<UFE52>;/
1345 <UFE54>..<UFE66>;<UFE68>;<UFE6B>;<UFEFF>;<UFF01>..<UFF0F>;<UFF1A>..<UFF20>;/
1346 <UFF3B>..<UFF40>;<UFF5B>..<UFF5E>;<UFF61>..<UFF65>;<UFF70>;<UFF9E>..<UFFA0>;/
1347 <UFFE0>..<UFFE6>;<UFFE8>..<UFFEE>;<UFFFC>;<UFFFD>
1348 %
1349 graph /
1350 <U0021>..<U007E>;<U00A0>..<U01F5>;<U01FA>..<U0217>;/
1351 <U0250>..<U02A8>;<U02B0>..<U02DE>;<U02E0>..<U02E9>;<U0300>..<U0345>;/
1352 <U0360>;<U0361>;<U0374>;<U0375>;<U037A>;<U037E>;<U0384>..<U038A>;<U038C>;/
1353 <U038E>..<U03A1>;<U03A3>..<U03CE>;<U03D0>..<U03D6>;<U03DA>;<U03DC>;<U03DE>;/
1354 <U03E0>;<U03E2>..<U03F3>;<U0401>..<U040C>;<U040E>..<U044F>;/
1355 <U0451>..<U045C>;<U045E>..<U0486>;<U0490>..<U04C4>;<U04C7>;<U04C8>;/
1356 <U04CB>;<U04CC>;<U04D0>..<U04EB>;<U04EE>..<U04F5>;<U04F8>;<U04F9>;/
1357 <U0531>..<U0556>;<U0559>..<U055F>;<U0561>..<U0587>;<U0589>;/
1358 <U0591>..<U05A1>;<U05A3>..<U05AF>;<U05B0>..<U05B9>;/
1359 <U05BB>..<U05C4>;<U05D0>..<U05EA>;<U05F0>..<U05F4>;<U060C>;<U061B>;<U061F>;/
1360 <U0621>..<U063A>;<U0640>..<U0652>;<U0660>..<U066D>;<U0670>..<U06B7>;/
1361 <U06BA>..<U06BE>;<U06C0>..<U06CE>;<U06D0>..<U06ED>;<U06F0>..<U06F9>;/
1362 <U0901>..<U0903>;<U0905>..<U0939>;<U093C>..<U094D>;<U0950>..<U0954>;/
1363 <U0958>..<U0970>;<U0981>..<U0983>;<U0985>..<U098C>;<U098F>;<U0990>;/
1364 <U0993>..<U09A8>;<U09AA>..<U09B0>;<U09B2>;<U09B6>..<U09B9>;<U09BC>;/
1365 <U09BE>;<U09C4>;<U09C7>;<U09C8>;<U09CB>..<U09CD>;<U09D7>;<U09DC>;<U09DD>;/
1366 <U09DF>..<U09E3>;<U09E6>..<U09FA>;<U0A02>;<U0A05>..<U0A0A>;<U0A0F>;<U0A10>;/
1367 <U0A13>..<U0A28>;<U0A2A>..<U0A30>;<U0A32>;<U0A33>;<U0A35>;<U0A36>;/
1368 <U0A38>;<U0A39>;<U0A3C>;<U0A3E>..<U0A42>;<U0A47>;<U0A48>;<U0A4B>..<U0A4D>;/
1369 <U0A59>..<U0A5C>;<U0A5E>;<U0A66>..<U0A74>;<U0A81>..<U0A83>;<U0A85>..<U0A8B>;/
1370 <U0A8D>;<U0A8F>..<U0A91>;<U0A93>..<U0AA8>;<U0AAA>..<U0AB0>;/
1371 <U0AB2>;<U0AB3>;<U0AB5>..<U0AB9>;<U0ABC>..<U0AC5>;<U0AC7>..<U0AC9>;/
1372 <U0ACB>..<U0ACD>;<U0AD0>;<U0AE0>;<U0AE6>..<U0AEF>;<U0B01>..<U0B03>;/
1373 <U0B05>..<U0B0C>;<U0B0F>;<U0B10>;<U0B13>..<U0B28>;<U0B2A>..<U0B30>;/
1374 <U0B32>;<U0B33>;<U0B36>..<U0B39>;<U0B3C>..<U0B43>;<U0B47>;<U0B48>;/
1375 <U0B4B>..<U0B4D>;<U0B56>;<U0B57>;<U0B5C>;<U0B5D>;<U0B5F>..<U0B61>;/
1376 <U0B66>..<U0B70>;<U0B82>;<U0B83>;<U0B85>..<U0B8A>;<U0B8E>..<U0B90>;/
1377 <U0B92>;<U0B95>;<U0B99>;<U0B9A>;<U0B9C>;<U0B9E>;<U0B9F>;<U0BA3>;<U0BA4>;/
1378 <U0BA8>..<U0BAA>;<U0BAE>..<U0BB5>;<U0BB7>..<U0BB9>;<U0BBE>..<U0BC2>;/
1379 <U0BC6>..<U0BC8>;<U0BCA>..<U0BCD>;<U0BD7>;<U0BE7>..<U0BF2>;<U0C01>..<U0C03>;/
1380 <U0C05>..<U0C0C>;<U0C0E>..<U0C10>;<U0C12>..<U0C28>;<U0C2A>..<U0C33>;/
1381 <U0C35>..<U0C39>;<U0C3E>..<U0C44>;<U0C46>..<U0C48>;<U0C4A>..<U0C4D>;/
1382 <U0C55>;<U0C56>;<U0C60>;<U0C61>;<U0C66>..<U0C6F>;<U0C82>;<U0C83>;/
1383 <U0C85>..<U0C8C>;<U0C8E>..<U0C90>;<U0C92>..<U0CA8>;<U0CAA>..<U0CB3>;/
1384 <U0CB5>..<U0CB9>;<U0CBE>..<U0CC4>;<U0CC6>..<U0CC8>;<U0CCA>..<U0CCD>;/
1385 <U0CD5>;<U0CD6>;<U0CDE>;<U0CE0>;<U0CE1>;<U0CE6>..<U0CEF>;<U0D02>;<U0D03>;/
1386 <U0D05>..<U0D0C>;<U0D0E>..<U0D10>;<U0D12>..<U0D28>;<U0D2A>..<U0D39>;/
1387 <U0D3E>..<U0D43>;<U0D46>..<U0D48>;<U0D4A>..<U0D4D>;<U0D57>;<U0D60>;<U0D61>;/
1388 <U0D66>;<U0D6F>;<U0E01>..<U0E3A>;<U0E3F>..<U0E5B>;<U0E81>;<U0E82>;<U0E84>;/
1389 <U0E87>;<U0E88>;<U0E8A>;<U0E8D>;<U0E94>..<U0E97>;<U0E99>;<U0E9F>;/
1390 <U0EA1>..<U0EA3>;<U0EA5>;<U0EA7>;<U0EAA>;<U0EAB>;<U0EAD>..<U0EB9>;/
1391 <U0EBB>..<U0EBD>;<U0EC0>..<U0EC4>;<U0EC6>;<U0EC8>..<U0ECD>;<U0ED0>..<U0ED9>;/
1392 <U0EDC>;<U0EDD>;/
1393 <U0F00>..<U0F47>;<U0F49>..<U0F69>;<U0F71>..<U0F7F>;/
1394 <U10A0>..<U10C5>;<U10D0>..<U10F6>;<U10FB>;<U1100>..<U1159>;/
1395 <U115F>..<U11A2>;<U11A8>..<U11F9>;<U1E00>..<U1E9B>;<U1EA0>..<U1EF9>;/
1396 <U1F00>..<U1F15>;<U1F18>..<U1F1D>;<U1F20>..<U1F45>;<U1F48>..<U1F4D>;/

```

```

1397 <U1F50>..<U1F57>;<U1F59>;<U1F5B>;<U1F5D>;<U1F5F>..<U1F7D>;<U1F80>..<U1FB4>;/
1398 <U1FB6>..<U1FC4>;<U1FC6>..<U1FD3>;<U1FD6>..<U1FDB>;<U1FDD>..<U1FEF>;/
1399 <U1FF2>..<U1FF4>;<U1FF6>..<U1FFE>;<U2000>..<U202E>;<U2030>..<U2046>;/
1400 <U206A>..<U2070>;<U2074>..<U208E>;<U20A0>..<U20AC>;<U20D0>..<U20E1>;/
1401 <U2100>..<U2138>;<U2153>..<U2182>;<U2190>..<U21EA>;<U2200>..<U22F1>;<U2300>;/
1402 <U2302>..<U237A>;<U2400>..<U2424>;<U2440>..<U244A>;<U2460>..<U24EA>;/
1403 <U2500>..<U2595>;<U25A0>..<U25EF>;<U2600>..<U2613>;<U261A>..<U266F>;/
1404 <U2701>..<U2704>;<U2706>..<U2709>;<U270C>..<U2727>;<U2729>..<U274B>;<U274D>;/
1405 <U274F>..<U2752>;<U2756>;<U2758>..<U275E>;<U2761>..<U2767>;<U2776>..<U2794>;/
1406 <U2798>..<U27AF>;<U27B1>..<U27BE>;<U3000>..<U3037>;<U303F>;<U3041>..<U3094>;/
1407 <U3099>..<U309E>;<U30A1>..<U30FE>;<U3105>..<U312C>;<U3131>..<U318E>;/
1408 <U3190>..<U319F>;<U3200>..<U321C>;<U3220>..<U3243>;<U3260>..<U327B>;/
1409 <U327F>..<U32B0>;<U32C0>..<U32CB>;<U32D0>..<U32FE>;<U3300>..<U3376>;/
1410 <U337B>..<U33DD>;<U33E0>..<U33FE>;/
1411 <UAC00>..<UD7A3>;<U4E00>..<U9FA5>;/
1412 <UFB00>..<UFB06>;<UFB13>..<UFB17>;/
1413 <UFB1E>..<UFB36>;<UFB38>..<UFB3C>;<UFB3E>;<UFB40>;<UFB41>;<UFB43>;<UFB44>;/
1414 <UFB46>..<UFBB1>;<UFBDB3>..<UFD3F>;<UFD50>..<UFD8F>;<UFD92>..<UFDCT>;/
1415 <UFDFO>..<UFDFF>;<UFE20>..<UFE23>;<UFE30>..<UFE44>;<UFE49>..<UFE52>;/
1416 <UFE54>..<UFE66>;<UFE68>..<UFE6B>;<UFE70>..<UFE72>;<UFE74>;<UFE76>..<UFEFC>;/
1417 <UFEFF>;<UFF01>..<UFF5E>;<UFF61>..<UFFBE>;<UFFC2>..<UFFC7>;/
1418 <UFFCA>..<UFFCF>;<UFFD2>..<UFFD7>;<UFFDA>..<UFFDC>;<UFFE0>..<UFFE6>;/
1419 <UFFE8>..<UFFEE>;<UFFFC>;<UFFFD>
1420 %
1421 % "print" is by default "graph", and the <space> character
1422 xdigit <U0030>..<U0039>;<U0041>..<U0046>;<U0061>..<U0066>
1423 blank <U0008>;<U0020>;<U2000>..<U2006>;<U2008>..<U200B>;<U3000>
1424 %
1425 toupper /
1426 (<U0061>,<U0041>);(<U0062>,<U0042>);(<U0063>,<U0043>);(<U0064>,<U0044>);/
1427 (<U0065>,<U0045>);(<U0066>,<U0046>);(<U0067>,<U0047>);(<U0068>,<U0048>);/
1428 (<U0069>,<U0049>);(<U006A>,<U004A>);(<U006B>,<U004B>);(<U006C>,<U004C>);/
1429 (<U006D>,<U004D>);(<U006E>,<U004E>);(<U006F>,<U004F>);(<U0070>,<U0050>);/
1430 (<U0071>,<U0051>);(<U0072>,<U0052>);(<U0073>,<U0053>);(<U0074>,<U0054>);/
1431 (<U0075>,<U0055>);(<U0076>,<U0056>);(<U0077>,<U0057>);(<U0078>,<U0058>);/
1432 (<U0079>,<U0059>);(<U007A>,<U005A>);(<U00E0>,<U00C0>);(<U00E1>,<U00C1>);/
1433 (<U00E2>,<U00C2>);(<U00E3>,<U00C3>);(<U00E4>,<U00C4>);(<U00E5>,<U00C5>);/
1434 (<U00E6>,<U00C6>);(<U00E7>,<U00C7>);(<U00E8>,<U00C8>);(<U00E9>,<U00C9>);/
1435 (<U00EA>,<U00CA>);(<U00EB>,<U00CB>);(<U00EC>,<U00CC>);(<U00ED>,<U00CD>);/
1436 (<U00EE>,<U00CE>);(<U00EF>,<U00CF>);(<U00F0>,<U00D0>);(<U00F1>,<U00D1>);/
1437 (<U00F2>,<U00D2>);(<U00F3>,<U00D3>);(<U00F4>,<U00D4>);(<U00F5>,<U00D5>);/
1438 (<U00F6>,<U00D6>);(<U00F7>,<U00D7>);(<U00F8>,<U00D8>);(<U00F9>,<U00D9>);/
1439 (<U00FA>,<U00DA>);(<U00FB>,<U00DB>);(<U00FC>,<U00DC>);(<U00FD>,<U00DD>);(<U00FE>,<U00DE>);/
1440 (<U00FF>,<U0178>);(<U0101>,<U0100>);(<U0103>,<U0102>);(<U0105>,<U0104>);/
1441 (<U0107>,<U0106>);(<U0109>,<U0108>);(<U010B>,<U010A>);(<U010D>,<U010C>);/
1442 (<U010F>,<U010E>);(<U0111>,<U0110>);(<U0113>,<U0112>);(<U0115>,<U0114>);/
1443 (<U0117>,<U0116>);(<U0119>,<U0118>);(<U011B>,<U011A>);(<U011D>,<U011C>);/
1444 (<U011F>,<U011E>);(<U0121>,<U0120>);(<U0123>,<U0122>);(<U0125>,<U0124>);/
1445 (<U0127>,<U0126>);(<U0129>,<U0128>);(<U012B>,<U012A>);(<U012D>,<U012C>);/
1446 (<U012F>,<U012E>);(<U0131>,<U0049>);/
1447 (<U0133>,<U0132>);(<U0135>,<U0134>);(<U0137>,<U0136>);/
1448 (<U013A>,<U0139>);(<U013C>,<U013B>);(<U013E>,<U013D>);(<U0140>,<U013F>);/
1449 (<U0142>,<U0141>);(<U0144>,<U0143>);(<U0146>,<U0145>);(<U0148>,<U0147>);/
1450 (<U014B>,<U014A>);(<U014D>,<U014C>);(<U014F>,<U014E>);(<U0151>,<U0150>);/
1451 (<U0153>,<U0152>);(<U0155>,<U0154>);(<U0157>,<U0156>);(<U0159>,<U0158>);/
1452 (<U015B>,<U015A>);(<U015D>,<U015C>);(<U015F>,<U015E>);(<U0161>,<U0160>);/
1453 (<U0163>,<U0162>);(<U0165>,<U0164>);(<U0167>,<U0166>);(<U0169>,<U0168>);/
1454 (<U016B>,<U016A>);(<U016D>,<U016C>);(<U016F>,<U016E>);(<U0171>,<U0170>);/
1455 (<U0173>,<U0172>);(<U0175>,<U0174>);(<U0177>,<U0176>);(<U017A>,<U0179>);/
1456 (<U017C>,<U017B>);(<U017E>,<U017D>);(<U017F>,<U0053>);(<U0183>,<U0182>);/
1457 (<U0185>,<U0184>);(<U0188>,<U0187>);(<U018C>,<U018B>);(<U0192>,<U0191>);/
1458 (<U0199>,<U0198>);(<U01A1>,<U01A0>);(<U01A3>,<U01A2>);(<U01A5>,<U01A4>);/
1459 (<U01A8>,<U01A7>);(<U01AD>,<U01AC>);(<U01B0>,<U01AF>);(<U01B4>,<U01B3>);/
1460 (<U01B6>,<U01B5>);(<U01B9>,<U01B8>);(<U01BD>,<U01BC>);(<U01C5>,<U01C4>);/
1461 (<U01C6>,<U01C4>);(<U01C8>,<U01C7>);/
1462 (<U01C9>,<U01C7>);(<U01CB>,<U01CA>);(<U01CC>,<U01CA>);/
1463 (<U01CE>,<U01CD>);(<U01D0>,<U01CF>);(<U01D2>,<U01D1>);(<U01D4>,<U01D3>);/
1464 (<U01D6>,<U01D5>);(<U01D8>,<U01D7>);(<U01DA>,<U01D9>);(<U01DC>,<U01DB>);/
1465 (<U01DD>,<U018E>);(<U01DF>,<U01DE>);(<U01E1>,<U01E0>);(<U01E3>,<U01E2>);/
1466 (<U01E5>,<U01E4>);(<U01E7>,<U01E6>);(<U01E9>,<U01E8>);(<U01EB>,<U01EA>);/
1467 (<U01ED>,<U01EC>);(<U01EF>,<U01EE>);(<U01F2>,<U01F1>);/
1468 (<U01F3>,<U01F1>);(<U01F5>,<U01F4>);(<U01FB>,<U01FA>);(<U01FD>,<U01FC>);/
1469 (<U01FF>,<U01FE>);(<U0201>,<U0200>);(<U0203>,<U0202>);(<U0205>,<U0204>);/
1470 (<U0207>,<U0206>);(<U0209>,<U0208>);(<U020B>,<U020A>);(<U020D>,<U020C>);/
1471 (<U020F>,<U020E>);(<U0211>,<U0210>);(<U0213>,<U0212>);(<U0215>,<U0214>);/
1472 (<U0217>,<U0216>);(<U0253>,<U0181>);(<U0254>,<U0186>);(<U0256>,<U0189>);/
1473 (<U0257>,<U018A>);(<U0259>,<U018F>);(<U025B>,<U0190>);(<U0260>,<U0193>);/
1474 (<U0263>,<U0194>);(<U0268>,<U0197>);(<U0269>,<U0196>);(<U026F>,<U019C>);/

```


1553 (<U1ED1>, <U1ED0>); (<U1ED3>, <U1ED2>); (<U1ED5>, <U1ED4>); (<U1ED7>, <U1ED6>); /
 1554 (<U1ED9>, <U1ED8>); (<U1EDB>, <U1EDA>); (<U1EDD>, <U1EDC>); (<U1EDF>, <U1EDE>); /
 1555 (<U1EE1>, <U1EE0>); (<U1EE3>, <U1EE2>); (<U1EE5>, <U1EE4>); (<U1EE7>, <U1EE6>); /
 1556 (<U1EE9>, <U1EE8>); (<U1EEB>, <U1EEA>); (<U1EED>, <U1EEC>); (<U1EEF>, <U1EEE>); /
 1557 (<U1EF1>, <U1EF0>); (<U1EF3>, <U1EF2>); (<U1EF5>, <U1EF4>); (<U1EF7>, <U1EF6>); /
 1558 (<U1EF9>, <U1EF8>); (<U1F00>, <U1F08>); (<U1F01>, <U1F09>); (<U1F02>, <U1F0A>); /
 1559 (<U1F03>, <U1F0B>); (<U1F04>, <U1F0C>); (<U1F05>, <U1F0D>); (<U1F06>, <U1F0E>); /
 1560 (<U1F07>, <U1F0F>); (<U1F10>, <U1F18>); (<U1F11>, <U1F19>); (<U1F12>, <U1F1A>); /
 1561 (<U1F13>, <U1F1B>); (<U1F14>, <U1F1C>); (<U1F15>, <U1F1D>); (<U1F20>, <U1F28>); /
 1562 (<U1F21>, <U1F29>); (<U1F22>, <U1F2A>); (<U1F23>, <U1F2B>); (<U1F24>, <U1F2C>); /
 1563 (<U1F25>, <U1F2D>); (<U1F26>, <U1F2E>); (<U1F27>, <U1F2F>); (<U1F30>, <U1F38>); /
 1564 (<U1F31>, <U1F39>); (<U1F32>, <U1F3A>); (<U1F33>, <U1F3B>); (<U1F34>, <U1F3C>); /
 1565 (<U1F35>, <U1F3D>); (<U1F36>, <U1F3E>); (<U1F37>, <U1F3F>); (<U1F40>, <U1F48>); /
 1566 (<U1F41>, <U1F49>); (<U1F42>, <U1F4A>); (<U1F43>, <U1F4B>); (<U1F44>, <U1F4C>); /
 1567 (<U1F45>, <U1F4D>); (<U1F51>, <U1F59>); (<U1F53>, <U1F5B>); (<U1F55>, <U1F5D>); /
 1568 (<U1F57>, <U1F5F>); (<U1F61>, <U1F68>); (<U1F61>, <U1F69>); (<U1F62>, <U1F6A>); /
 1569 (<U1F63>, <U1F6B>); (<U1F64>, <U1F6C>); (<U1F65>, <U1F6D>); (<U1F66>, <U1F6E>); /
 1570 (<U1F67>, <U1F6F>); (<U1F70>, <U1FBA>); (<U1F71>, <U1FBB>); (<U1F72>, <U1FC8>); /
 1571 (<U1F73>, <U1FC9>); (<U1F74>, <U1FCA>); (<U1F75>, <U1FCB>); (<U1F76>, <U1FDA>); /
 1572 (<U1F77>, <U1FDB>); (<U1F78>, <U1FF8>); (<U1F79>, <U1FF9>); (<U1F7A>, <U1FEA>); /
 1573 (<U1F7B>, <U1FEB>); (<U1F7C>, <U1FFA>); (<U1F7D>, <U1FFB>); (<U1F80>, <U1F88>); /
 1574 (<U1F81>, <U1F89>); (<U1F82>, <U1F8A>); (<U1F83>, <U1F8B>); (<U1F84>, <U1F8C>); /
 1575 (<U1F85>, <U1F8D>); (<U1F86>, <U1F8E>); (<U1F87>, <U1F8F>); (<U1F90>, <U1F98>); /
 1576 (<U1F91>, <U1F99>); (<U1F92>, <U1F9A>); (<U1F93>, <U1F9B>); (<U1F94>, <U1F9C>); /
 1577 (<U1F95>, <U1F9D>); (<U1F96>, <U1F9E>); (<U1F97>, <U1F9F>); (<U1FA0>, <U1FA8>); /
 1578 (<U1FA1>, <U1FA9>); (<U1FA2>, <U1FAA>); (<U1FA3>, <U1FAB>); (<U1FA4>, <U1FAC>); /
 1579 (<U1FA5>, <U1FAD>); (<U1FA6>, <U1FAE>); (<U1FA7>, <U1FAF>); (<U1FB0>, <U1FB8>); /
 1580 (<U1FB1>, <U1FB9>); (<U1FB3>, <U1FBC>); (<U1FC3>, <U1FCC>); (<U1FD0>, <U1FD8>); /
 1581 (<U1FD1>, <U1FD9>); (<U1FE0>, <U1FE8>); (<U1FE1>, <U1FE9>); (<U1FE5>, <U1FEC>); /
 1582 (<U1FF3>, <U1FFC>); /
 1583 tolower /
 1584 (<U0041>, <U0061>); (<U0042>, <U0062>); (<U0043>, <U0063>); (<U0044>, <U0064>); /
 1585 (<U0045>, <U0065>); (<U0046>, <U0066>); (<U0047>, <U0067>); (<U0048>, <U0068>); /
 1586 (<U0049>, <U0069>); (<U004A>, <U006A>); (<U004B>, <U006B>); (<U004C>, <U006C>); /
 1587 (<U004D>, <U006D>); (<U004E>, <U006E>); (<U004F>, <U006F>); (<U0050>, <U0070>); /
 1588 (<U0051>, <U0071>); (<U0052>, <U0072>); (<U0053>, <U0073>); (<U0054>, <U0074>); /
 1589 (<U0055>, <U0075>); (<U0056>, <U0076>); (<U0057>, <U0077>); (<U0058>, <U0078>); /
 1590 (<U0059>, <U0079>); (<U005A>, <U007A>); (<U00C0>, <U00E0>); (<U00C1>, <U00E1>); /
 1591 (<U00C2>, <U00E2>); (<U00C3>, <U00E3>); (<U00C4>, <U00E4>); (<U00C5>, <U00E5>); /
 1592 (<U00C6>, <U00E6>); (<U00C7>, <U00E7>); (<U00C8>, <U00E8>); (<U00C9>, <U00E9>); /
 1593 (<U00CA>, <U00EA>); (<U00CB>, <U00EB>); (<U00CC>, <U00EC>); (<U00CD>, <U00ED>); /
 1594 (<U00CE>, <U00EE>); (<U00CF>, <U00EF>); (<U00D0>, <U00F0>); (<U00D1>, <U00F1>); /
 1595 (<U00D2>, <U00F2>); (<U00D3>, <U00F3>); (<U00D4>, <U00F4>); (<U00D5>, <U00F5>); /
 1596 (<U00D6>, <U00F6>); (<U00D8>, <U00F8>); (<U00D9>, <U00F9>); (<U00DA>, <U00FA>); /
 1597 (<U00DB>, <U00FB>); (<U00DC>, <U00FC>); (<U00DD>, <U00FD>); (<U00DE>, <U00FE>); /
 1598 (<U0178>, <U00FF>); (<U0100>, <U0101>); (<U0102>, <U0103>); (<U0104>, <U0105>); /
 1599 (<U0106>, <U0107>); (<U0108>, <U0109>); (<U010A>, <U010B>); (<U010C>, <U010D>); /
 1600 (<U010E>, <U010F>); (<U0110>, <U0111>); (<U0112>, <U0113>); (<U0114>, <U0115>); /
 1601 (<U0116>, <U0117>); (<U0118>, <U0119>); (<U011A>, <U011B>); (<U011C>, <U011D>); /
 1602 (<U011E>, <U011F>); (<U0120>, <U0121>); (<U0122>, <U0123>); (<U0124>, <U0125>); /
 1603 (<U0126>, <U0127>); (<U0128>, <U0129>); (<U012A>, <U012B>); (<U012C>, <U012D>); /
 1604 (<U012E>, <U012F>); (<U0130>, <U0069>); /
 1605 (<U0132>, <U0133>); (<U0134>, <U0135>); (<U0136>, <U0137>); /
 1606 (<U0139>, <U013A>); (<U013B>, <U013C>); (<U013D>, <U013E>); (<U013F>, <U0140>); /
 1607 (<U0141>, <U0142>); (<U0143>, <U0144>); (<U0145>, <U0146>); (<U0147>, <U0148>); /
 1608 (<U014A>, <U014B>); (<U014C>, <U014D>); (<U014E>, <U014F>); (<U0150>, <U0151>); /
 1609 (<U0152>, <U0153>); (<U0154>, <U0155>); (<U0156>, <U0157>); (<U0158>, <U0159>); /
 1610 (<U015A>, <U015B>); (<U015C>, <U015D>); (<U015E>, <U015F>); (<U0160>, <U0161>); /
 1611 (<U0162>, <U0163>); (<U0164>, <U0165>); (<U0166>, <U0167>); (<U0168>, <U0169>); /
 1612 (<U016A>, <U016B>); (<U016C>, <U016D>); (<U016E>, <U016F>); (<U0170>, <U0171>); /
 1613 (<U0172>, <U0173>); (<U0174>, <U0175>); (<U0176>, <U0177>); (<U0179>, <U017A>); /
 1614 (<U017B>, <U017C>); (<U017D>, <U017E>); (<U0182>, <U0183>); (<U0184>, <U0185>); /
 1615 (<U0187>, <U0188>); (<U0256>, <U0189>); (<U018B>, <U018C>); (<U018E>, <U01DD>); /
 1616 (<U0191>, <U0192>); (<U0198>, <U0199>); (<U019F>, <U0275>); /
 1617 (<U01A0>, <U01A1>); (<U01A2>, <U01A3>); /
 1618 (<U01A4>, <U01A5>); (<U01A6>, <U0280>); /
 1619 (<U01A7>, <U01A8>); (<U01AC>, <U01AD>); (<U01AF>, <U01B0>); /
 1620 (<U01B3>, <U01B4>); (<U01B5>, <U01B6>); (<U01B8>, <U01B9>); (<U01BC>, <U01BD>); /
 1621 (<U01C4>, <U01C6>); (<U01C5>, <U01C6>); (<U01C7>, <U01C9>); /
 1622 (<U01C8>, <U01C9>); (<U01CA>, <U01CC>); (<U01CB>, <U01CC>); /
 1623 (<U01CD>, <U01CE>); (<U01CF>, <U01D0>); (<U01D1>, <U01D2>); /
 1624 (<U01D3>, <U01D4>); (<U01D5>, <U01D6>); (<U01D7>, <U01D8>); (<U01D9>, <U01DA>); /
 1625 (<U01DB>, <U01DC>); (<U01DE>, <U01DF>); (<U01E0>, <U01E1>); (<U01E2>, <U01E3>); /
 1626 (<U01E4>, <U01E5>); (<U01E6>, <U01E7>); (<U01E8>, <U01E9>); (<U01EA>, <U01EB>); /
 1627 (<U01EC>, <U01ED>); (<U01EE>, <U01EF>); (<U01F1>, <U01F3>); /
 1628 (<U01F2>, <U01F3>); (<U01F4>, <U01F5>); (<U01FA>, <U01FB>); (<U01FC>, <U01FD>); /
 1629 (<U01FE>, <U01FF>); (<U0200>, <U0201>); (<U0202>, <U0203>); (<U0204>, <U0205>); /
 1630 (<U0206>, <U0207>); (<U0208>, <U0209>); (<U020A>, <U020B>); (<U020C>, <U020D>); /

1631 (<U020E>, <U020F>); (<U0210>, <U0211>); (<U0212>, <U0213>); (<U0214>, <U0215>); /
 1632 (<U0216>, <U0217>); (<U0181>, <U0253>); (<U0186>, <U0254>); (<U018A>, <U0257>); /
 1633 (<U018E>, <U0258>); (<U018F>, <U0259>); (<U0190>, <U025B>); (<U0193>, <U0260>); /
 1634 (<U0194>, <U0263>); (<U0197>, <U0268>); (<U0196>, <U0269>); (<U019C>, <U026F>); /
 1635 (<U019D>, <U0272>); (<U01A9>, <U0283>); (<U01AE>, <U0288>); (<U01B1>, <U028A>); /
 1636 (<U01B2>, <U028B>); (<U01B7>, <U0292>); (<U0386>, <U03AC>); (<U0388>, <U03AD>); /
 1637 (<U0389>, <U03AE>); (<U038A>, <U03AF>); (<U0391>, <U03B1>); (<U0392>, <U03B2>); /
 1638 (<U0393>, <U03B3>); (<U0394>, <U03B4>); (<U0395>, <U03B5>); (<U0396>, <U03B6>); /
 1639 (<U0397>, <U03B7>); (<U0398>, <U03B8>); (<U0399>, <U03B9>); (<U039A>, <U03BA>); /
 1640 (<U039B>, <U03BB>); (<U039C>, <U03BC>); (<U039D>, <U03BD>); (<U039E>, <U03BE>); /
 1641 (<U039F>, <U03BF>); (<U03A0>, <U03C0>); (<U03A1>, <U03C1>); (<U03A3>, <U03C3>); /
 1642 (<U03A4>, <U03C4>); (<U03A5>, <U03C5>); (<U03A6>, <U03C6>); (<U03A7>, <U03C7>); /
 1643 (<U03A8>, <U03C8>); (<U03A9>, <U03C9>); (<U03AA>, <U03CA>); (<U03AB>, <U03CB>); /
 1644 (<U038C>, <U03CC>); (<U038E>, <U03CD>); (<U038F>, <U03CE>); (<U03E2>, <U03E3>); /
 1645 (<U03E4>, <U03E5>); (<U03E6>, <U03E7>); (<U03E8>, <U03E9>); (<U03EA>, <U03EB>); /
 1646 (<U03EC>, <U03ED>); (<U03EE>, <U03EF>); (<U0410>, <U0430>); /
 1647 (<U0411>, <U0431>); (<U0412>, <U0432>); (<U0413>, <U0433>); (<U0414>, <U0434>); /
 1648 (<U0415>, <U0435>); (<U0416>, <U0436>); (<U0417>, <U0437>); (<U0418>, <U0438>); /
 1649 (<U0419>, <U0439>); (<U041A>, <U043A>); (<U041B>, <U043B>); (<U041C>, <U043C>); /
 1650 (<U041D>, <U043D>); (<U041E>, <U043E>); (<U041F>, <U043F>); (<U0420>, <U0440>); /
 1651 (<U0421>, <U0441>); (<U0422>, <U0442>); (<U0423>, <U0443>); (<U0424>, <U0444>); /
 1652 (<U0425>, <U0445>); (<U0426>, <U0446>); (<U0427>, <U0447>); (<U0428>, <U0448>); /
 1653 (<U0429>, <U0449>); (<U042A>, <U044A>); (<U042B>, <U044B>); (<U042C>, <U044C>); /
 1654 (<U042D>, <U044D>); (<U042E>, <U044E>); (<U042F>, <U044F>); (<U0401>, <U0451>); /
 1655 (<U0402>, <U0452>); (<U0403>, <U0453>); (<U0404>, <U0454>); (<U0405>, <U0455>); /
 1656 (<U0406>, <U0456>); (<U0407>, <U0457>); (<U0408>, <U0458>); (<U0409>, <U0459>); /
 1657 (<U040A>, <U045A>); (<U040B>, <U045B>); (<U040C>, <U045C>); (<U040E>, <U045E>); /
 1658 (<U040F>, <U045F>); (<U0460>, <U0461>); (<U0462>, <U0463>); (<U0464>, <U0465>); /
 1659 (<U0466>, <U0467>); (<U0468>, <U0469>); (<U046A>, <U046B>); (<U046C>, <U046D>); /
 1660 (<U046E>, <U046F>); (<U0470>, <U0471>); (<U0472>, <U0473>); (<U0474>, <U0475>); /
 1661 (<U0476>, <U0477>); (<U0478>, <U0479>); (<U047A>, <U047B>); (<U047C>, <U047D>); /
 1662 (<U047E>, <U047F>); (<U0480>, <U0481>); (<U0490>, <U0491>); (<U0492>, <U0493>); /
 1663 (<U0494>, <U0495>); (<U0496>, <U0497>); (<U0498>, <U0499>); (<U049A>, <U049B>); /
 1664 (<U049C>, <U049D>); (<U049E>, <U049F>); (<U04A0>, <U04A1>); (<U04A2>, <U04A3>); /
 1665 (<U04A4>, <U04A5>); (<U04A6>, <U04A7>); (<U04A8>, <U04A9>); (<U04AA>, <U04AB>); /
 1666 (<U04AC>, <U04AD>); (<U04AE>, <U04AF>); (<U04B0>, <U04B1>); (<U04B2>, <U04B3>); /
 1667 (<U04B4>, <U04B5>); (<U04B6>, <U04B7>); (<U04B8>, <U04B9>); (<U04BA>, <U04BB>); /
 1668 (<U04BC>, <U04BD>); (<U04BE>, <U04BF>); (<U04C1>, <U04C2>); (<U04C3>, <U04C4>); /
 1669 (<U04C7>, <U04C8>); (<U04CB>, <U04CC>); (<U04D0>, <U04D1>); (<U04D2>, <U04D3>); /
 1670 (<U04D4>, <U04D5>); (<U04D6>, <U04D7>); (<U04D8>, <U04D9>); (<U04DA>, <U04DB>); /
 1671 (<U04DC>, <U04DD>); (<U04DE>, <U04DF>); (<U04E0>, <U04E1>); (<U04E2>, <U04E3>); /
 1672 (<U04E4>, <U04E5>); (<U04E6>, <U04E7>); (<U04E8>, <U04E9>); (<U04EA>, <U04EB>); /
 1673 (<U04EE>, <U04EF>); (<U04F0>, <U04F1>); (<U04F2>, <U04F3>); (<U04F4>, <U04F5>); /
 1674 (<U04F8>, <U04F9>); (<U0531>, <U0561>); (<U0532>, <U0562>); (<U0533>, <U0563>); /
 1675 (<U0534>, <U0564>); (<U0535>, <U0565>); (<U0536>, <U0566>); (<U0537>, <U0567>); /
 1676 (<U0538>, <U0568>); (<U0539>, <U0569>); (<U053A>, <U056A>); (<U053B>, <U056B>); /
 1677 (<U053C>, <U056C>); (<U053D>, <U056D>); (<U053E>, <U056E>); (<U053F>, <U056F>); /
 1678 (<U0540>, <U0570>); (<U0541>, <U0571>); (<U0542>, <U0572>); (<U0543>, <U0573>); /
 1679 (<U0544>, <U0574>); (<U0545>, <U0575>); (<U0546>, <U0576>); (<U0547>, <U0577>); /
 1680 (<U0548>, <U0578>); (<U0549>, <U0579>); (<U054A>, <U057A>); (<U054B>, <U057B>); /
 1681 (<U054C>, <U057C>); (<U054D>, <U057D>); (<U054E>, <U057E>); (<U054F>, <U057F>); /
 1682 (<U0550>, <U0580>); (<U0551>, <U0581>); (<U0552>, <U0582>); (<U0553>, <U0583>); /
 1683 (<U1E02>, <U1E03>); (<U1E04>, <U1E05>); (<U1E06>, <U1E07>); (<U1E08>, <U1E09>); /
 1684 (<U1E0A>, <U1E0B>); (<U1E0C>, <U1E0D>); (<U1E0E>, <U1E0F>); (<U1E10>, <U1E11>); /
 1685 (<U1E12>, <U1E13>); (<U1E14>, <U1E15>); (<U1E16>, <U1E17>); (<U1E18>, <U1E19>); /
 1686 (<U1E1A>, <U1E1B>); (<U1E1C>, <U1E1D>); (<U1E1E>, <U1E1F>); (<U1E20>, <U1E21>); /
 1687 (<U1E22>, <U1E23>); (<U1E24>, <U1E25>); (<U1E26>, <U1E27>); (<U1E28>, <U1E29>); /
 1688 (<U1E2A>, <U1E2B>); (<U1E2C>, <U1E2D>); (<U1E2E>, <U1E2F>); (<U1E30>, <U1E31>); /
 1689 (<U1E32>, <U1E33>); (<U1E34>, <U1E35>); (<U1E36>, <U1E37>); (<U1E38>, <U1E39>); /
 1690 (<U1E3A>, <U1E3B>); (<U1E3C>, <U1E3D>); (<U1E3E>, <U1E3F>); (<U1E40>, <U1E41>); /
 1691 (<U1E42>, <U1E43>); (<U1E44>, <U1E45>); (<U1E46>, <U1E47>); (<U1E48>, <U1E49>); /
 1692 (<U1E4A>, <U1E4B>); (<U1E4C>, <U1E4D>); (<U1E4E>, <U1E4F>); (<U1E50>, <U1E51>); /
 1693 (<U1E52>, <U1E53>); (<U1E54>, <U1E55>); (<U1E56>, <U1E57>); (<U1E58>, <U1E59>); /
 1694 (<U1E5A>, <U1E5B>); (<U1E5C>, <U1E5D>); (<U1E5E>, <U1E5F>); (<U1E60>, <U1E61>); /
 1695 (<U1E62>, <U1E63>); (<U1E64>, <U1E65>); (<U1E66>, <U1E67>); (<U1E68>, <U1E69>); /
 1696 (<U1E6A>, <U1E6B>); (<U1E6C>, <U1E6D>); (<U1E6E>, <U1E6F>); (<U1E70>, <U1E71>); /
 1697 (<U1E72>, <U1E73>); (<U1E74>, <U1E75>); (<U1E76>, <U1E77>); (<U1E78>, <U1E79>); /
 1698 (<U1E7A>, <U1E7B>); (<U1E7C>, <U1E7D>); (<U1E7E>, <U1E7F>); (<U1E80>, <U1E81>); /
 1699 (<U1E82>, <U1E83>); (<U1E84>, <U1E85>); (<U1E86>, <U1E87>); (<U1E88>, <U1E89>); /
 1700 (<U1E8A>, <U1E8B>); (<U1E8C>, <U1E8D>); (<U1E8E>, <U1E8F>); (<U1E90>, <U1E91>); /
 1701 (<U1E92>, <U1E93>); (<U1E94>, <U1E95>); (<U1EA0>, <U1EA1>); (<U1EA2>, <U1EA3>); /
 1702 (<U1EA4>, <U1EA5>); (<U1EA6>, <U1EA7>); (<U1EA8>, <U1EA9>); (<U1EAA>, <U1EAB>); /
 1703 (<U1EAC>, <U1EAD>); (<U1EAE>, <U1EAF>); (<U1EB0>, <U1EB1>); (<U1EB2>, <U1EB3>); /
 1704 (<U1EB4>, <U1EB5>); (<U1EB6>, <U1EB7>); (<U1EB8>, <U1EB9>); (<U1EBA>, <U1EBB>); /
 1705 (<U1EBC>, <U1EBD>); (<U1EBE>, <U1EBF>); (<U1EC0>, <U1EC1>); (<U1EC2>, <U1EC3>); /
 1706 (<U1EC4>, <U1EC5>); (<U1EC6>, <U1EC7>); (<U1EC8>, <U1EC9>); (<U1ECA>, <U1ECB>); /
 1707 (<U1ECC>, <U1ECD>); (<U1ECE>, <U1ECF>); (<U1ED0>, <U1ED1>); (<U1ED2>, <U1ED3>); /
 1708 (<U1ED4>, <U1ED5>); (<U1ED6>, <U1ED7>); (<U1ED8>, <U1ED9>); (<U1EDA>, <U1EDB>); /

```

1709 (<U1EDC>, <U1EDD>); (<U1EDE>, <U1EDF>); (<U1EE0>, <U1EE1>); (<U1EE2>, <U1EE3>); /
1710 (<U1EE4>, <U1EE5>); (<U1EE6>, <U1EE7>); (<U1EE8>, <U1EE9>); (<U1EEA>, <U1EEB>); /
1711 (<U1EEC>, <U1EED>); (<U1EEE>, <U1EEF>); (<U1EF0>, <U1EF1>); (<U1EF2>, <U1EF3>); /
1712 (<U1EF4>, <U1EF5>); (<U1EF6>, <U1EF7>); (<U1EF8>, <U1EF9>); (<U1F00>, <U1F01>); /
1713 (<U1F02>, <U1F03>); (<U1F04>, <U1F05>); (<U1F06>, <U1F07>); (<U1F08>, <U1F09>); /
1714 (<U1F10>, <U1F11>); (<U1F12>, <U1F13>); (<U1F14>, <U1F15>); (<U1F16>, <U1F17>); /
1715 (<U1F18>, <U1F19>); (<U1F20>, <U1F21>); (<U1F22>, <U1F23>); (<U1F24>, <U1F25>); /
1716 (<U1F26>, <U1F27>); (<U1F28>, <U1F29>); (<U1F30>, <U1F31>); (<U1F32>, <U1F33>); /
1717 (<U1F34>, <U1F35>); (<U1F36>, <U1F37>); (<U1F38>, <U1F39>); (<U1F40>, <U1F41>); /
1718 (<U1F42>, <U1F43>); (<U1F44>, <U1F45>); (<U1F46>, <U1F47>); (<U1F48>, <U1F49>); /
1719 (<U1F50>, <U1F51>); (<U1F52>, <U1F53>); (<U1F54>, <U1F55>); (<U1F56>, <U1F57>); /
1720 (<U1F58>, <U1F59>); (<U1F60>, <U1F61>); (<U1F62>, <U1F63>); (<U1F64>, <U1F65>); /
1721 (<U1F66>, <U1F67>); (<U1F68>, <U1F69>); (<U1F70>, <U1F71>); (<U1F72>, <U1F73>); /
1722 (<U1F74>, <U1F75>); (<U1F76>, <U1F77>); (<U1F78>, <U1F79>); (<U1F80>, <U1F81>); /
1723 (<U1F82>, <U1F83>); (<U1F84>, <U1F85>); (<U1F86>, <U1F87>); (<U1F88>, <U1F89>); /
1724 (<U1F90>, <U1F91>); (<U1F92>, <U1F93>); (<U1F94>, <U1F95>); (<U1F96>, <U1F97>); /
1725 (<U1F98>, <U1F99>); (<U1FA0>, <U1FA1>); (<U1FA2>, <U1FA3>); (<U1FA4>, <U1FA5>); /
1726 (<U1FA6>, <U1FA7>); (<U1FA8>, <U1FA9>); (<U1FAB>, <U1FAC>); (<U1FAD>, <U1FAE>); /
1727 (<U1FAF>, <U1FB0>); (<U1FB2>, <U1FB3>); (<U1FB4>, <U1FB5>); (<U1FB6>, <U1FB7>); /
1728 (<U1FB8>, <U1FB9>); (<U1FBA>, <U1FBB>); (<U1FBC>, <U1FBD>); (<U1FBE>, <U1FBF>); /
1729 (<U1FC0>, <U1FC1>); (<U1FC2>, <U1FC3>); (<U1FC4>, <U1FC5>); (<U1FC6>, <U1FC7>); /
1730 (<U1FC8>, <U1FC9>); (<U1FCA>, <U1FCB>); (<U1FCC>, <U1FCD>); (<U1FCE>, <U1FCF>); /
1731 (<U1FD0>, <U1FD1>); (<U1FD2>, <U1FD3>); (<U1FD4>, <U1FD5>); (<U1FD6>, <U1FD7>); /
1732 (<U1FD8>, <U1FD9>); (<U1FE0>, <U1FE1>); (<U1FE2>, <U1FE3>); (<U1FE4>, <U1FE5>); /
1733 (<U1FE6>, <U1FE7>); (<U1FE8>, <U1FE9>); (<U1FF0>, <U1FF1>); (<U1FF2>, <U1FF3>); /
1734
1735 %
1736 % The "combining" class reflects ISO/IEC 10646-1 annex B.1
1737 % That is, all combining characters (level 2+3).
1738 class "combining"; /
1739 <U0300>..<U036F>; <U20D0>..<U20FF>; <UFE20>..<UFE2F>; /
1740 <U0483>..<U0486>; <U0591>..<U05A1>; <U05A3>..<U05B9>; /
1741 <U05BB>..<U05BD>; <U05BF>; <U05C1>; <U05C2>; <U05C4>; <U064B>..<U0652>; <U0670>; /
1742 <U06D6>..<U06E4>; <U06E7>; <U06E8>; <U06EA>..<U06ED>; <U0901>..<U0903>; <U093C>; /
1743 <U093E>..<U094D>; <U0951>..<U0954>; <U0962>; <U0963>; <U0981>..<U0983>; <U09BC>; /
1744 <U09BE>..<U09C4>; <U09C7>; <U09C8>; <U09CB>..<U09CD>; <U09D7>; <U09E2>; <U09E3>; /
1745 <U0A02>; <U0A3C>; <U0A3E>..<U0A42>; <U0A47>; <U0A48>; <U0A4B>..<U0A4D>; /
1746 <U0A70>; <U0A71>; <U0A81>..<U0A83>; <U0ABC>; <U0ABE>..<U0AC5>; <U0AC7>..<U0AC9>; /
1747 <U0ACB>..<U0ACD>; <U0B01>..<U0B03>; <U0B3C>; <U0B3E>..<U0B43>; <U0B47>; <U0B48>; /
1748 <U0B4B>..<U0B4D>; <U0B56>; <U0B57>; <U0B82>; <U0B83>; <U0BBE>..<U0BC2>; /
1749 <U0BC6>..<U0BC8>; <U0BCA>..<U0BCD>; <U0BD7>; <U0C01>..<U0C03>..<U0C3E>..<U0C44>; /
1750 <U0C46>..<U0C48>; <U0C4A>..<U0C4D>; <U0C55>; <U0C56>; <U0C82>; <U0C83>; /
1751 <U0CBE>..<U0CC4>; <U0CC6>..<U0CC8>; <U0CCA>..<U0CCD>; <U0CD5>; <U0CD6>; /
1752 <U0D02>; <U0D03>; <U0D3E>..<U0D43>; <U0D46>..<U0D48>; <U0D4A>..<U0D4D>; <U0D57>; /
1753 <U0E31>; <U0E34>..<U0E3A>; <U0E47>..<U0E4E>; <U0EB1>; <U0EB4>..<U0EB9>; /
1754 <U0EBB>; <U0EBC>; <U0EC8>..<U0ECD>; <U0F18>; <U0F19>; <U0F35>; <U0F37>; <U0F39>; /
1755 <U0F3E>; <U0F3F>; <U0F71>..<U0F84>; <U0F86>..<U0F87>; <U0F8B>; <U0F90>..<U0F95>; /
1756 <U0F97>; <U0F99>..<U0FAD>; <U0FB1>..<U0FB7>; <U0FB9>; <U302A>..<U302F>; /
1757 <U3099>; <U309A>; <UFB1E>
1758
1759 %
1760 % The "combining_level3" class reflects ISO/IEC 10646-1 annex B.2
1761 % That is, combining characters of level 3.
1762 class "combining_level3"; /
1763 <U0300>..<U036F>; <U20D0>..<U20FF>; <U1100>..<U11FF>; <UFE20>..<UFE2F>; /
1764 <U0483>..<U0486>; <U0591>..<U05A1>; <U05A3>..<U05AF>; <U05C4>; /
1765 <U093C>; <U0953>; <U0954>; <U09BC>; <U09D7>; <U0A3C>; /
1766 <U0A70>; <U0A71>; <U0ABC>; <U0B3C>; <U0B56>; <U0B57>; <U0BD7>; <U0C55>; <U0C56>; /
1767 <U0CD5>; <U0CD6>; <U0D57>; <U0F39>; <U302A>..<U302F>; <U3099>; <U309A>
1768
1769 %
1770 width /
1771 <U200B>; <U200C>; <U200D>; <U200E>; <U200F>; <U202A>; <U202B>; /
1772 <U202C>; <U202D>; <U202E>; <UFEFF> : 0; /
1773 <U1100>..<U115F>; <U2E80>..<U3009>; <U300C>..<U3019>; /
1774 <U301C>..<U303E>; <U3040>..<UA4CF>; <UAC00>..<UD7A3>; /
1775 <UF900>..<UFAFF>; <UFE30>..<UFE6F>; <UFFF0>..<UFFF5F>; /
1776 <UFFE0>..<UFFE6> : 2
1777 END LC_CTYPE

```

4.4 LC_COLLATE

A collation sequence definition defines the relative order between collating elements (characters and multicharacter collating elements) in the FDCC-set. This order is expressed in terms of collation values; i.e., by assigning each element one or more collation values

1784 (also known as collation weights). This does not imply that applications assign such
 1785 values, but that ordering of strings using the resultant collation definition in the FDCC-set
 1786 behaves as if such assignment is done and used in the collation process. The collation
 1787 sequence definition is used by regular expressions, pattern matching. When no weights are
 1788 specified the collation sequence definition also is used for sorting, else the weighting
 1789 defines the sorting. The following capabilities are provided:

- 1790
- 1791 (1) Multicharacter collating elements. Specification of multicharacter collating elements
 1792 (i.e., sequences of two or more characters to be collated as an entity).
 - 1793 (2) User-defined ordering of collating elements. Each collating element is assigned a
 1794 collation value defining its order in the character (or basic) collation sequence. This
 1795 ordering is used by regular expressions and pattern matching and, unless collation
 1796 weights are explicitly specified, also as the collation weight to be used in sorting.
 - 1797 (3) Multiple weights and equivalence classes. Collating elements can be assigned one
 1798 or more (up to the limit (COLL_WEIGHTS_MAX)) collating weights for use in
 1799 sorting. The first weight is hereafter referred to as the primary weight.
 - 1800 (4) One-to Many mapping. A single character is mapped into a string of collating
 1801 elements.
 - 1802 (5) Many-to-Many substitution. A string of one or more characters is substituted by
 1803 another string (or an empty string, i.e., the character or characters are ignored for
 1804 collation purposes).
 - 1805 (6) Equivalence class definition. Two or more collating elements have the same
 1806 collation value (primary weight).
 - 1807 (7) Ordering by weights. When two strings are compared to determine their relative
 1808 order, the two strings are first broken up into a series of collating elements, and
 1809 each successive pair of elements are compared according to the relative primary
 1810 weights for the elements. If equal, and more than one weight has been assigned,
 1811 then the pairs of collating elements are recompared according to the relative
 1812 subsequent weights, until either a pair of collating elements compare unequal or the
 1813 weights are exhausted.
 - 1814 (8) Easy reordering of characters. ISO/IEC 14651 has a template for collation
 1815 specification that with just a few modifications can be culturally correct for a
 1816 specific culture. Here the "reorder-after" keyword gives a convenient way to
 1817 modify a FDCC-set template.
 - 1818 (9) Easy reordering of sections. The template in ISO/IEC 14651 gives an ordering of
 1819 the sections that may not be culturally acceptable in certain cultures. The keyword
 1820 "reorder-section-after" gives a convenient way to modify the order of sections in a
 1821 FDCC-set template.

1822

1823

1824 The following keywords are recognized in a collation sequence definition. Some of them
 1825 are described in detail in the following subclauses. The keywords are mandatory unless
 1826 otherwise noted.

1827	
1828	copy
1829	
1830	Specify the name of an existing FDCC-set to be used
1831	as the source for the definition of this category. If
1832	this keyword is specified, only the "reorder-after",
1833	"reorder-end", "reorder-section-after" and "reorder-
1834	section-end" keywords may also be specified. The
1835	FDCC-set is copied in source form.
	coll_weight_max
	Define as a decimal number the number of collation
	levels that an interpreting system needs to support

1836		for this FDCC-set, this value is elsewhere referred to
1837		as the COLL_WEIGHT_MAX limit (e.g. in the
1838		"order_start" statement). An interpreting system
1839		caters for up to 7 collating levels.
1840	section-symbol	Define a section symbol representing a set of
1841		collation order statements. The section is defined
1842		with the "order_start" keyword until the next
1843		"order_start" or "order_end" keyword. This keyword
1844		is optional.
1845	collating-element	Define a collating-element symbol representing a
1846		multicharacter collating element. This keyword is
1847		optional.
1848	collating-symbol	Define one or more collating symbols for use in
1849		collation order statements. This keyword is optional.
1850	symbol-equivalence	Define a collating-symbol to be equivalent to another
1851		defined collating-symbol.
1852	order_start	Define collation rules. This statement is followed by
1853		one or more collation order statements, assigning
1854		character collation values and collation weights to
1855		collating elements.
1856	order_end	Specify the end of the collation-order statements.
1857	section	Specify a section of collation order statements, and
1858		optionally a subrepertoire thereof.
1859	reorder-after	Redefine collating rules. Specify after which
1860		collating element the redefinition of collation order
1861		takes order. This statement is followed by one or
1862		more collation order statements, reassigning character
1863		collation values and collation weights to collating
1864		elements.
1865	reorder-end	Specify the end of the "reorder-after" collating order
1866		statements.
1867	reorder-section-after	Redefine the order of sections. This statement is
1868		followed by one or more section symbols,
1869		reassigning character collation values and collation
1870		weights to collating elements.
1871	reorder-section-end	Specify the end of the "reorder-section" section order
1872		statements.

4.4.1 Collation statements

The "order_start", "reorder-after" and "section" keywords are followed by collating statements. The syntax for the collating statements is

```
"%s %s;%s;...;%s\n", <collating-identifier>, <weight>, <weight>, ...
```

Each <collating-identifier> consists of either a character (in any of the forms defined in 4.1.1), a <collating-element>, a <collating-symbol>, an ellipsis, or the special symbol "UNDEFINED". The weights for each of the collation elements determines the character collation sequence - such that each collation statement does not need to be in collation order, and weights could be rearranged via for example the "reorder-after" keyword. No character has any specific predetermined placement in the collation sequence. The order in which collating elements are specified determines the character collation sequence, such

1888 that each collating element compares less than the elements following it.
1889

1890 A <collating-element> is used to specify multicharacter collating elements, and indicates
1891 that the character sequence specified via the <collating-element> is to be collated as a unit
1892 and in the relative order specified by its place in the list of collating statements.
1893

1894 A <collating-symbol> is used to define a position in the relative order for use in weights.
1895

1896 The absolute ellipsis symbol ("...") specifies that a sequence of characters collate according
1897 to their encoded character values. It is interpreted as indicating that all characters with a
1898 coded character set value higher than the value of the character in the preceding line, and
1899 lower than the coded character set value for the character in the following line, in the
1900 current coded character set, are placed in the character collation order between the
1901 previous and the following character in ascending order according to their coded character
1902 set values. An initial ellipsis is interpreted as if the preceding line specified the <NUL>
1903 character, and a trailing ellipsis as if the following line specified the highest coded
1904 character set value in the current coded character set. An ellipsis is treated as invalid if the
1905 preceding or following lines do not specify characters in the current coded character set.
1906 The use of the ellipsis symbol ties the definition to a specific coded character set and may
1907 preclude the definition from being portable between applications, and is depreciated.
1908 Symbolic ellipses may be used as the ellipses symbol, but generating symbolic character
1909 names, and thus have a better chance of portability between applications.
1910

1911 The symbolic ellipses (".." or "....") specifies a sequence of collating statements. It is
1912 interpreted as indicating that all characters with symbolic names higher than the symbolic
1913 name of the character in the preceding line, and lower in the sequence of symbolic names
1914 for the character in the following line, is placed in the character collation order between
1915 the previous and the following character in ascending order.
1916

1917 The symbol "UNDEFINED" is interpreted as including all coded character set values not
1918 specified explicitly or via the ellipsis or one of the symbolic ellipses symbols. Such
1919 characters are inserted in the character collation order at the point indicated by the symbol,
1920 and in ascending order according to their coded character set values. If no "UNDEFINED"
1921 symbol is specified, and the current coded character set contains characters not specified
1922 in this clause, the utility issues a warning message and place such characters at the end of
1923 the character collation order.
1924

1925 The optional operands for each collation-element are used to define the primary,
1926 secondary, or subsequent weights for the collating element. The first operand specifies the
1927 relative primary weight, the second the relative secondary weight, and so on. Two or more
1928 collation-elements can be assigned the same weight; they belong to the same equivalence
1929 class if they have the same primary weight. Collation behaves as if, for each weight level,
1930 "IGNORE"d elements are removed. Then each successive pair of elements is compared
1931 according to the relative weights for the elements. If the two strings compare equal, the
1932 process is repeated for the next weight level, up to the limit "COLL_WEIGHTS_MAX" of
1933 the associated FDCC-set.
1934

1935 Weights are expressed as characters (in any of the forms specified here), <collating-
1936 symbol>s, <collating-element>s, an ellipsis, or the special symbol "IGNORE". A single
1937 character, a <collating-symbol>, or a <collating-element> represent the relative order in
1938 the character collating sequence of the character or symbol, rather than the character or
1939 characters themselves.

1940 One-to-many mapping is indicated by specifying two or more concatenated characters or
 1941 symbolic names. Thus, if the character <ss> is given the string <s><s> as a weight,
 1942 comparisons are performed as if all occurrences of the character <ss> are replaced by
 1943 <s><s>. If it is desirable to define <ss> and <s><s> as an equivalence class, then a
 1944 collating-element must be defined for the string "ss", as in the example below.
 1945

1946 All characters specified via an ellipsis are by default assigned unique weights, equal to the
 1947 relative order of characters. Characters specified via an explicit or implicit "UNDEFINED"
 1948 special symbol are by default assigned the same primary weight (i.e., belong to the same
 1949 equivalence class). An ellipsis symbol as a weight is interpreted to mean that each
 1950 character in the sequence has unique weights, equal to the relative order of their character
 1951 in the character collation sequence. Secondary and subsequent weights have unique values.
 1952 The use of the ellipsis as a weight is treated as an error if the collating element is neither
 1953 an ellipsis nor the special symbol "UNDEFINED".
 1954

1955 The special keyword "IGNORE" as a weight indicates that when strings are compared
 1956 using the weights at the level where "IGNORE" is specified, the collating element is
 1957 ignored; i.e., as if the string did not contain the collating element. In regular expressions
 1958 and pattern matching, all characters that are "IGNORE"d in their primary weight form an
 1959 equivalence class.
 1960

1961 A <comment_character> occurring where the delimiter ";" may occur, terminates the
 1962 collating statement.
 1963

1964 An empty operand is interpreted as the collating-element itself.
 1965

1966 For example, the collation statement

```
1967     <a>  <a>;<a>
```

1969 is equal to

```
1970     <a>
```

1971
 1972
 1973
 1974 An ellipsis (absolute or symbolic) can be used as an operand if the collating-element was
 1975 an ellipsis, and is interpreted as the value of each character defined by the ellipsis.
 1976

1977 Example:

```
1978
1979 collating-element <ch> from "<c><h>"
1980 collating-element <Ch> from "<C><h>"
1981 order_start      forward;backward
1982 UNDEFINED      IGNORE;IGNORE
1983 <LOW>
1984 <space>        <LOW>;<space>
1985 ...           <LOW>;
1986 <a>           <a>;<a>
1987 <a'>         <a>;<a'>
1988 <A>          <a>;<A>
1989 <A'>        <a>;<A'>
1990 <ch>         <ch>;<ch>
1991 <Ch>         <ch>;<Ch>
1992 <s>          <s>;<s>
1993 <ss>         "<s><s>" ; "<ss><ss>"
1994 order_end
```

1995
 1996 This example is interpreted as follows:
 1997
 1998

- 1999 (1) The UNDEFINED means that all characters not specified in this definition (explicitly or via the
2000 ellipsis) is ignored.
- 2001 (2) <LOW> defines the first collating weight, and thus the lowest weight in this example.
- 2002 (3) All characters between <space> and <a> have the same primary equivalence class <LOW> and
2003 individual secondary weights based on their ordinal encoded values. (The use of absolute ellipses is
2004 deprecated, but used here to illustrate generic use of ellipses. Symbolic ellipses should be used
2005 instead).
- 2006 (4) All characters based on the upper or lowercase character "a" belong to the same primary equivalence
2007 class.
- 2008 (5) The multicharacter collating element <c><h> is represented by the collating symbol <ch> and belongs
2009 to the same primary equivalence class as the multicharacter collating element <C><H>.
- 2010 (6) The <ss> collating element has two weights on the primary level, and it is in the same primary
2011 equivalence class as two consecutive <s>-es; on the secondary level the collating element has two
2012 weights of the equivalence class <ss>.

2013 4.4.2 "copy" keyword

2014 This keyword specifies the name of an existing FDCC-set to be used as the source for the
2015 definition of this category. The syntax is

2016 "copy %s\n", <FDCC-set-name>

2017 The <FDCC-set-name> consists of one or more characters (in any of the forms defined in
2018 4.1.1). The FDCC-set is copied in source form.

2019 4.4.3 "coll_weight_max" keyword

2020 This keyword defines as a decimal number the number of collation levels that an
2021 interpreting system needs to support. An interpreting system caters for up to 7 collating
2022 levels. The syntax is

2023 "coll_weight_max %d\n", <value>

2024 4.4.4 "section-symbol" keyword

2025 This keyword is used to define symbols for use in section related statements; such as the
2026 "order_start", and "reorder-section-after" keywords and section-reordering statements. The
2027 syntax is

2028 "section-symbol %s\n", <section-symbol>

2029 The <section-symbol> is a symbolic name, enclosed between angle brackets (< and >),
2030 and does not duplicate any symbolic name in the current charmap (if any), or any other
2031 symbolic name defined in this collation definition. A <section-symbol> defined via this
2032 keyword is only defined within the LC_COLLATE category.

2033 Example:
2034 section-symbol <LATIN>
2035 section-symbol <ARABIC>

2036 4.4.5 "collating-element" keyword

2037 In addition to the collating elements in the character set, the collating-element keyword is
2038 used to define multicharacter collating elements. The syntax is

2039 "collating-element %s from %s\n", <collating-symbol>, <string>

2055 The <collating-symbol> operand is a symbolic name, enclosed between angle brackets (<
 2056 and >), and does not duplicate any symbolic name in the current charmap or repertoire
 2057 file (if any), or any other symbolic name defined in this collation definition. The string
 2058 operand is a string of two or more characters that collates as an entity. A <collating-
 2059 element> defined via this keyword is only defined within the LC_COLLATE category.

2060
 2061 Example with ISO/IEC 10646-1:
 2062 collating-element <ch> from "<c><h>"
 2063 collating-element <e-acute> from "<e><combining-acute>"
 2064 collating-element <aa> from "<a><a>"

2065
 2066 Note: The problem of comparing a fully composed character of ISO/IEC 10646 with a decomposed
 2067 representation of the same text is sometimes handled by the two strings comparing equal up to level 3 (the
 2068 case level) of ISO/IEC 14651, but distinguishing the two at the 4th level.

2070 4.4.6 "collating-symbol" keyword

2071
 2072 This keyword is used to define symbols for use in collation sequence statements; e.g.,
 2073 between the order_start and the order_end keywords. The syntax is

2074
 2075 "collating-symbol %s;%s;...%s\n", <collating-symbol>, <collating-symbol> ...

2076
 2077 The <collating-symbol> is a symbolic name, enclosed between angle brackets (< and >),
 2078 and does not duplicate any symbolic name in the current charmap (if any), or any other
 2079 symbolic name defined in this collation definition. A <collating-symbol> defined via this
 2080 keyword is only defined within the LC_COLLATE category. More than one <collating-
 2081 symbol> may be defined with one "collating-symbol" keyword, and symbolic ellipses may
 2082 be used.

2083
 2084 Example:
 2085 collating-symbol <CAPITAL>
 2086 collating-symbol <HIGH>

2088 4.4.7 "symbol-equivalence" keyword

2089
 2090 This keyword is used to define symbols for use in collation sequence statements; and
 2091 assign the same weight as another defined symbol. The syntax is

2092
 2093 "symbol-equivalence %s %s\n", <collating-symbol-1>, <collating-symbol-2>

2094
 2095 The <collating-symbol-1> and <collating-symbol-2> are symbolic names, enclosed
 2096 between angle brackets (< and >). <collating-symbol-1> does not duplicate any symbolic
 2097 name in the current charmap (if any), or any other symbolic name defined in this collation
 2098 definition. <collating-symbol-2> is defined elsewhere in the LC_COLLATE category as a
 2099 collating-symbol. The use of <collating-symbol-2> is equivalent to using the <collating-
 2100 symbol-1> in the LC_COLLATE category. A <collating-symbol-1> defined via this
 2101 keyword is only defined within the LC_COLLATE category.

2102
 2103 Example
 2104 collating-symbol <CAP>
 2105 symbol-equivalence <CAPITAL> <CAP>

2109 4.4.8 "order_start" keyword

2110 The "order_start" keyword precedes collation order entries and also defines the number of
2111 weights for this collation sequence definition, the collation section name and other
2112 collation rules.
2113

2114 The syntax of the "order_start" keyword has two forms:
2115

2116 "order_start %s;%s;...;%s\n", <sort-rule>, <sort-rule> ...
2117

2118 and

2119 "order_start %s;%s;...;%s\n", <section-symbol>, <sort-rules>, <sort-rules> ...
2120

2121 The operands to the order_start keyword are optional. If present, the operands define rules
2122 to be applied when strings are compared. The first operand may be a <section-symbol>
2123 surrounded by "<" and ">" and the set of collating statements following the "order_start"
2124 keyword until the "order_end" keyword are identified with this <section-symbol> or
2125 another "order_start" keyword is encountered. The remaining number of operands define
2126 how many weights each element is assigned; if no operands are present, one forward
2127 operand is assumed. If present, the first operand defines rules to be applied when
2128 comparing strings using the first (primary) weight; the second when comparing strings
2129 using the second weight, and so on. Operands are separated by semicolons (;). Each
2130 operand consists of one or more collation directives, separated by commas (.). If the
2131 number of operands exceeds the (COLL_WEIGHTS_MAX) limit, a utility parsing the
2132 FDCC-set description issues a warning message. The following directives are supported:
2133

2134 **forward** Specifies that the direction of scanning a part of a string at a given point in a
2135 string is done towards the logical end of the whole string for this weight level.

2136 **backward** Specifies that the direction of scanning a part of a string at a given point in a
2137 string is done towards the logical beginning of the whole string for this weight
2138 level.

2139 **position** Specifies that comparison operations for the weight level will consider the
2140 relative position of non-"IGNORE"d elements in the strings. The string
2141 containing a non-"IGNORE"d element after the fewest IGNOREd collating
2142 elements from the start of the compare collates first. If both strings contain a
2143 non-"IGNORE"d character in the same relative position, the collating values
2144 assigned to the elements determine the ordering. In case of equality,
2145 subsequent non-IGNOREd characters are considered in the same manner.
2146

2147 The directives "forward" and "backward" are mutually exclusive at a given level. The
2148 directives "backward" and "position" are mutually exclusive at a given level.
2149

2150 Examples:

2151 order_start forward;backward

2152 order_start <CYRILLIC>;forward;forward
2153

2154 If no operands are specified, a single forward operand is assumed.
2155

2156 4.4.9 "order_end" keyword

2157 The collating order entries are terminated with an "order_end" keyword.
2158
2159
2160
2161

2162 **4.4.10 "reorder-after" keyword**

2163
2164 The "reorder-after" keyword is used to specify a modification to a copied collation
2165 specification of an existing FDCC-set. There can be more than one "reorder-after"
2166 statement in a collating specification. The syntax is:

2167
2168 `"reorder-after %s\n",<collating-symbol>`
2169

2170 The <collating-symbol> operand is a symbolic name, enclosed between angle brackets,
2171 and is present in the source FDCC-set copied via the "copy" keyword.

2172 The "reorder-after" statement is followed by one or more collation statements as described
2173 in the "Collating Order" clause (4.4.5), with the exception that the ellipsis symbol (...) is
2174 not used.

2175
2176 Each collation statement reassigns character collation values and collation weights to
2177 collating elements existing in the copied collation specification, by removing the collating
2178 statement from the copied specification, and inserting the collating element in the collating
2179 sequence with the new collation weights after the preceding collating element of the
2180 "reorder-after" specification, the first collating element in the collation sequence being the
2181 <collating-symbol> specified in the "reorder-after" statement.
2182

2183 A "reorder-after" specification is terminated by another "reorder-after" specification or the
2184 "reorder-end" statement.
2185

2186
2187 **4.4.10.1 Example of "reorder-after"**

```
2188 reorder-after <y8>
2189 <U:>      <Y>;<U:>;<CAPITAL>
2190 <u:>      <Y>;<U:>;<SMALL>
2191 reorder-after <z8>
2192 <AE>      <AE>;<NONE>;<CAPITAL>
2193 <ae>      <AE>;<NONE>;<SMALL>
2194 <A:>      <AE>;<DIAERESIS>;<CAPITAL>
2195 <a:>      <AE>;<DIAERESIS>;<SMALL>
2196 <O/>      <O/>;<NONE>;<CAPITAL>
2197 <o/>      <O/>;<NONE>;<SMALL>
2198 <AA>      <AA>;<NONE>;<CAPITAL>
2199 <aa>      <AA>;<NONE>;<SMALL>
2200 reorder-end
```

2201
2202 The example is interpreted as follows (using the "i18nrep" repertoire map):
2203

- 2204
- 2205 1. The collating element <U:> is removed from the copied collating sequence and inserted after <y8> in the
2206 collating sequence with the new weights. The collating element <u:> is removed from the copied collating
2207 sequence and inserted in the resulting collation sequence after <U:> with the new weights. <y8> is used to
2208 indicate the position of the last y letter.
2209
 - 2210 2. The second "reorder-after" statement terminates the first list of reordering collation identifier entries, and
2211 initiates a second list, rearranging the order and weights for the <AE>, <ae>, <A:>, <a:>, <O/>, and <o/>
2212 collating elements after the <z8> collating symbol in the copied specification. <z8> is used to indicate the
2213 position of the last z letter.
2214
 - 2215 3. The "reorder-end" statement terminates the second list of reordering entries.
2216
 - 2217 4. Thus for the original sequence
2218
2219 ... (U u Ü ü) V v W w X x Y y Z z
2220
2221 this example reordering gives

2222 ... U u V v W w X x (Y y Ü ü) Z z (Æ æ Ä ä) Ø ø Å å

2223

2224 where the parenthesis indicate ordering with the same weight on the first level for multiple upper/lowercase
2225 pairs.

2226

2227 **4.4.11 "reorder-end" keyword**

2228

2229 The "reorder-end" keyword specifies the end of a list of collating statements, initiated by
2230 the "reorder-after" keyword.

2231

2232 **4.4.12 "section" keyword**

2233

2234 The "section" keyword is used to define a section of the table. A section consists of a set
2235 of collation elements with their associated collation weights. A section can be moved as a
2236 whole via the "reorder-section-after" keyword.

2237

2238 Each "section" keyword has the syntax:

2239

2240 "section %s %s;...;%s\n", <section-symbol>, <collation-symbol>,

2241

2242 The <section-symbol> is a symbolic name, enclosed between angle brackets "<" and ">",
2243 and it defines the name of the section in question. It may have been defined in a "section-
2244 symbol" statement.

2245

2246 The <collation-symbol> is a symbolic name, enclosed between angle brackets "<" and ">",
2247 and it references a collating element previously specified, with associated weights. More
2248 than one <collating-symbol> may be referenced in one "section" statement, and symbolic
2249 ellipses may be used. The <collation-symbol>s identified via this list are removed from
2250 other parts of the collation specification. The list of <collation-symbol>s is optional.

2251

2252 A section consists of the collating elements identified on the "section" keyword line and
2253 with relative order and weights as specified earlier, plus the collation elements defined via
2254 the optionally following collating statements as described in 4.4.1. The section is
2255 terminated by another keyword line.

2256

2257 **4.4.13 "reorder-section-after" keyword**

2258

2259 The "reorder-section-after" keyword is used to specify a modification to a copied collation
2260 specification of an existing FDCC-set. The "reorder-section-after" statement is followed by
2261 one or more statements consisting of section reordering statements.

2262

2263 Each "reorder-section-after" keyword has either the syntax:

2264

2265 "reorder-section-after %s\n", <collation-symbol>

2266

2267 or:

2268

2269 "reorder-section-after %s %s", <section-symbol>, <collation-symbol>

2270

2271 The <collation-symbol> is a symbolic name, enclosed between angle brackets "<" and ">",
2272 and it references a collating element previously specified.

2273

2274

2275 The <section-symbol> is a symbolic name, enclosed between angle brackets "<" and ">",
 2276 and it refers to the name of the section in question, previously defined in a "section-
 2277 symbol" or "section" keyword, and with contents allocated via a "order_start" or "section"
 2278 keyword.

2279
 2280 If there is no <section-symbol> given with the keyword, the keyword is followed by a
 2281 number of section reordering statements, terminated by a "reorder-section-end" keyword.
 2282

2283 The collating elements and associated weights of the section given with the keyword line,
 2284 or the sections given on the following lines, are removed from the current sorting table,
 2285 possibly reassigned sorting rules according to the section reordering statements, and
 2286 inserted in the sorting table after the <collating-symbol>.
 2287

2288 **4.4.13.1 section reordering statements**

2289
 2290 The section reordering statements rearranges the set of collating entries and changes
 2291 sorting rules for the set of collating entries identified by a section symbol in a preceding
 2292 "order_start" statement. Each section reorder statement has the syntax:
 2293

```
2294 "%s %s;...%s\n", <section-symbol>, <sort-rule>, <sort-rule> ...
```

2295
 2296 The <section-symbol> identifies the set of collating entries. The <section-symbol> is
 2297 defined via a "section-symbol" or the "section" keyword, and values identified by the
 2298 <section-symbol> is assigned via the "order_start" or "section" keywords.
 2299

2300 The <sort-rule>s are as described for the "order_start" keyword. Specified <sort-rule>s
 2301 replace the specification of the ordering given on the first "order_start" statement, for the
 2302 section identified by the <section-symbol> . The <sort-rule>s are optional, and <sort-
 2303 rule>s not to be changed from the first "order_start" specification is given by empty
 2304 specifications on the "section" statement.
 2305

2306 Note: The <sort-rule> capability is an extension over ISO/IEC 14651 functionality.
 2307

2308 The order of the section reordering statements rearranges the assignment of collation
 2309 entries for the sets of collation entries identified by the <section-symbols> to the order
 2310 that the <section-symbols> occur after the "reorder-section-after" statement.
 2311

2312 The section reordering statements are terminated by a "reorder-section-end" statement.
 2313

2314 **4.4.13.2 Example of section reordering**

```
2315 copy "i18n"  

  2316 section <DEVANAGARI> <U0905>..<U0939>;<U093D>..<U0950>  

  2317 reorder-section-after <DEVANAGARI> <U3361>
```

2318
 2319 This example is interpreted as follows: The LC_COLLATE category of the "i18n" FDCC-set is copied. Then a
 2320 definition of the section <DEVANAGARI> is done, and the collating elements of this section is removed from
 2321 the table and inserted in the same relative order and with the same weights after the collating element <U3361>,
 2322 which is the last of the digits. In this way the <DEVANAGARI> section is reordered to be sorted before all other
 2323 letters.
 2324

2325 **4.4.14 "reorder-section-end" keyword**

2326
 2327 The "reorder-section-end" keyword specifies the end of a list of section symbols, initiated
 2328 by the "reorder-section-after" keyword.
 2329

2330 4.4.15 "i18n" LC_COLLATE category

2331
2332 The "i18n" LC_COLLATE category is defined as the following, which includes the
2333 tailorable template in ISO/IEC 14651.
2334

```

2335
2336 LC_COLLATE
2337 % This is the ISO/IEC TR 14652 i18n fdcc-set definition for
2338 % the LC_COLLATE category.
2339 %
2340 % equivalences
2341 symbol-equivalence <NONE> <BLANK>
2342 symbol-equivalence <CAPITAL> <CAP>
2343 symbol-equivalence <SMALL> <MIN>
2344 symbol-equivalence <CAPITAL-SMALL> <COMPATCAP>
2345 symbol-equivalence <SMALL-CAPITAL> <COMPAT>
2346 symbol-equivalence <MACRON> <MACRO>
2347 symbol-equivalence <STROKE> <OBLIK>
2348 symbol-equivalence <ACUTE> <AIGUT>
2349 symbol-equivalence <CIRCUMFLEX> <CIRCF>
2350 symbol-equivalence <RING> <CRCLE>
2351 symbol-equivalence <DIAERESIS> <TREMA>
2352 symbol-equivalence <DOT> <POINT>
2353 symbol-equivalence <CEDILLA> <CEDIL>
2354 symbol-equivalence <OGONEK> <OGONK>
2355 symbol-equivalence <HOOK> <CROOK>
2356 symbol-equivalence <HORN> <HORNU>
2357 symbol-equivalence <DOT-BELOW> <POINS>
2358
2359 order_start forward;forward;forward;forward,position
2360
2361 % Copy the template from ISO/IEC 14651
2362 copy "ISO14651_2000_TABLE1.txt"
2363
2364 order_end
2365
2366 END LC_COLLATE

```

2367 4.5 LC_MONETARY (controversial)

2370 The LC_MONETARY category defines the rules and symbols that are used to format
2371 monetary numeric information. The operands are strings. For some keywords, the strings
2372 can contain only integers. More than one set of monetary values may be provided, and for
2373 each set a period of validity and conversion rate may be given. Keywords that are not
2374 provided, string values set to the empty string "", or integer keywords set to -1, are used
2375 to indicate that the value is unspecified, and then no default is implied. The following
2376 keywords are defined:

2377	
2378	copy
2379	
2380	
2381	valid_from
2382	
2383	
2384	
2385	
2386	
2387	
2388	
2389	
2390	

Specify the name of an existing FDCC-set to be used as the source for the definition of this category. If this keyword is specified, no other keyword is specified.

One or more strings separated by semicolons, representing a Gregorian date in the form "YYYYMMDD" according to ISO 8601, specifying the beginning date (inclusive from the beginning of day local time) of the validity of a currency. The position of the string in the list corresponds to the position of operands in other keywords in the LC_MONETARY category. The currencies should be ordered in terms of validity dates, and for each validity period with the currency that the amounts are stored in first. If not specified, it is taken to be an implementation-defined

2391		beginning of time. This keyword is optional.
2392	valid_to	One or more strings, separated by semicolons, each
2393		representing a Gregorian date in the form "YYYYMMDD"
2394		according to ISO 8601, that specify the last date (inclusive
2395		to the end of day local time) of the validity of a currency. If
2396		not specified, it is taken to be an implementation-defined end
2397		of time. This keyword is optional.
2398	conversion_rate	one or more pairs of integers separated by a <semicolon>
2399		specifying the fixed conversion rate between the current
2400		currency (determined by the parameter number) and the first
2401		currency that is valid, determined by a date provided by the
2402		application. If the currency is not the first valid currency for
2403		the period in question, the first integer is for multiplying the
2404		first valid currency, and the second for dividing this result to
2405		get the amount in the current currency. The currency to be
2406		the current currency is selected by the application from the
2407		date applicable and the currency number (first, second, third
2408		etc valid currency at that date); and whether domestic or
2409		international formatting is used is also determined by the
2410		application. Each pair of integers are separated by a <slash>.
2411		The default value is "1/100". This keyword is optional.
2412		Note: The two integers are used instead of a floating point
2413		value, to be able to cater for legal requirements on Euro
2414		conversion where a multiplication and division is prescribed,
2415		instead of just one floating point multiplication.
2416	currency_symbol	One or more strings separated by semicolons that are used as
2417		the local currency symbol.
2418	mon_decimal_point	The operand is a string containing the symbol that is used as
2419		the decimal delimiter in monetary formatted quantities. In
2420		contexts where other standards limit the "mon_decim-
2421		al_point" to a single byte, the result of specifying a
2422		multibyte operand is unspecified. The keyword is specified,
2423		unless the "copy" keyword is used.
2424	mon_thousands_sep	The operand is a string containing the symbol that is used as
2425		a separator for groups of digits to the left of the decimal
2426		delimiter in formatted monetary quantities. In contexts where
2427		other standards limit the "mon_thousands_sep" to a single
2428		byte, the result of specifying a multibyte operand is
2429		unspecified. The keyword is specified, unless the "copy"
2430		keyword is used.
2431	mon_grouping	Define the size of each group of digits in formatted
2432		monetary quantities. The operand is a sequence of integers
2433		separated by semicolons. Each integer specifies the number
2434		of digits in each group, with the initial integer defining the
2435		size of the group immediately preceding the decimal
2436		delimiter, and the following integers defining the preceding
2437		groups. If the last integer is not -1, then the size of the
2438		previous group (if any) is repeatedly used for the remainder
2439		of the digits. If the last integer is -1, then no further
2440		grouping is performed. The keyword is specified, unless the
2441		"copy" keyword is used.
2442	positive_sign	A string that is used to indicate a nonnegative-valued

2443		formatted monetary quantity. The keyword is specified,
2444		unless the "copy" keyword is used.
2445	negative_sign	A string that is used to indicate a negative-valued formatted
2446		monetary quantity. The keyword is specified, unless the
2447		"copy" keyword is used.
2448	frac_digits	One or more integers separated by semicolons, representing
2449		the number of fractional digits (those to the right of the
2450		decimal delimiter) to be written in a formatted monetary
2451		quantity using "currency_symbol". The keyword is specified,
2452		unless the "copy" keyword is used.
2453	p_cs_precedes	One or more integers separated by semicolons, set to 1 if the
2454		"currency_symbol" precedes the value for a nonnegative
2455		formatted monetary quantity, and set to 0 if the symbol
2456		succeeds the value. The keyword is specified, unless the
2457		"copy" keyword is used.
2458	p_sep_by_space	One or more integers separated by semicolons, set to 0 if no
2459		space separates the "currency_symbol" from the value for a
2460		nonnegative formatted monetary quantity, set to 1 if a space
2461		separates the symbol from the value, and set to 2 if a space
2462		separates the symbol and the sign string, if adjacent. The
2463		keyword is specified, unless the "copy" keyword is used.
2464	n_cs_precedes	One or more integers separated by semicolons, set to 1 if the
2465		"currency_symbol" precedes the value for a negative
2466		formatted monetary quantity, and set to 0 if the symbol
2467		succeeds the value. The keyword is specified, unless the
2468		"copy" keyword is used.
2469	n_sep_by_space	One or more integers separated by semicolons, set to 0 if no
2470		space separates the "currency_symbol" from the value for a
2471		negative formatted monetary quantity, set to 1 if a space
2472		separates the symbol from the value, and set to 2 if a space
2473		separates the symbol and the sign string, if adjacent. The
2474		keyword is specified, unless the "copy" keyword is used.
2475	p_sign_posn	One or more integers separated by semicolons, set to a value
2476		indicating the positioning of the "positive_sign" for a
2477		nonnegative formatted monetary quantity using the
2478		"currency_symbol". The following integer values are defined:
2479		
2480		0 Parentheses enclose the quantity and the
2481		"currency_symbol".
2482		1 The sign string precedes the quantity and the
2483		"currency_symbol".
2484		2 The sign string succeeds the quantity and the
2485		"currency_symbol".
2486		3 The sign string immediately precedes the
2487		"currency_symbol".
2488		4 The sign string immediately succeeds the
2489		"currency_symbol".
2490		The keyword is specified, unless the "copy" keyword is used.
2491		
2492	n_sign_posn	One or more integers separated by semicolons, set to a value
2493		indicating the positioning of the "negative_sign" for a
2494		negative formatted monetary quantity using the

2495		"currency_symbol". The following integer values are defined:
2496		
2497	0	Parentheses enclose the quantity and the
2498		"currency_symbol".
2499	1	The sign string precedes the quantity and the
2500		"currency_symbol".
2501	2	The sign string succeeds the quantity and the
2502		"currency_symbol".
2503	3	The sign string immediately precedes the
2504		"currency_symbol".
2505	4	The sign string immediately succeeds the
2506		"currency_symbol".
2507		The keyword is specified, unless the "copy" keyword is used.
2508	int_curr_symbol	One or more strings separated by semicolons that are used as
2509		the international currency symbols. Each operand is a four
2510		character string, with the first three characters containing the
2511		alphabetic international currency symbol in accordance with
2512		those specified in ISO 4217, <i>Codes for the representation of</i>
2513		<i>currencies and funds</i> . The fourth character is the character
2514		used to separate the international currency symbol from the
2515		monetary quantity. The keyword is specified, unless the
2516		"copy" keyword is used.
2517	int_frac_digits	One or more integers separated by semicolons, representing
2518		the number of fractional digits (those to the right of the
2519		decimal delimiter) to be written in a formatted monetary
2520		quantity using "int_curr_symbol". The keyword is specified,
2521		unless the "copy" keyword is used.
2522	int_p_cs_precedes	One or more integers separated by semicolons; set to 1 if the
2523		"int_curr_symbol" precedes the value for a nonnegative
2524		formatted monetary quantity, and set to 0 if the symbol
2525		succeeds the value. If not specified, the value of
2526		"p_cs_precedes" is taken.
2527	int_p_sep_by_space	One or more integers separated by semicolons; set to 0 if no
2528		space separates the "int_curr_symbol" from the value for a
2529		nonnegative formatted monetary quantity, set to 1 if a space
2530		separates the symbol from the value, and set to 2 if a space
2531		separates the symbol and the sign string, if adjacent. If not
2532		specified, the value of "p_sep_by_space" is taken.
2533	int_n_cs_precedes	One or more integers separated by semicolons; set to 1 if the
2534		"int_curr_symbol" precedes the value for a negative
2535		formatted monetary quantity, and set to 0 if the symbol
2536		succeeds the value. If not specified, the value of
2537		"n_cs_precedes" is taken.
2538	int_n_sep_by_space	One or more integers separated by semicolons; set to 0 if no
2539		space separates the "int_curr_symbol" from the value for a
2540		negative formatted monetary quantity, set to 1 if a space
2541		separates the symbol from the value, and set to 2 if a space
2542		separates the symbol and the sign string, if adjacent. If not
2543		specified, the value of "n_sep_by_space" is taken.
2544	int_p_sign_posn	One or more integers separated by semicolons, set to a value
2545		indicating the positioning of the "positive_sign" for a
2546		nonnegative formatted monetary quantity using the

2547		"int_curr_symbol". The following integer values are defined:
2548		
2549	0	Parentheses enclose the quantity and the
2550		"int_curr_symbol".
2551	1	The sign string precedes the quantity and the
2552		"int_curr_symbol".
2553	2	The sign string succeeds the quantity and the
2554		"int_curr_symbol".
2555	3	The sign string immediately precedes the
2556		"int_curr_symbol".
2557	4	The sign string immediately succeeds the
2558		"int_curr_symbol".
2559		If no "int_p_sign_posn" is present the value of the
2560		"p_sign_posn" is taken.
2561		
2562	int_n_sign_posn	One or more integers separated by semicolons, set to a value
2563		indicating the positioning of the "negative_sign" for a
2564		negative formatted monetary quantity using the
2565		"int_curr_symbol". The following integer values are defined:
2566		
2567	0	Parentheses enclose the quantity and the
2568		"int_curr_symbol".
2569	1	The sign string precedes the quantity and the
2570		"int_curr_symbol".
2571	2	The sign string succeeds the quantity and the
2572		"int_curr_symbol".
2573	3	The sign string immediately precedes the
2574		"int_curr_symbol".
2575	4	The sign string immediately succeeds the
2576		"int_curr_symbol".
2577		If no "int_n_sign_posn" is present the value of the
2578		"n_sign_posn" is taken.
2579		

The "i18n" FDCC-set is defined as follows for the LC_MONETARY category.

```

2580 LC_MONETARY
2581 % This is the 14652 i18n fdcc-set definition for
2582 % the LC_MONETARY category.
2583 %
2584 int_curr_symbol      ""
2585 currency_symbol     ""
2586 mon_decimal_point   "<U002C>"
2587 mon_thousands_sep  ""
2588 mon_grouping        -1
2589 positive_sign       ""
2590 negative_sign       "<U002E>"
2591 int_frac_digits     -1
2592 frac_digits         -1
2593 p_cs_precedes       -1
2594 p_sep_by_space      -1
2595 n_cs_precedes       -1
2596 n_sep_by_space      -1
2597 p_sign_posn         -1
2598 n_sign_posn         -1
2599 %
2600 END LC_MONETARY
2601
2602
2603
2604
```

2605 4.6 LC_NUMERIC

2606

2607 The LC_NUMERIC category defines the rules and symbols that are used to format
2608 nonmonetary numeric information. The operands are strings. For some keywords, the
2609 strings only can contain integers. Keywords that are not provided, string values set to the
2610 empty string (""), or integer keywords set to -1, are used to indicate that the value is
2611 unspecified. The following keywords are defined:

2612

2613 **copy** Specify the name of an existing FDCC-set to be used as the
2614 source for the definition of this category. If this keyword is
2615 specified, no other keyword is specified.

2616 **decimal_point** The operand is a string containing the symbol that is used as the
2617 decimal delimiter in numeric, nonmonetary formatted quantities.
2618 This keyword cannot be omitted and cannot be set to the empty
2619 string. In contexts where other standards limit the decimal point
2620 to a single byte, the result of specifying a multibyte operand is
2621 unspecified.

2622 **thousands_sep** The operand is a string containing the symbol that is used as a
2623 separator for groups of digits to the left of the decimal delimiter
2624 in numeric, nonmonetary formatted monetary quantities. In
2625 contexts where other standards limit the "thousands_sep" to a
2626 single byte, the result of specifying a multibyte operand is
2627 unspecified.

2628 **grouping** Define the size of each group of digits in formatted non-
2629 monetary quantities. The operand is a sequence of integers
2630 separated by semicolons. Each integer specifies the number of
2631 digits in each group, with the initial integer defining the size of
2632 the group immediately preceding the decimal delimiter, and the
2633 following integers defining the preceding groups. If the last
2634 integer is not -1, then the size of the previous group (if any) is
2635 repeatedly used for the remainder of the digits. If the last integer
2636 is -1, then no further grouping is performed.

2637

2638 The "i18n" FDCC-set is for the LC_NUMERIC category:

2639

```
2640 LC_NUMERIC
2641 % This is the 14652 i18n fdcc-set definition for
2642 % the LC_NUMERIC category.
2643 %
2644 decimal_point    "<U002C>"
2645 thousands_sep    ""
2646 grouping         -1
2647 %
2648 END LC_NUMERIC
```

2649

2650

2651 4.7 LC_TIME (controversial)

2652

2653 The LC_TIME category defines the rules and symbols that are used to format date and
2654 time information.

2655

2656 Note: ISO 8601 allows different formats for dates, one form is YYYY-MM-DD, another is
2657 YYYYMMDD. Each clause in this specification specifies which specific format of ISO
2658 8601 that is used there.

2659

2660	The following keywords are defined:	
2661		
2662	copy	Specify the name of an existing FDCC-set to be used as the source for the definition of this category. If this keyword is specified, no other keyword is specified.
2663		
2664		
2665	abday	Define the abbreviated weekday names for calendar systems with weeks of constant length, to be referenced by the %a field descriptor. The length of the week and a Gregorian date for the first weekday is defined by the "week" keyword. The operand consists of semicolon-separated strings. The first string is the abbreviated name of the day corresponding to the first day of the week (default Sunday), the second the abbreviated name of the day corresponding to the second day of the week (default Monday), and so on.
2666		
2667		
2668		
2669		
2670		
2671		
2672		
2673	day	Define the full weekday names for calendar systems with weeks of constant length, to be referenced by the %A field descriptor. The length of the week and a Gregorian date for the first weekday is defined by the "week" keyword. The operand consists of semicolon-separated strings. The first string is the full name of the day corresponding to the first day of the week (default Sunday), the second the full name of the day corresponding to the second day of the week (default Monday), and so on.
2674		
2675		
2676		
2677		
2678		
2679		
2680		
2681	week	Is used to define the number of days in a week, and which weekday is the first weekday (the first weekday has the value 1), and which week is to be considered the first in a year. The first operand is an integer specifying the number of days in the week. The second operand is an integer specifying the Gregorian date in the format YYYYMMDD, and it specifies a day that is a first weekday (all other first weekdays may then be calculated by adding or subtracting a whole multiplum of the number of days in the week as specified with the first operand). The third operand is an integer specifying the weekday number to be contained in the first week of the year. The third operand may also be understood as the number of days required in a week for it to be considered the first week of the year. If the keyword is not specified the values are taken as 7, 19971130 (a Sunday), and 7 (Saturday), respectively. ISO 8601 conforming applications should use the values 7, 19971201 (a Monday), and 4 (Thursday), respectively. This keyword is optional.
2682		
2683		
2684		
2685		
2686		
2687		
2688		
2689		
2690		
2691		
2692		
2693		
2694		
2695		
2696		
2697	abmon	Define the abbreviated month names, to be referenced by the %b field descriptor. The operand consists of twelve or thirteen semicolon-separated strings. The first string is the abbreviated name of the first month of the year (January), the second the abbreviated name of the second month, and so on.
2698		
2699		
2700		
2701		
2702	mon	Define the full month names, to be referenced by the %B field descriptor. The operand consists of twelve or thirteen semicolon-separated strings. The first string is the full name of the first month of the year (January), the second the full name of the second month, and so on.
2703		
2704		
2705		
2706		
2707	d_t_fmt	Define the appropriate date and time representation, to be referenced by the %c field descriptor. The operand consists of a string, and can contain any combination of characters and field descriptors. In addition, the string can contain field descriptors defined in Table 3.
2708		
2709		
2710		
2711	d_fmt	Define the appropriate date representation, to be referenced by the

2712		%x field descriptor. The operand consists of a string, and can contain
2713		any combination of characters and field descriptors. In addition, the
2714		string can contain field descriptors defined in Table 3.
2715	t_fmt	Define the appropriate time representation, to be referenced by the
2716		%X field descriptor. The operand consists of a string, and can
2717		contain any combination of characters and field descriptors. In
2718		addition, the string can contain field descriptors defined in Table 3.
2719	am_pm	Define the appropriate representation of the ante meridiem and post
2720		meridiem strings, to be referenced by the %p field descriptor. The
2721		operand consists of two strings, separated by a semicolon. The first
2722		string represents the antemeridiem designation, the last string the
2723		postmeridiem designation. The keyword is optional. If unspecified,
2724		the %p field descriptor refers to the empty string.
2725	t_fmt_ampm	Define the appropriate time representation in the 12-hour clock
2726		format with "am_pm", to be referenced by the %r field descriptor.
2727		The operand consists of a string and can contain any combination of
2728		characters and field descriptors. If the string is empty, the 12-hour
2729		format is not supported in the FDCC-set.

The following keywords are all optional

2731		
2732		
2733	era	Define how years are counted and displayed for each era in a locale.
2734		The operand shall consist of semicolon-separated strings. Each string
2735		shall be an era description segment with the format:
2736		direction:offset:start_date:end_date:era_name:era_format
2737		according to the definitions below. There can be as many era
2738		description segments as are necessary to describe the different eras.
2739		NOTE: The start of an era might not be the earliest point in the
2740		era - it may be the AD 1, and increases with earlier time.
2741	direction	Either a '+' or a '-' character. The '+' character shall
2742		indicate that years closer to the start_date have lower
2743		numbers than those closer to the end_date. The '-'
2744		character shall indicate that years closer to the start_date
2745		have higher numbers than those closer to the end_date.
2746	offset	The number of the year closest to the start_date in the
2747		era, corresponding to the %Ey conversion specification
2748	start_date	A date in the format YYYYMMDD, where YYYY,
2749		MM, and DD are the year, month, and day numbers
2750		respectively according to ISO 8601 of the start of the
2751		era. Years prior to AD 1 shall be represented as
2752		negative numbers.
2753	end_date	The ending date of the era, in the same format as the
2754		start_date, or one of the two special values "-*" or "+*".
2755		The value "-*" shall indicate that the ending date is the
2756		beginning of time. The value "+*" shall indicate that the
2757		ending date is the end of time.
2758	era_name	A string representing the name of the era, corresponding
2759		to the %EC conversion specification.
2760	era_format	A string for formatting the year in the era,
2761		corresponding to the %EY conversion specification.
2762	era_year	Define the format of the year in alternate Era format, corresponding
2763		to the %EY field descriptor.

2764	era_d_t_fmt	Define the format of the date and time in alternate Era notation, corresponding to the %Ec field descriptor.
2765		
2766	era_d_fmt	Define the format of the date in alternate Era notation, corresponding to the %Ex field descriptor.
2767		
2768	era_t_fmt	Define the format of the time in alternate Era notation, corresponding to the %EX field descriptor.
2769		
2770	alt_digits	Define alternate symbols for digits, corresponding to the %O field descriptor modifier. The operand consists of semicolon-separated strings. The first string is the alternate symbol corresponding with zero, the second string the symbol corresponding with one, and so on. Up to 100 alternate symbol strings can be specified. The %O modifier indicates that the string corresponding to the value specified via the field descriptor is used instead of the value.
2771		
2772		
2773		
2774		
2775		
2776		
2777	first_weekday	Define the first day to be displayed, for example in a calendar display utility. The operand is an integer specifying the day number (1 = first) according to the information specified with the "day" keyword. The keyword may be omitted, and then the value 1 is taken, corresponding to Sunday for a week beginning Sunday, or to Monday for a week beginning Monday.
2778		
2779		
2780		
2781		
2782		
2783	first_workday	Define the first workday as an integer according to the day numbering specified with the "week" keyword.
2784		
2785	cal_direction	Define the direction of the display of dates, for example in a calendar display utility. The operand is an integer, and the following values are defined:
2786		
2787		
2788		1 left-right from top
2789		2 top-down from left
2790		3 right-left from top
2791		The keyword may be omitted, and then the value 1 is taken.
2792	timezone	Define one or more timezones, each defined by a string, and the strings separated by a <semicolon>. In the following the characters <, >, [and] are used as metacharacters. Only characters with a visible glyph from the portable character set may be used, except in the <std> and <dst> fields. The syntax of a string is:
2793		
2794		
2795		
2796		
2797		
2798		<std><offset><dst>[<offset>][,<rule>[,<rule>...]];
2799		
2800		where
2801		
2802		<std> and <dst> Indicates no less than three, nor more than 10
2803		characters that are the designation for the
2804		standard <std>, or Daylight Savings Time or
2805		summer time <dst> zone. Only <std> is
2806		required; if <dst> is missing, then Daylight
2807		Savings Time or summer time does not apply
2808		in this category. Upper- and lowercase letters
2809		are explicitly allowed. Any characters except a
2810		leading colon <:> or digits, the comma <,>, the
2811		minus <->, the plus <+>, and the null character
2812		are permitted to appear in these fields, but their
2813		meaning is unspecified.
2814		
2815		<offset> Indicates the value one must add to the local

2816		time to arrive at the Coordinated Universal
2817		Time. The <offset> has the form:
2818		
2819		hh[:mm[:ss]]
2820		
2821		The minutes (mm) and seconds (ss) are
2822		optional. The hour (hh) is required and may be
2823		a single digit. The <offset> following <std> is
2824		required. If no <offset> follows <dst>, summer
2825		time is assumed to be one hour ahead of
2826		standard time. One or more digits may be used;
2827		the value is always interpreted as a decimal
2828		number. The hour is between zero and 24, and
2829		the minutes (and seconds) - if present - is
2830		between zero and 59. If preceded by a "-", the
2831		time zone is east of the Prime Meridian;
2832		otherwise it is west of (which may be indicated
2833		by an optional preceding "+").
2834	<rule>	A specification for Daylight Savings Time
2835		changes that indicates when to change to and
2836		back from summer time. The <rule> has the
2837		form:
2838		<date>[/<time>/<year>],<date>[/<time>
2839		>/<year>]
2840		where the first <date> describes when the
2841		change from standard time to summer time
2842		occurs, and the second <date> describes when
2843		the change back happens. Each <time> field
2844		describes when, in current local time, the
2845		change to the other time is made. The first
2846		<year> field defines the beginning of the
2847		validity of this rule, and the second <year>
2848		field defines the end of the validity of the rule.
2849		A number of rules may be given.
2850		
2851		The format of <date> is one of the following:
2852		
2853	J<n>	The Julian day <n> (1 <= n
2854		<= 365) Leap years are not
2855		counted. That is, in all years -
2856		including leap years -
2857		February 28 is day 59 and
2858		March 1 is day 60. It is
2859		impossible to explicitly refer
2860		to the occasional February 29.
2861	<n>	The zero-based Julian day (0
2862		<= n <= 365). Leap years are
2863		counted and it is possible to
2864		refer to February 29.
2865	M<m>.<n>.<d>	
2866		the <d>th day (0 <= d <= 7)
2867		of week <n> of month <m> (1

2868 <= n <= 5, 1 <= m <= 12,
 2869 where week 5 means "the last
 2870 <d> day in month <m>"
 2871 which may occur in either the
 2872 fourth or fifth week). Week 1
 2873 is the first week in which the
 2874 <d>th day occurs. Day zero
 2875 and day seven is Sunday.
 2876

2877 The <time> has the same format as <offset>
 2878 except that no leading sign ("- or "+) is
 2879 allowed. The default, if <time> is not given, is
 2880 "02:00:00".
 2881

2882 The <year> has the format YYYY.
 2883

2884 NOTE: This way of specifying the timezone is compatible with the
 2885 format for the environment variable TZ described in Section 8.1.1 of
 2886 POSIX.1.
 2887

4.7.1 Date Field Descriptors

2888
 2889 The LC_TIME category defines the interpretation of a number of field descriptors. The
 2890 field descriptors are also available in the definitions with the following LC_TIME
 2891 keywords: "d_t_fmt", "d_fmt", "t_fmt", "t_fmt_ampm", "era", "era_d_t_fmt", "era_d_fmt",
 2892 and "era_t_fmt". A field descriptor may not be used with the LC_TIME keywords defining
 2893 it.
 2894

2895
 2896 **Table 3: Field descriptors for the date field**
 2897

2898	%a	FDCC-set's abbreviated weekday name.
2899	%A	FDCC-set's full weekday name.
2900	%b	FDCC-set's abbreviated month name.
2901	%B	FDCC-set's full month name.
2902	%c	FDCC-set's appropriate date and time representation.
2903	%C	Century (a year divided by 100 and truncated to integer) as decimal number (00-99).
2904	%d	Day of the month as a decimal number (01-31).
2905	%D	Date in the format mm/dd/yy.
2906	%e	Day of the month as a decimal number (1-31 in at two-digit field with leading <space> fill).
2907	%F	The date in the format YYYY-MM-DD (An ISO 8601 format).
2908	%g	Week-based year within century, as a decimal number (00-99).
2909	%G	Week-based year with century, as a decimal number (for example 1997).
2910	%h	A synonym for %b.
2911	%H	Hour (24-hour clock), as a decimal number (00-23).
2912	%I	Hour (12-hour clock), as a decimal number (01-12).
2913	%j	Day of the year, as a decimal number (001-366).
2914	%m	Month, as a decimal number (01-13).
2915	%M	Minute, as a decimal number (00-59).
2916	%n	A <newline> character.
2917	%p	FDCC-set's equivalent of either AM or PM.
2918		
2919		

2920	%r	12-hour clock time (01-12), using the AM/PM notation.
2921	%R	24-hour clock time, in the format "%H:%M".
2922	%S	Seconds, as a decimal number (00-61).
2923	%t	A <tab> character.
2924	%T	24-hour clock time, in the format HH:MM:SS.
2925	%u	Weekday, as a decimal number (1(Monday)-7).
2926	%U	Week number of the year (Sunday as the first day of the week) as a decimal number (00-53). All days in a new year preceding the first Sunday are considered to be in week 0.
2927		
2928		
2929	%v	Week number of the year, as a decimal number with two digits including a possible leading zero, according to "week" keyword.
2930		
2931	%V	Week of the year (Monday as the first day of the week), as a decimal number (01-53). The method for determining the week number is as specified by ISO 8601.
2932		
2933		
2934	%w	Weekday, as a decimal number (0(Sunday)-6).
2935	%W	Week number of the year (Monday as the first day of the week), as a decimal number (00-53). All days in a new year preceding the first Monday are considered to be in week 0.
2936		
2937		
2938	%x	FDCC-set's appropriate date representation.
2939	%X	FDCC-set's appropriate time representation.
2940	%y	Year within century (00-99).
2941	%Y	Year with century, as a decimal number.
2942	%z	The offset from UTC in the ISO 8601 format "-0430" (meaning 4 hours 30 minutes behind UTC, west of Greenwich), or by no characters if no time zone is determinable.
2943		
2944		
2945	%Z	Time-zone name, or no characters if no time zone is determinable.
2946	%%	A <percent-sign> character.

2947
2948 NOTE: %g, %G and %V give values according to the ISO 8601 week-based year. In this system, weeks
2949 begin on a Monday and week 1 of the year is the week that includes 4th January, which is also the week
2950 that includes the first Thursday of the year, and is also the first week that contains at least four days in
2951 the year. If the first Monday of the year is the 2nd, 3rd or 4th, the preceding days are part of the last
2952 week of the preceding year; thus, for Saturday 2nd January 1999, %G is replaced by 1998 and %V is
2953 replaced by 53. If the 29th, 30th or 31st December is a Monday, it and any following days are part of
2954 week 1 of the following year. Thus, for Tuesday 30th December 1997, %G is replaced by 1998 and %V
2955 is replaced by 1.

2956

2957 4.7.2 Modified Field Descriptors

2958

2959 Some field descriptors can be modified by the E and O modifier characters to indicate a
2960 different format or specification as specified in the LC_TIME FDCC-set description. If the
2961 corresponding keyword (see "era", "era_year", "era_d_t_fmt", "era_d_fmt", "era_t_fmt"
2962 and "alt_digits") is not specified for the current FDCC-set, the unmodified field descriptor
2963 value is used.

2964

2965	%Ec	FDCC-set's alternate date and time representation.
2966	%EC	The name of the base year (period) in the FDCC-set's alternate representation.
2967		
2968	%Ex	FDCC-set's alternate date representation.
2969	%EX	FDCC-set's alternate time representation.
2970	%Ey	Offset from %EC (year only) in the FDCC-set's alternate representation.
2971	%EY	Full alternate year representation.
2972	%Od	Day of month using the FDCC-set's alternate numeric symbols.

2973	%Oe	Day of month using the FDCC-set's alternate numeric symbols.
2974	%Of	Weekday as a decimal number according to alt_day (1 is first day).
2975	%OH	Hour (24-hour clock) using the FDCC-set's alternate numeric symbols.
2976	%OI	Hour (12-hour clock) using the FDCC-set's alternate numeric symbols.
2977	%Om	Month using the FDCC-set's alternate numeric symbols.
2978	%OM	Minutes using the FDCC-set's alternate numeric symbols.
2979	%OS	Seconds using the FDCC-set's alternate numeric symbols.
2980	%Ou	Weekday as a number in the alternate representation of the FDCC-set
2981		(Monday=1).
2982	%OU	Week number of the year (Sunday as the first day of the week) using the
2983		FDCC-set's alternate numeric symbols.
2984	%OV	Week number of the year (Monday as the first day of the
2985		week, ISO 8601 rules) using the alternate numeric symbols
2986		of the FDCC-set.
2987	%Ow	Weekday as number in the FDCC-set's alternate representation
2988		(Sunday=0).
2989	%OW	Week number of the year (Monday as the first day of the week) using the
2990		FDCC-set's alternate numeric symbols.
2991	%Oy	Year (offset from %C) in alternate representation.

4.7.3 "i18n" LC_TIME category

The "i18n" LC_TIME category is (following ISO 8601):

```

2997 LC_TIME
2998 % This is the ISO/IEC TR 14652 "i18n" definition for
2999 % the LC_TIME category.
3000 %
3001 % Weekday and week numbering according to ISO 8601
3002 abday  "<U0031>"; "<U0032>"; "<U0033>"; "<U0034>"; /
3003        "<U0035>"; "<U0036>"; "<U0037>"
3004 day    "<U0031>"; "<U0032>"; "<U0033>"; "<U0034>"; /
3005        "<U0035>"; "<U0036>"; "<U0037>"
3006 week  7;19971201;4
3007 abmon  "<U0030><U0031>"; "<U0030><U0032>"; "<U0030><U0033>"; /
3008        "<U0030><U0034>"; "<U0030><U0035>"; "<U0030><U0036>"; /
3009        "<U0030><U0037>"; "<U0030><U0038>"; "<U0030><U0039>"; /
3010        "<U0031><U0030>"; "<U0031><U0031>"; "<U0031><U0032>"
3011 mon   "<U0030><U0031>"; "<U0030><U0032>"; "<U0030><U0033>"; /
3012        "<U0030><U0034>"; "<U0030><U0035>"; "<U0030><U0036>"; /
3013        "<U0030><U0037>"; "<U0030><U0038>"; "<U0030><U0039>"; /
3014        "<U0031><U0030>"; "<U0031><U0031>"; "<U0031><U0032>"
3015 am_pm  "";"
3016 % Date formats following ISO 8601
3017 % Appropriate date and time representation (%c)
3018 % "%F %T"
3019 d_t_fmt "<U0025><U0046><U0020><U0025><U0054>"
3020 %
3021 % Appropriate date representation (%x) "%F"
3022 d_fmt  "<U0025><U0046>"
3023 %
3024 % Appropriate time representation (%X) "%T"
3025 t_fmt  "<U0025><U0054>"
3026 t_fmt_ampm ""
3027 %
3028 END LC_TIME
3029

```

4.8 LC_MESSAGES

The LC_MESSAGES category defines the format and values for affirmative and negative responses. The operands are strings or extended regular expressions to specify which response strings that should be considered matches; see ISO/IEC 9945-2:1993 clause 2.8.4 for a definition of extended regular expressions. The following keywords are defined:

3036	copy	Specify the name of an existing FDCC-set to be used as the source for the definition of this category. If this keyword is specified, no other keyword is specified.
3037		
3038		
3039	yesexpr	The operand consists of an extended regular expression that describes the acceptable affirmative response to a question expecting an affirmative or negative response.
3040		
3041		
3042	noexpr	The operand consists of an extended regular expression that describes the acceptable negative response to a question expecting an affirmative or negative response.
3043		
3044		
3045		

The "i18n" LC_MESSAGES category is:

```

3046 LC_MESSAGES
3047 % This is the ISO/IEC 14652 "i18n" definition for
3048 % the LC_MESSAGES category.
3049 %
3050 yesexpr "<U005B><U002B><U0031><U005D>"
3051 noexpr  "<U005B><U002D><U0030><U005D>"
3052 END LC_MESSAGES
3053

```

Note: This uses regular expression syntax with brackets ([]) to for example specify that both <+> and <1> is allowed as an affirmative answer.

4.9 LC_XLITERATE (controversial)

The LC_XLITERATE category defines formats to transform strings, by transforming substrings in the source to substrings in the target string. The target is the culture of the FDCC-set in question. The capabilities can be used for for simple transliteration or fallback based on substring substitution, while more advanced transliteration schemes, for example based on pattern matching, sound equivalences, or using a database, is either cumbersome to specify, or not addressed. The transliteration may for example be from the Cyrillic script to the Latin script.

Transliteration of an incoming character string to a character string in a FDCC-set can be specified with the following transliteration keywords and transliteration statements.

3072	copy	Specify the name of an existing FDCC-set to be used as the source for the definition of this category. If this keyword is specified, no other keyword is specified.
3073		
3074		
3075	include	The name of the FDCC-set in text form to transliterate from, and the repertoiremap for the FDCC-set to be used for the definition of the transliteration statements. Other transliteration statements may follow to replace specification of the copied FDCC-set. This keyword is optional.
3076		
3077		
3078		
3079		
3080	default_missing	defines a string of one or more characters to be put in the output string if no transliteration statement can be applied to a input <transliteration-source>. This keyword is optional.
3081		
3082		
3083	translit_ignore	defines a set of characters, separated by semicolons, that are to be ignored in the incoming character string, that is, each of the occurrences of such characters is treated as the empty string. The characters may use the notations defined in 4.3 for lists of characters. This keyword is optional.
3084		
3085		
3086		
3087		
3088	redefine	This keyword introduces a list of transliteration statements where each of the <transliteration_source> strings have been defined previously in the specification, and the new
3089		
3090		

3091 transliteration statements then replaces the old transliteration
 3092 statements for the <transliteration_source> strings specified.
 3093 This keyword is optional.
 3094

3095 **4.9.1 Transliteration statements**

3096
 3097 The syntax for a transliteration statement is:

```
3098     "%s %s;%s;...;%s\n",<transliteration_source>,<transliteration_string>,...
```

3100
 3101 Each <transliteration_source> consists of one or more characters (in any of the forms
 3102 defined in 4.1.1). The <transliteration_source> that is the longest in terms of number of
 3103 characters that match the input string is the one selected for transliteration.
 3104

3105 If a transliteration statement contains more than one <transliteration_string>, the order that
 3106 each <transliteration_string> occurs in the transliteration statement defines the precedence
 3107 order for choosing a particular <transliteration_string> to substitute for the
 3108 <transliteration_source>. When a process makes use of a transliteration statement to
 3109 transliterate text, and that transliteration statement contains more than one
 3110 <transliteration_string>, that process chooses the first <transliteration_string>, in the
 3111 defined precedence order, that satisfies the requirements of the transliteration.
 3112

3113 Note: the exact definition of the concept of satisfying the requirements of the transliteration is outside the
 3114 context of this Technical Report. If, for example, a transliteration involves a change in the coded character set
 3115 of a string, a <transliteration_string> must be chosen, all of whose elements are members of that coded
 3116 character set. In order to determine this, it would be expected that a repertoire describing which characters are
 3117 to be present in the resulting transformed string be available to the transliteration API. Also, a transliteration
 3118 may involve requirements such as that string length not change under transliteration. Such requirements may
 3119 also affect the choice among alternative <transliteration_string> values.
 3120

3121 If more than one transliteration statement is given for a given <transliteration_source> this
 3122 is an error, and duplicate transliteration statements are ignored. Tailoring of transliteration
 3123 statements may be done via the "redefine" keyword.
 3124

3125 **4.9.2 "include" keyword**

3126
 3127 The "include" keyword specifies a set of transliteration statements in text form to be
 3128 included in the applied transliteration. The syntax of the "include" statement is:

```
3129     "include %s;%s\n", <FDCC-set>, <repertoiremap>
```

3130
 3131 <FDCC-set> is a string identifying the FDCC-set to be included from.
 3132

3133
 3134 <repertoiremap> is a string identifying the repertoiremap used in the FDCC-set being
 3135 included, and is used to map character specifications from the specified FDCC-set into the
 3136 current FDCC-set.
 3137

3138 **4.9.3 Example of use of transliteration**

```
3139     LC_XLITERATE  

  3140     include "de_DE"; "de_remap"  

  3141     default_missing <?>  

  3142     translit_ignore <U3200>..
  3143     <ae>          <a:>; <e*>; "<a><e>" ; "<e>"  

  3144     <s>          <s*>; <s=>  

  3145     "<K><O>"     <KO>  

  3146     END LC_XLITERATE
```


The "LC_XLITERATE" statement introduces the transliteration category.

The "include" keyword specifies that the FDCC-set "de_DE" is copied and that the repertoiremap "de_repmap" is used to define the symbolic character names in the FDCC-set "de_DE".

The "default_missing" keyword introduces the character sequence "<?>" as the string to transform into for input characters that cannot be transformed into other strings, because no transliteration statement is applicable to the character.

The "translit_ignore" keyword specifies that a set of Ideographic characters, Hangul, East Asian symbols and the private use area etc. (the range <U3200>..<>UFAFF>) is ignored for the transliteration.

The next 3 lines are transliteration statements.

The first transliteration statement defines a number of transliterations for the LATIN LETTER AE, including into LATIN LETTER A WITH DIAERESIS, GREEK LETTER EPSILON, the two Latin letters A and E, and finally the LATIN LETTER E.

The second transliteration statement defines transliteration of the LATIN LETTER S into GREEK LETTER SIGMA, and CYRILLIC LETTER ES.

The third transliteration statement transliterates the two Latin letters K and O into the Japanese Hiragana character KO.

The transliteration category is terminated via the "END LC_XLITERATE" statement in the above example.

There is no "i18n" entry for the LC_XLITERATE category

4.10 LC_NAME

The LC_NAME category defines formats to be used in addressing a person, e.g. in a postal address or in a letter. The following keywords are defined:

- | | |
|------------------|---|
| copy | Specify the name of an existing FDCC-set to be used as the source for the definition of this category. If this keyword is specified, no other keyword is specified. |
| name_fmt | Define the appropriate representation of a person's name and title. The operand consists of a string, and can contain any combination of characters and field descriptors. In addition, the string can contain field descriptors defined below. |
| name_gen | The operand is a string defining a salutation valid for all persons. |
| name_miss | The operand is a string defining a salutation valid for unmarried females. |
| name_mr | The operand is a string defining a salutation valid for males. |
| name_mrs | The operand is a string defining a salutation valid for married females. |
| name_ms | The operand is a string defining a salutation valid for all females. |

NOTE: There are a number of variations for addressing a person among the cultures. Middle names are not used in many countries and even the family name is not used in some countries. In other countries there is extensive use of one or more middle names and corresponding initials. The specification below should be regarded as a starting point for this problem.

The LC_NAME category defines the interpretation of a number of field descriptors. The field descriptors are also available in the definitions with the following LC_NAME keywords: "name_fmt".

Field descriptors for the "name_fmt" keyword:

- | | |
|----|---|
| %f | Family names. |
| %F | Family names in uppercase. |
| %g | First given name. |
| %G | First given initial. |
| %l | First given name with latin letters. In some cultures, eg on Taiwan it is customary |

3209		to also have a first name written with Latin letters, although the rest of the name is
3210		written in another script.
3211	%o	Other shorter name, eg. "Bill".
3212	%m	Additional given names.
3213	%M	Initials for additional given names.
3214	%p	Profession.
3215	%s	Salutation, such as "Doctor"
3216	%S	Abbreviated salutation, such as "Mr." or "Dr."
3217	%d	Salutation, using the FDCC-sets conventions, with 1 for the name_gen, 2 for
3218		name_mr, 3 for name_mrs, 4 for name_miss, 5 for name_ms.
3219	%t	If the preceding field descriptor resulted in an empty string, then the empty string,
3220		else a <space>.

3221
3222 Each field descriptor may have an <R> after the <%> to specify that the information is
3223 taken from a Romanized version string of the entity. An initial is any string, normally
3224 consisting of one letter and a punctuation mark; the Dutch "IJ" is an example of a two
3225 character initial.

3226
3227 The "i18n" LC_NAME category is:
3228

```

3229 LC_NAME
3230 % This is the ISO/IEC TR 14652 "i18n" definition for
3231 % the LC_NAME category.
3232 name_fmt "<U0025><U0070><U0025><U0074><U0025><U0067><U0025><U0074>/
3233 <U0025><U006D><U0025><U0074><U0025><U0066>"
3234 % This corresponds to "%p%t%g%m%t%f" which is
3235 % Profession Primary Additional Family
3236 END LC_NAME

```

3237

3238 4.11 LC_ADDRESS

3239

3240 The LC_ADDRESS category defines formats to be used in specifying a location like a
3241 person's home or office, for use in a postal address or in a letter, and other items related
3242 to geography, including natural language. All keywords are strings and may contain non-
3243 digits, and all keywords are optional. The following keywords are recognized:

3244

3245	copy	Specify the name of an existing FDCC-set to be used as the source
3246		for the definition of this category. If this keyword is specified, no
3247		other keyword is specified.
3248	postal_fmt	Define the appropriate representation of a postal address such as
3249		street and city. The proper formatting of a person's name and title is
3250		done with the "name_fmt" keyword of the LC_NAME category. The
3251		operand consists of a string, and can contain any combination of
3252		characters and field descriptors. In addition, the string can contain
3253		field descriptors defined below.
3254	country_name	The operand is a string with the name of the country in the language
3255		of the FDCC-set.
3256	country_post	The operand is a string with the abbreviation of the country, used for
3257		postal addresses, for example by the CEPT-MAILCODE codes
3258		designating countries in Europe. Other abbreviation systems are also
3259		allowed, and there is no specific way to identify which abbreviation
3260		system is being used.
3261	lang_name	The operand is a string with the name of the language in the
3262		language of the FDCC-set.
3263	lang_ab2	The operand is a string with the two-letter abbreviation of the

3264 language, according to ISO 639.
 3265 **lang_ab3_term** The operand is a string with the three-letter abbreviation of the
 3266 language for terminology use, according to ISO 639-2.
 3267 **lang_ab3_lib** The operand is a string with the three-letter abbreviation of the
 3268 language for library use, according to ISO 639-2. If not specified, the
 3269 value of the "lang_ab3_term" keyword is taken.
 3270

3271 Note: The "lang_ab3_term" and "lang_ab3_lib" keywords will in most cases contain the
 3272 same value, but they may differ, e.g the values for the German language is "deu" and
 3273 "ger" respectively.
 3274

3275 The LC_ADDRESS category defines the interpretation of a number of field descriptors.
 3276 The field descriptors are also available in the definitions with the following
 3277 LC_ADDRESS keywords: "postal_fmt".
 3278

3279 Field descriptors for the "postal_fmt" keyword:
 3280

3281	%n	Person's name, possibly constructed with LC_NAME.
3282	%a	Care of person, or organization.
3283	%f	Firm name.
3284	%d	Department name.
3285	%b	Building name.
3286	%s	Street or block (eg. Japanese) name.
3287	%h	House number or designation.
3288	%N	Insert an <end-of-line> if the previous descriptor's value was not an 3289 empty string; otherwise ignore.
3290	%t	Insert a <space> if the previous descriptor's value was not an empty 3291 string; otherwise ignore.
3292	%r	Room number, door designation.
3293	%e	Floor number.
3294	%C	Country designation, from the <country_post> keyword.
3295	%l	Local township within town or city
3296	%z	Zip number, postal code.
3297	%T	Town, city.
3298	%S	State, province, or prefecture.
3299	%c	Country, as taken from data record.

3300
 3301 Each field descriptor may have an <R> after the <%> to specify that the information is
 3302 taken from a Romanized version string of the entity.
 3303

3304 NOTE: There are a number of variations for specifying a location among the cultures.
 3305 Some of the information, like the middle names, or even the family name, is not used
 3306 in some cultures. The specification here should be regarded as a starting point for this
 3307 problem.
 3308

3309
 3310 Examples:

3311 A specification for the USA could be:

3312
 3313
 3314 "%n%N%a%N%d%N%f%N%b%N%h %s%N%e %r%N%l%N%C-%z %T%, %S %z%N%c%N"

3315
 3316 Giving:

3317
 3318 Person's name

3319 C/o address
 3320 Department
 3321 Firm
 3322 Building
 3323 number street
 3324 floor room
 3325 Local Town
 3326 City, State Zip
 3327 Country

3328
 3329 An example for South Korea could be:

3330 "%S %T %l %s %h %N%f %d%N%b %e %r%N%n %a%N%z"

3331

3332 Giving:

3333 State City Town Street number

3334 Firm department

3335 Building floor room

3336 Person's name C/o address

3337 Zip

3338

3339

3340 The "i18n" LC_ADDRESS category is:

3341

3342 LC_ADDRESS

3343 % This is the ISO/IEC TR 14652 "i18n" definition for

3344 % the LC_ADDRESS category.

3345 %

3346 postal_fmt "<U0025><U006E><U0025><U004E>/

3347 <U0025><U0061><U0025><U004E><U0025><U0066><U0025><U004E>/

3348 <U0025><U0064><U0025><U004E><U0025><U0062><U0025><U004E><U0025><U0073>/

3349 <U0020><U0025><U0068><U0020><U0025><U0065><U0020><U0025><U0072>/

3350 <U0025><U004E><U0025><U006C><U0025><U004E><U0025><U0043><U002D>/

3351 <U0025><U007A><U0020><U0025><U0054><U0025><U004E>/

3352 <U0025><U0053><U0025><U004E><U0025><U0063><U0025><U004E>"

3353 %

3354 % "%n%N%a%N%f%N%d%N%b%N%s %h %e %r%N%l%N%C-%z %T%N%S%N%c%N" resulting in

3355 % Person's_Name

3356 % C/o_person_or_org

3357 % Firm

3358 % Department

3359 % Building_name

3360 % Street_or_block number floor room

3361 % Local_township

3362 % Country-Zip City

3363 % State_or_province

3364 % Country

3365 %

3366 END LC_ADDRESS

3367

3368

3369 4.12 LC_TELEPHONE

3370

3371 The LC_TELEPHONE category defines formats to be used with telephone services. All
 3372 keywords are optional. The strings are not restricted in what characters they can contain.
 3373 The following keywords are defined:

3374

3375 **copy** Specify the name of an existing FDCC-set to be used as the source
 3376 for the definition of this category. If this keyword is specified, no
 3377 other keyword is specified.

3378 **tel_int_fmt** Define the appropriate representation of a telephone number for
 3379 international use. The operand consists of a string, and can contain
 3380 any combination of characters and field descriptors. In addition, the
 3381 string can contain field descriptors defined below.

3382 **tel_dom_fmt** Define the appropriate representation of a telephone number for

3383 domestic use. The operand consists of a string, and can contain any
 3384 combination of characters and field descriptors. In addition, the string
 3385 can contain field descriptors defined below.
 3386 **int_select** The operand is a string with the digits used to call international
 3387 telephone numbers.
 3388 **int_prefix** The operand is a string with the prefix used from other countries to
 3389 call the area.
 3390

3391 The LC_TELEPHONE category defines the interpretation of a number of field descriptors.
 3392 The field descriptors are also available in the definitions with the following
 3393 LC_TELEPHONE keywords: "tel_int_fmt" and "tel_dom_fmt".
 3394

3395 %a area code without nationwide prefix (prefix is often <0>).
 3396 %A area code including nationwide prefix (prefix is often <0>).
 3397 %l local number (within area code).
 3398 %e extension (to local number)
 3399 %c country code
 3400 %C alternate carrier service code used for dialling abroad
 3401 %t Insert a <space> if the previous descriptor's value was not an empty
 3402 string; otherwise ignore.
 3403
 3404

3405 The "i18n" LC_TELEPHONE category is:
 3406

```

3407 LC_TELEPHONE
3408 % This is the ISO/IEC TR 14652 "i18n" definition for
3409 % the LC_TELEPHONE category.
3410 %
3411 %
3412 tel_int_fmt      "<U002B><U0025><U0063><U0020><U0025><U0061><U0025><U0074>/
3413 <U0025><U006C>"
3414 % "+%c %a%t%l" which is
3415 % +country area local
3416 END LC_TELEPHONE

```

3417

3418

3419 5. CHARMAP

3420

3421 A character set description may exist for each coded character set supported by the
 3422 implementation. This file is referred to elsewhere in this Technical Report as a charmap.
 3423

3424

3425 A conforming charmap to be used with a FDCC-set supports the portable character set
 3426 specified in Table 1.

3427

3428 Conforming charmaps specify certain character and character set attributes, as defined in
 3429 5.1.

3430

3431

3432 5.1 Character Set Description Text

3433

3434 The character set description text (charmap) describes the mapping between symbolic
 3435 character names and actual encoding of a coded character set. It is used to bind the
 3436 symbolic character names in a FDCC-set to an actual encoding, so an application can
 3437 process data in this encoding.
 3438

3439	The following declarations can precede the character definitions. Each consist of the	
3440	symbol shown in the following list, starting in column 1, including the surrounding	
3441	brackets, followed by one of more "blank"s, followed by the value to be assigned to the	
3442	symbol. If any of the declarations are included, they are specified in the order shown in	
3443	the following list:	
3444		
3445	<code_set_name>	The name of the coded character set for which the character set
3446		description text is defined. The characters of the name are taken
3447		from the set of characters with visible glyphs defined in Table 1.
3448		
3449	<mb_cur_max>	The maximum number of bytes in a multibyte character. This
3450		defaults to 1.
3451		
3452	<mb_cur_min>	An unsigned positive integer value that defines the minimum
3453		number of bytes in a character for the encoded character set. The
3454		value is less or equal to "mb_cur_max". If not specified, the
3455		minimum number is equal to "mb_cur_max".
3456		
3457	<escape_char>	The escape character used to indicate that the characters
3458		following is interpreted in a special way, as defined later in this
3459		subclause. This defaults to backslash (\). The character slash (/)
3460		is used in all the following text and examples, unless otherwise
3461		noted.
3462		
3463	<comment_char>	The character that when placed in column 1 of a charmap line,
3464		is used to indicate that the line is ignored. The default character
3465		is the number sign (#). The character percent-sign (%) is used in
3466		all the following text and examples, unless otherwise noted.
3467		
3468	<repertoiremap>	The name of the repertoiremap used to define the symbolic
3469		character names in the charmap. The characters of the name are
3470		taken from the set of characters with visible glyphs defined in
3471		Table 1.
3472		
3473	<escseq2022>	defines the escape sequences for ISO 2022 shifting for the coded
3474		character set defined by the charmap. The semicolon-separated
3475		operands are all strings with characters taken from the set of
3476		characters with visible glyphs defined in table 1. The first
3477		operand defines the g-set or c-set to be defined, and the
3478		following values are defined: c0, c1, g0, g1, g2, g3. The second
3479		operand defines what range of characters in the charmap is
3480		affected, and the values defined are: c0, c1, g0, g1. The third
3481		operand is the escape sequence that is defined.
3482		
3483		
3484	<addset>	the name of the charmap to be added to the current coded
3485		character set, and to be selected by the escape sequences defined
3486		by <escseq2022> of the added charmap.
3487		
3488	<include>	include the encoding of another charmap in the current charmap.
3489		The semicolon-separated operands are all strings with characters
3490		taken from the set of characters with visible glyphs defined in

3491 table 1. The first operand defines the g-set or c-set to be defined
 3492 in the current charmap, and the following values are defined: c0,
 3493 c1, g0, g1, g2, g3. The second operand defines a range of
 3494 characters in the referenced charmap, and the values defined are:
 3495 c0, c1, g0, g1. The third operand is the name of the charmap to
 3496 be included. The coded character sets are defined initially for the
 3497 encoding, and therefore do not need escape sequences for
 3498 identification. If two g0 sets are defined, the second is switched
 3499 to using the SHIFT OUT control character, while the first is
 3500 shifted to using the SHIFT IN control character.

3501
 3502 The character set mapping definitions are all the lines immediately following an identifier
 3503 line containing the string "CHARMAP" starting in column 1, and preceding a trailer line
 3504 containing the string "END CHARMAP" starting in column 1. Empty lines and lines
 3505 containing a <comment_char> in the first column are ignored. Each non-comment line of
 3506 the character set mapping definition (i.e., between the "CHARMAP" and "END
 3507 CHARMAP" lines of the text) is in one of the following syntaxes.

3508
 3509 "%s %s %s\n", <symbolic-name>,<encoding>,<comments>

3510
 3511 "%s...%s %s %s\n", <symbolic-name>,<symbolic-name>,<encoding>,<comments>

3512
 3513 "%s....%s %s %s\n", <symbolic-name>,<symbolic-name>,<encoding>,<comments>

3514
 3515 "%s..%s %s %s\n", <symbolic-name>,<symbolic-name>,<encoding>,<comments>

3516
 3517 In the first syntax, the line of the character set mapping definition starts with the symbolic
 3518 name, immediately preceded by a <less-than> character and immediately followed by a
 3519 <greater-than> character. Symbolic names only contain characters from the set shown
 3520 with a visible glyph in Table 1.

3521
 3522 The same symbolic name may occur several times, with different values. The first value is
 3523 the one used when generating an encoding, while the other values are accepted in
 3524 decoding. Symbolic names may be included to identify values that can overlap with each
 3525 other or with the values of the symbolic names shown in Table 1. It is possible to specify
 3526 symbolic names for which no encoding exists in the encoded character set, by not
 3527 specifying a value.

3528
 3529 In the second and third syntax (symbolic decimal ellipsis), the line in the character set
 3530 mapping defines a range of one or more symbolic names. The difference between the
 3531 second and the third syntax is the number of dots in the ellipsis: the second has 3 dots, the
 3532 third has 4 dots. In these forms the symbolic names consist of zero or more nonnumeric
 3533 characters from the set shown with visible glyphs in Table 1, followed by an integer
 3534 formed by one or more decimal digits. The characters preceding the integer are identical
 3535 in the two symbolic names, and the integer formed by the digits in the second symbolic
 3536 name are identical to or greater than the integer formed by the digits in the first name.
 3537 This is interpreted as a series of symbolic names formed from the common part and each
 3538 of the integers in decimal format between the first and the second integer, inclusive, and
 3539 with a length of the symbolic names generated that is equal to the length of the first (and
 3540 also the second) symbolic name. As an example, <j0101>...<j0104> is interpreted as the
 3541 symbolic names <j0101>, <j0102>, <j0103>, and <j0104>, in that order.
 3542

3543 Note: The rationale to allow both a 3-dot and a 4-dot symbol for symbolic decimal
 3544 ellipses is that in the POSIX standard the decimal symbolic ellipses was defined by a 3-
 3545 dot symbol for charmaps, while the 3-dot symbol was an absolute ellipses for POSIX
 3546 locales, and this Technical Report specifies a 4-dot symbol for the decimal symbolic
 3547 ellipses. The 3-dot symbolic decimal ellipses in charmaps is deprecated.
 3548

3549 In the fourth syntax (symbolic hexadecimal ellipsis, with two dots), the line in the
 3550 character set mapping defines a range of one or more symbolic names. In this form the
 3551 symbolic names consist of zero or more nonnumeric characters from the set shown with
 3552 visible glyphs in Table 1, followed by an integer formed by one or more hexadecimal
 3553 digits, using uppercase letters only for the range "A" to "F". The characters preceding the
 3554 hexadecimal integer are identical in the two symbolic names, and the integer formed by
 3555 the hexadecimal digits in the second symbolic name is identical to or greater than the
 3556 integer formed by the hexadecimal digits in the first name. This is interpreted as a series
 3557 of symbolic names formed from the common part and each of the integers in hexadecimal
 3558 format using uppercase letters only between the first and the second integer, inclusive, and
 3559 with a length of the symbolic names generated that is equal to the length of the first (and
 3560 also the second) symbolic name. As an example, <U010E>..<U0111> is interpreted as the
 3561 symbolic names <U010E>, <U010F>, <U0110>, and <U0111>, in that order.
 3562

3563 The encoding part is expressed as one (for single-byte values) or more concatenated
 3564 decimal, octal or hexadecimal constants (hexadecimal constants are recommended).
 3565 Decimal constants are represented by two or three decimal digits, preceded by the escape
 3566 character and the lowercase letter "d"; for example /d05, /d97, or /d143. Hexadecimal
 3567 constants are represented by two hexadecimal digits, preceded by the escape character and
 3568 the lowercase letter "x"; for example /x05, /x61, or /x8f. Octal constants are represented
 3569 by two or three octal digits, preceded by the escape character; for example /05, /141, or
 3570 /217. In a charmap, each constant should represent an 8 bit byte for portability reasons.
 3571 Applications supporting other byte sizes may allow constants to represent values larger
 3572 than those that can be represented in 8 bit bytes, and to allow additional digits in
 3573 constants. When constants are concatenated for multibyte character values, they may be of
 3574 different types, and interpreted in byte order from the first to the last with the least
 3575 significant byte of the multibyte character specified by the last byte. The manner in which
 3576 these constants are represented in the character stored in the system is application defined.
 3577 Omitting bytes from a multibyte character produces undefined results.
 3578

3579 In lines defining ranges of symbolic names, the encoded value is the value for the first
 3580 symbolic name in the range (the symbolic name preceding the ellipsis). Subsequent
 3581 symbolic names defined by the range have encoding values in increasing order. For
 3582 example the line

3583 <j0101>....<j0104> /d129/d254

3584 is interpreted as

3587
 3588 <j0101> /d129/d254
 3589 <j0102> /d129/d255
 3590 <j0103> /d130/d000
 3591 <j0104> /d130/d001
 3592

3593 The comments parameter is optional.
 3594

3595 Example of using ISO 2022 techniques:

3596
3597 The following example defines two coded character sets, a 7-bit and a 14-bit. They are then merged into one
3598 encoding. It is an example on how encodings used in Eastern Asia could be specified.
3599

3600 The 7-bit charmap

```
3601 <escape_char> /
3602 <comment_char> %
3603 % The 7-bit charmap defines both control and graphic characters
3604 <code_set_name> "eastern7bit"
3605 <escseq2022> "c0";"c0", "/x21/x40"
3606 <escseq2022> "g0";"g0", "/x28/x48"
3607 <escseq2022> "g1";"g0", "/x29/x48"
3608 <escseq2022> "g2";"g0", "/x2A/x48"
3609 <escseq2022> "g3";"g0", "/x2B/x48"
3610
3611 CHARMAP
3612 <tab> /x08
3613 <newline> /x0D
3614 <a> /x61
3615 % more character encodings to be defined here
3616 END CHARMAP
3617
```

3618

3619

3620 The 14-bit charmap

```
3621 <escape_char> /
3622 <comment_char> %
3623 <code_set_name> "eastern14bit"
3624 <mb_cur_max> 2
3625 <esqseq2022> "g0";"g0";"/x24/x40"
3626 <esqseq2022> "g1";"g0";"/x24/x29/x40"
3627 <esqseq2022> "g2";"g0";"/x24/x2A/x40"
3628 <esqseq2022> "g3";"g0";"/x24/x2B/x40"
3629
3630 CHARMAP
3631 <U0165> /d036/d055 % the character codes are only examples
3632 <U0166> /d036/d056
3633 % more character encodings to be defined here
3634 END CHARMAP
3635
```

3636

3637

3638 The merged encoding

```
3639 <escape_char> /
3640 <comment_char> %
3641 <code_set_name> "shift-eastern"
3642 <mb_cur_max> 2
3643 <mb_cur_min> 1
3644 <include> "c0";"c0";"eastern7bit"
3645 <include> "g0";"g0";"eastern7bit"
3646 <include> "g1";"g0";"eastern14bit"
3647 % This defines the g0 values of "eastern14bit" (without the 8th
3648 % bit set) to be the g1 in this encoding (with the 8th bit set).
3649 %
3650 % So the bytes without the 8th bit set is from the "eastern7bit"
3651 % coded character set, while bytes with the 8th bit set are from
3652 % the 14-bit set.
3653
```

3654

3655 Another merged encoding using the same charmaps:

```
3656 <escape_char> /
3657 <comment_char> %
3658 <code_set_name> "EUC-eastern"
3659 <mb_cur_max> 2
3660 <mb_cur_min> 1
3661 <include> "c0";"c0";"eastern7bit"
3662 <include> "g0";"g0";"eastern7bit"
3663 <include> "g0";"g0";"eastern14bit"
3664 % As there are two "g0" sets defined, the first referenced is the
```

3665 % initial g0 set, while the second can be shifted to via the SHIFT OUT
 3666 % control character. The first can then be shifted to by the SHIFT IN
 3667 % control character.

3670 **WIDTH section**

3671
 3672 After the "END CHARMAP" statement the following declarations may follow. Each
 3673 consists of the keyword shown in the following list, starting in column 1, followed by the
 3674 value(s) to be associated to the keyword, as defined below.

3675
 3676 **WIDTH** An unsigned positive integer value defining the column width for the characters
 3677 in the coded character set. Coded character values are defined using symbolic character
 3678 names followed by a column width value. Defining a character with more than one
 3679 **WIDTH** produces undefined results. The **END WIDTH** keyword is used to terminate the
 3680 **WIDTH** definitions.

3681
 3682 **WIDTH_DEFAULT** An unsigned positive integer value defining the column width for any
 3683 character not listed by one of the **WIDTH** keywords. If no **WIDTH_DEFAULT** keyword
 3684 is included in the charmap, the default character width is 1.

3685
 3686 Example:

3687
 3688 After the "END CHARMAP" statement, a syntax for width definition would be:

```
3689 WIDTH
3690 <A> 1
3691 <B> 1
3692 <j0101>...<j0195> 2
3693 <U4E00>..<U9FA5> 2
3694 END WIDTH
3695 WIDTH_DEFAULT 1
```

3696
 3697 In this example, the code point values represented by <A> and are assigned a width of 1. The code
 3698 point values <j0101>...<j0195> (decimal ellipses) and <U4E00>..<U9FA5> are assigned a width of 2. The
 3699 last line defines the **DEFAULT_WIDTH** to 1.

3703 **6 REPERTOIREMAP** (controversial)

3704
 3705 FDCC-set and Charmap sources may be specified in a coded character set independent
 3706 way, using symbolic character names. The relation between the symbolic character names
 3707 and characters may be specified via a Repertoiremap, which defines the repertoire of
 3708 characters defined for a FDCC-set, and the symbolic character names and corresponding
 3709 abstract character (by a reference to ISO/IEC 10646).

3710
 3711 The repertoire mapping is defined by specifying the symbolic character name and the
 3712 ISO/IEC 10646 code position in hexadecimal form (with a preceding 'U') and optionally
 3713 the long ISO/IEC 10646 character name in the following syntax:

```
3714 "%s %s %s\n",<symbolic-name>,<short-identifier>,<comments>
```

3715
 3716 The symbolic character name and the short identifier are each surrounded by angle
 3717 brackets <>, and the fields are separated by one or more spaces or tabs on a line. If a
 3718 right angle bracket or an escape character is used within a symbolic name, it is preceded
 3719 by the escape character. The short identifier is either a ISO/IEC 10646 short identifier, or,
 3720

3721 if that does not exist, a short identifier in the range <P0000>..<>PFFFF> or
 3722 <P00000000>..<>P7FFFFFFF>.

3723
 3724 The escape character can be redefined from the default reverse solidus (\) with the first
 3725 line of the Repertoiremap containing the string "escape_char" followed by one or more
 3726 spaces or tabs and then the escape character.

3727
 3728 Several symbolic character names can refer to the same abstract character, and are then
 3729 used as synonyms in FDCC-sets and charmaps. The set of <U0000>..<>UFFFF> and
 3730 <U00000000>..<>U7FFFFFFF> symbolic names (no lowercase letters) are predefined and
 3731 refer to the corresponding code points of ISO/IEC 10646 with the same short identifier.
 3732

3733 The "i18nrep" repertoiremap is defined to accommodate prior art, such as defined in
 3734 Annex G of the ISO/IEC 9945-2:1993 standard, and used by ISO and IEC member bodies
 3735 in their national POSIX locale specifications, and as used in POSIX locales distributed by
 3736 the ISO/IEC POSIX working group and The Open Group. Many POSIX charmaps
 3737 registered with ISO/IEC 15897 use these symbolic names. It also reflects use on the
 3738 Internet, and many of the Internet registered charsets are specified using these symbolic
 3739 names. The "i18nrep" repertoiremap thus facilitates reuse of both POSIX locale data and
 3740 POSIX charmaps with data from this Technical Report. The sequence <a8>..<>z8> are used
 3741 as hooks for tailoring to denote the last accented Latin letter of each of the ISO/IEC 646
 3742 letters <a>..<>z>, so that tailorings that need to have specifications after the last letter of
 3743 such a family, for example to introduce a new letter of an alphabet, can do so with a
 3744 reference that is stable over different versions of the "i18n" FDCC-set. The contents of the
 3745 "i18nrep" repertoiremap is as follows:

```

3746 escape_char /
3747 <NUL>          <U0000>  NULL (NUL)
3748 <SOH>         <U0001>  START OF HEADING (SOH)
3749 <STX>         <U0002>  START OF TEXT (STX)
3750 <ETX>         <U0003>  END OF TEXT (ETX)
3751 <EOT>         <U0004>  END OF TRANSMISSION (EOT)
3752 <ENQ>         <U0005>  ENQUIRY (ENQ)
3753 <ACK>         <U0006>  ACKNOWLEDGE (ACK)
3754 <alert>      <U0007>  BELL (BEL)
3755 <BEL>         <U0007>  BELL (BEL)
3756 <backspace>  <U0008>  BACKSPACE (BS)
3757 <tab>         <U0009>  CHARACTER TABULATION (HT)
3758 <newline>    <U000A>  LINE FEED (LF)
3759 <vertical-tab> <U000B>  LINE TABULATION (VT)
3760 <form-feed>  <U000C>  FORM FEED (FF)
3761 <carriage-return> <U000D>  CARRIAGE RETURN (CR)
3762 <DLE>         <U0010>  DATALINK ESCAPE (DLE)
3763 <DC1>         <U0011>  DEVICE CONTROL ONE (DC1)
3764 <DC2>         <U0012>  DEVICE CONTROL TWO (DC2)
3765 <DC3>         <U0013>  DEVICE CONTROL THREE (DC3)
3766 <DC4>         <U0014>  DEVICE CONTROL FOUR (DC4)
3767 <NAK>         <U0015>  NEGATIVE ACKNOWLEDGE (NAK)
3768 <SYN>         <U0016>  SYNCHRONOUS IDLE (SYN)
3769 <ETB>         <U0017>  END OF TRANSMISSION BLOCK (ETB)
3770 <CAN>         <U0018>  CANCEL (CAN)
3771 <SUB>         <U001A>  SUBSTITUTE (SUB)
3772 <ESC>         <U001B>  ESCAPE (ESC)
3773 <IS4>         <U001C>  FILE SEPARATOR (IS4)
3774 <IS3>         <U001D>  GROUP SEPARATOR (IS3)
3775 <intro>      <U001D>  GROUP SEPARATOR (IS3)
3776 <IS2>         <U001E>  RECORD SEPARATOR (IS2)
3777 <IS1>         <U001F>  UNIT SEPARATOR (IS1)
3778 <DEL>         <U007F>  DELETE (DEL)
3779 <space>      <U0020>  SPACE
3780 <exclamation-mark> <U0021>  EXCLAMATION MARK
3781 <quotation-mark> <U0022>  QUOTATION MARK
3782 <number-sign>  <U0023>  NUMBER SIGN
3783 <dollar-sign> <U0024>  DOLLAR SIGN
3784 <percent-sign> <U0025>  PERCENT SIGN
3785 <ampersand>    <U0026>  AMPERSAND
3786 <apostrophe> <U0027>  APOSTROPHE
3787 <left-parenthesis> <U0028>  LEFT PARENTHESIS
3788 <right-parenthesis> <U0029>  RIGHT PARENTHESIS
3789 <asterisk>    <U002A>  ASTERISK
3790 <plus-sign>   <U002B>  PLUS SIGN
  
```

<comma>	<U002C>	COMMA
<hyphen>	<U002D>	HYPHEN-MINUS
<hyphen-minus>	<U002D>	HYPHEN-MINUS
<period>	<U002E>	FULL STOP
<full-stop>	<U002E>	FULL STOP
<slash>	<U002F>	SOLIDUS
<solidus>	<U002F>	SOLIDUS
<zero>	<U0030>	DIGIT ZERO
<one>	<U0031>	DIGIT ONE
<two>	<U0032>	DIGIT TWO
<three>	<U0033>	DIGIT THREE
<four>	<U0034>	DIGIT FOUR
<five>	<U0035>	DIGIT FIVE
<six>	<U0036>	DIGIT SIX
<seven>	<U0037>	DIGIT SEVEN
<eight>	<U0038>	DIGIT EIGHT
<nine>	<U0039>	DIGIT NINE
<colon>	<U003A>	COLON
<semicolon>	<U003B>	SEMICOLON
<less-than-sign>	<U003C>	LESS-THAN SIGN
<equals-sign>	<U003D>	EQUALS SIGN
<greater-than-sign>	<U003E>	GREATER-THAN SIGN
<question-mark>	<U003F>	QUESTION MARK
<commercial-at>	<U0040>	COMMERCIAL AT
<left-square-bracket>	<U005B>	LEFT SQUARE BRACKET
<backslash>	<U005C>	REVERSE SOLIDUS
<reverse-solidus>	<U005C>	REVERSE SOLIDUS
<right-square-bracket>	<U005D>	RIGHT SQUARE BRACKET
<circumflex>	<U005E>	CIRCUMFLEX ACCENT
<circumflex-accent>	<U005E>	CIRCUMFLEX ACCENT
<underscore>	<U005F>	LOW LINE
<low-line>	<U005F>	LOW LINE
<grave-accent>	<U0060>	GRAVE ACCENT
<left-brace>	<U007B>	LEFT CURLY BRACKET
<left-curly-bracket>	<U007B>	LEFT CURLY BRACKET
<vertical-line>	<U007C>	VERTICAL LINE
<right-brace>	<U007D>	RIGHT CURLY BRACKET
<right-curly-bracket>	<U007D>	RIGHT CURLY BRACKET
<tilde>	<U007E>	TILDE
<a8>	<P0001>	Weight indicating the position of the last a
<b8>	<P0002>	Weight indicating the position of the last b
<c8>	<P0003>	Weight indicating the position of the last c
<d8>	<P0004>	Weight indicating the position of the last d
<e8>	<P0005>	Weight indicating the position of the last e
<f8>	<P0006>	Weight indicating the position of the last f
<g8>	<P0007>	Weight indicating the position of the last g
<h8>	<P0008>	Weight indicating the position of the last h
<i8>	<P0009>	Weight indicating the position of the last i
<j8>	<P0010>	Weight indicating the position of the last j
<k8>	<P0011>	Weight indicating the position of the last k
<l8>	<P0012>	Weight indicating the position of the last l
<m8>	<P0013>	Weight indicating the position of the last m
<n8>	<P0014>	Weight indicating the position of the last n
<o8>	<P0015>	Weight indicating the position of the last o
<p8>	<P0016>	Weight indicating the position of the last p
<q8>	<P0017>	Weight indicating the position of the last q
<r8>	<P0018>	Weight indicating the position of the last r
<s8>	<P0019>	Weight indicating the position of the last s
<t8>	<P0020>	Weight indicating the position of the last t
<u8>	<P0021>	Weight indicating the position of the last u
<v8>	<P0022>	Weight indicating the position of the last v
<w8>	<P0023>	Weight indicating the position of the last w
<x8>	<P0024>	Weight indicating the position of the last x
<y8>	<P0025>	Weight indicating the position of the last y
<z8>	<P0026>	Weight indicating the position of the last z
<NU>	<U0000>	NULL (NUL)
<SH>	<U0001>	START OF HEADING (SOH)
<SX>	<U0002>	START OF TEXT (STX)
<EX>	<U0003>	END OF TEXT (ETX)
<ET>	<U0004>	END OF TRANSMISSION (EOT)
<EQ>	<U0005>	ENQUIRY (ENQ)
<AK>	<U0006>	ACKNOWLEDGE (ACK)
<BL>	<U0007>	BELL (BEL)
<BS>	<U0008>	BACKSPACE (BS)
<HT>	<U0009>	CHARACTER TABULATION (HT)
<LF>	<U000A>	LINE FEED (LF)
<VT>	<U000B>	LINE TABULATION (VT)
<FF>	<U000C>	FORM FEED (FF)
<CR>	<U000D>	CARRIAGE RETURN (CR)
<SO>	<U000E>	SHIFT OUT (SO)
<SI>	<U000F>	SHIFT IN (SI)
<DL>	<U0010>	DATALINK ESCAPE (DLE)
<D1>	<U0011>	DEVICE CONTROL ONE (DC1)
<D2>	<U0012>	DEVICE CONTROL TWO (DC2)
<D3>	<U0013>	DEVICE CONTROL THREE (DC3)
<D4>	<U0014>	DEVICE CONTROL FOUR (DC4)

<NK>	<U0015>	NEGATIVE ACKNOWLEDGE (NAK)
<SY>	<U0016>	SYNCHRONOUS IDLE (SYN)
<EB>	<U0017>	END OF TRANSMISSION BLOCK (ETB)
<CN>	<U0018>	CANCEL (CAN)
	<U0019>	END OF MEDIUM (EM)
<SB>	<U001A>	SUBSTITUTE (SUB)
<EC>	<U001B>	ESCAPE (ESC)
<FS>	<U001C>	FILE SEPARATOR (IS4)
<GS>	<U001D>	GROUP SEPARATOR (IS3)
<RS>	<U001E>	RECORD SEPARATOR (IS2)
<US>	<U001F>	UNIT SEPARATOR (IS1)
<DT>	<U007F>	DELETE (DEL)
<PA>	<U0080>	PADDING CHARACTER (PAD)
<HO>	<U0081>	HIGH OCTET PRESET (HOP)
<BH>	<U0082>	BREAK PERMITTED HERE (BPH)
<NH>	<U0083>	NO BREAK HERE (NBH)
<IN>	<U0084>	INDEX (IND)
<NL>	<U0085>	NEXT LINE (NEL)
<SA>	<U0086>	START OF SELECTED AREA (SSA)
<ES>	<U0087>	END OF SELECTED AREA (ESA)
<HS>	<U0088>	CHARACTER TABULATION SET (HTS)
<HJ>	<U0089>	CHARACTER TABULATION WITH JUSTIFICATION (HTJ)
<VS>	<U008A>	LINE TABULATION SET (VTS)
<PD>	<U008B>	PARTIAL LINE FORWARD (PLD)
<PU>	<U008C>	PARTIAL LINE BACKWARD (PLU)
<RI>	<U008D>	REVERSE LINE FEED (RI)
<S2>	<U008E>	SINGLE-SHIFT TWO (SS2)
<S3>	<U008F>	SINGLE-SHIFT THREE (SS3)
<DC>	<U0090>	DEVICE CONTROL STRING (DCS)
<P1>	<U0091>	PRIVATE USE ONE (PU1)
<P2>	<U0092>	PRIVATE USE TWO (PU2)
<TS>	<U0093>	SET TRANSMIT STATE (STS)
<CC>	<U0094>	CANCEL CHARACTER (CCH)
<MW>	<U0095>	MESSAGE WAITING (MW)
<SG>	<U0096>	START OF GUARDED AREA (SPA)
<EG>	<U0097>	END OF GUARDED AREA (EPA)
<SS>	<U0098>	START OF STRING (SOS)
<GC>	<U0099>	SINGLE GRAPHIC CHARACTER INTRODUCER (SGCI)
<SC>	<U009A>	SINGLE CHARACTER INTRODUCER (SCI)
<CI>	<U009B>	CONTROL SEQUENCE INTRODUCER (CSI)
<ST>	<U009C>	STRING TERMINATOR (ST)
<OC>	<U009D>	OPERATING SYSTEM COMMAND (OSC)
<PM>	<U009E>	PRIVACY MESSAGE (PM)
<AC>	<U009F>	APPLICATION PROGRAM COMMAND (APC)
<SP>	<U0020>	SPACE
<!>	<U0021>	EXCLAMATION MARK
<">	<U0022>	QUOTATION MARK
<N>	<U0023>	NUMBER SIGN
<D>	<U0024>	DOLLAR SIGN
<%>	<U0025>	PERCENT SIGN
<&>	<U0026>	AMPERSAND
<'>	<U0027>	APOSTROPHE
<(>	<U0028>	LEFT PARENTHESIS
<)>	<U0029>	RIGHT PARENTHESIS
<*>	<U002A>	ASTERISK
<+>	<U002B>	PLUS SIGN
<, >	<U002C>	COMMA
<- >	<U002D>	HYPHEN-MINUS
<. >	<U002E>	FULL STOP
<//>	<U002F>	SOLIDUS
<0>	<U0030>	DIGIT ZERO
<1>	<U0031>	DIGIT ONE
<2>	<U0032>	DIGIT TWO
<3>	<U0033>	DIGIT THREE
<4>	<U0034>	DIGIT FOUR
<5>	<U0035>	DIGIT FIVE
<6>	<U0036>	DIGIT SIX
<7>	<U0037>	DIGIT SEVEN
<8>	<U0038>	DIGIT EIGHT
<9>	<U0039>	DIGIT NINE
<:>	<U003A>	COLON
< ; >	<U003B>	SEMICOLON
<< >	<U003C>	LESS-THAN SIGN
<=>	<U003D>	EQUALS SIGN
</>>	<U003E>	GREATER-THAN SIGN
<?>	<U003F>	QUESTION MARK
<@>	<U0040>	COMMERCIAL AT
<A>	<U0041>	LATIN CAPITAL LETTER A
	<U0042>	LATIN CAPITAL LETTER B
<C>	<U0043>	LATIN CAPITAL LETTER C
<D>	<U0044>	LATIN CAPITAL LETTER D
<E>	<U0045>	LATIN CAPITAL LETTER E
<F>	<U0046>	LATIN CAPITAL LETTER F
<G>	<U0047>	LATIN CAPITAL LETTER G
<H>	<U0048>	LATIN CAPITAL LETTER H
<I>	<U0049>	LATIN CAPITAL LETTER I
<J>	<U004A>	LATIN CAPITAL LETTER J
<K>	<U004B>	LATIN CAPITAL LETTER K

<L>	<U004C>	LATIN CAPITAL LETTER L
<M>	<U004D>	LATIN CAPITAL LETTER M
<N>	<U004E>	LATIN CAPITAL LETTER N
<O>	<U004F>	LATIN CAPITAL LETTER O
<P>	<U0050>	LATIN CAPITAL LETTER P
<Q>	<U0051>	LATIN CAPITAL LETTER Q
<R>	<U0052>	LATIN CAPITAL LETTER R
<S>	<U0053>	LATIN CAPITAL LETTER S
<T>	<U0054>	LATIN CAPITAL LETTER T
<U>	<U0055>	LATIN CAPITAL LETTER U
<V>	<U0056>	LATIN CAPITAL LETTER V
<W>	<U0057>	LATIN CAPITAL LETTER W
<X>	<U0058>	LATIN CAPITAL LETTER X
<Y>	<U0059>	LATIN CAPITAL LETTER Y
<Z>	<U005A>	LATIN CAPITAL LETTER Z
<<(>	<U005B>	LEFT SQUARE BRACKET
</////>	<U005C>	REVERSE SOLIDUS
<)/>>	<U005D>	RIGHT SQUARE BRACKET
<' />>	<U005E>	CIRCUMFLEX ACCENT
<_>	<U005F>	LOW LINE
<'!>	<U0060>	GRAVE ACCENT
<a>	<U0061>	LATIN SMALL LETTER A
	<U0062>	LATIN SMALL LETTER B
<c>	<U0063>	LATIN SMALL LETTER C
<d>	<U0064>	LATIN SMALL LETTER D
<e>	<U0065>	LATIN SMALL LETTER E
<f>	<U0066>	LATIN SMALL LETTER F
<g>	<U0067>	LATIN SMALL LETTER G
<h>	<U0068>	LATIN SMALL LETTER H
<i>	<U0069>	LATIN SMALL LETTER I
<j>	<U006A>	LATIN SMALL LETTER J
<k>	<U006B>	LATIN SMALL LETTER K
<l>	<U006C>	LATIN SMALL LETTER L
<m>	<U006D>	LATIN SMALL LETTER M
<n>	<U006E>	LATIN SMALL LETTER N
<o>	<U006F>	LATIN SMALL LETTER O
<p>	<U0070>	LATIN SMALL LETTER P
<q>	<U0071>	LATIN SMALL LETTER Q
<r>	<U0072>	LATIN SMALL LETTER R
<s>	<U0073>	LATIN SMALL LETTER S
<t>	<U0074>	LATIN SMALL LETTER T
<u>	<U0075>	LATIN SMALL LETTER U
<v>	<U0076>	LATIN SMALL LETTER V
<w>	<U0077>	LATIN SMALL LETTER W
<x>	<U0078>	LATIN SMALL LETTER X
<y>	<U0079>	LATIN SMALL LETTER Y
<z>	<U007A>	LATIN SMALL LETTER Z
<(!>	<U007B>	LEFT CURLY BRACKET
<!!!>	<U007C>	VERTICAL LINE
<!)>	<U007D>	RIGHT CURLY BRACKET
<' ?>	<U007E>	TILDE
<NS>	<U00A0>	NO-BREAK SPACE
<!I>	<U00A1>	INVERTED EXCLAMATION MARK
<Ct>	<U00A2>	CENT SIGN
<Pd>	<U00A3>	POUND SIGN
<Cu>	<U00A4>	CURRENCY SIGN
<Ye>	<U00A5>	YEN SIGN
<BB>	<U00A6>	BROKEN BAR
<SE>	<U00A7>	SECTION SIGN
<' :>	<U00A8>	DIAERESIS
<Co>	<U00A9>	COPYRIGHT SIGN
<-a>	<U00AA>	FEMININE ORDINAL INDICATOR
<<<>	<U00AB>	LEFT-POINTING DOUBLE ANGLE QUOTATION MARK
<NO>	<U00AC>	NOT SIGN
<-->	<U00AD>	SOFT HYPHEN
<Rg>	<U00AE>	REGISTERED SIGN
<' m>	<U00AF>	MACRON
<DG>	<U00B0>	DEGREE SIGN
<+ ->	<U00B1>	PLUS-MINUS SIGN
<2S>	<U00B2>	SUPERSCRIFT TWO
<3S>	<U00B3>	SUPERSCRIFT THREE
<' ' >	<U00B4>	ACUTE ACCENT
<My>	<U00B5>	MICRO SIGN
<PI>	<U00B6>	PILCROW SIGN
<.M>	<U00B7>	MIDDLE DOT
<' , >	<U00B8>	CEDILLA
<1S>	<U00B9>	SUPERSCRIFT ONE
<-o>	<U00BA>	MASCULINE ORDINAL INDICATOR
</>/>>	<U00BB>	RIGHT-POINTING DOUBLE ANGLE QUOTATION MARK
<14>	<U00BC>	VULGAR FRACTION ONE QUARTER
<12>	<U00BD>	VULGAR FRACTION ONE HALF
<34>	<U00BE>	VULGAR FRACTION THREE QUARTERS
<?I>	<U00BF>	INVERTED QUESTION MARK
<A!>	<U00C0>	LATIN CAPITAL LETTER A WITH GRAVE
<A'>	<U00C1>	LATIN CAPITAL LETTER A WITH ACUTE
<A/ >>	<U00C2>	LATIN CAPITAL LETTER A WITH CIRCUMFLEX
<A?>	<U00C3>	LATIN CAPITAL LETTER A WITH TILDE
<A: >	<U00C4>	LATIN CAPITAL LETTER A WITH DIAERESIS

40	<AA>	<U00C5>	LATIN CAPITAL LETTER A WITH RING ABOVE
41	<AE>	<U00C6>	LATIN CAPITAL LETTER AE (ash)
42	<C,>	<U00C7>	LATIN CAPITAL LETTER C WITH CEDILLA
43	<E!>	<U00C8>	LATIN CAPITAL LETTER E WITH GRAVE
44	<E'>	<U00C9>	LATIN CAPITAL LETTER E WITH ACUTE
45	<E/>>	<U00CA>	LATIN CAPITAL LETTER E WITH CIRCUMFLEX
46	<E:>	<U00CB>	LATIN CAPITAL LETTER E WITH DIAERESIS
47	<I!>	<U00CC>	LATIN CAPITAL LETTER I WITH GRAVE
48	<I'>	<U00CD>	LATIN CAPITAL LETTER I WITH ACUTE
49	<I/>>	<U00CE>	LATIN CAPITAL LETTER I WITH CIRCUMFLEX
50	<I:>	<U00CF>	LATIN CAPITAL LETTER I WITH DIAERESIS
51	<D->	<U00D0>	LATIN CAPITAL LETTER ETH (Icelandic)
52	<N?>	<U00D1>	LATIN CAPITAL LETTER N WITH TILDE
53	<O!>	<U00D2>	LATIN CAPITAL LETTER O WITH GRAVE
54	<O'>	<U00D3>	LATIN CAPITAL LETTER O WITH ACUTE
55	<O/>>	<U00D4>	LATIN CAPITAL LETTER O WITH CIRCUMFLEX
56	<O?>	<U00D5>	LATIN CAPITAL LETTER O WITH TILDE
57	<O:>	<U00D6>	LATIN CAPITAL LETTER O WITH DIAERESIS
58	<*X>	<U00D7>	MULTIPLICATION SIGN
59	<O//>	<U00D8>	LATIN CAPITAL LETTER O WITH STROKE
60	<U!>	<U00D9>	LATIN CAPITAL LETTER U WITH GRAVE
61	<U'>	<U00DA>	LATIN CAPITAL LETTER U WITH ACUTE
62	<U/>>	<U00DB>	LATIN CAPITAL LETTER U WITH CIRCUMFLEX
63	<U:>	<U00DC>	LATIN CAPITAL LETTER U WITH DIAERESIS
64	<Y'>	<U00DD>	LATIN CAPITAL LETTER Y WITH ACUTE
65	<TH>	<U00DE>	LATIN CAPITAL LETTER THORN (Icelandic)
66	<ss>	<U00DF>	LATIN SMALL LETTER SHARP S (German)
67	<a!>	<U00E0>	LATIN SMALL LETTER A WITH GRAVE
68	<a'>	<U00E1>	LATIN SMALL LETTER A WITH ACUTE
69	<a/>>	<U00E2>	LATIN SMALL LETTER A WITH CIRCUMFLEX
70	<a?>	<U00E3>	LATIN SMALL LETTER A WITH TILDE
71	<a:>	<U00E4>	LATIN SMALL LETTER A WITH DIAERESIS
72	<aa>	<U00E5>	LATIN SMALL LETTER A WITH RING ABOVE
73	<ae>	<U00E6>	LATIN SMALL LETTER AE (ash)
74	<c,>	<U00E7>	LATIN SMALL LETTER C WITH CEDILLA
75	<e!>	<U00E8>	LATIN SMALL LETTER E WITH GRAVE
76	<e'>	<U00E9>	LATIN SMALL LETTER E WITH ACUTE
77	<e/>>	<U00EA>	LATIN SMALL LETTER E WITH CIRCUMFLEX
78	<e:>	<U00EB>	LATIN SMALL LETTER E WITH DIAERESIS
79	<i!>	<U00EC>	LATIN SMALL LETTER I WITH GRAVE
80	<i'>	<U00ED>	LATIN SMALL LETTER I WITH ACUTE
81	<i/>>	<U00EE>	LATIN SMALL LETTER I WITH CIRCUMFLEX
82	<i:>	<U00EF>	LATIN SMALL LETTER I WITH DIAERESIS
83	<d->	<U00F0>	LATIN SMALL LETTER ETH (Icelandic)
84	<n?>	<U00F1>	LATIN SMALL LETTER N WITH TILDE
85	<o!>	<U00F2>	LATIN SMALL LETTER O WITH GRAVE
86	<o'>	<U00F3>	LATIN SMALL LETTER O WITH ACUTE
87	<o/>>	<U00F4>	LATIN SMALL LETTER O WITH CIRCUMFLEX
88	<o?>	<U00F5>	LATIN SMALL LETTER O WITH TILDE
89	<o:>	<U00F6>	LATIN SMALL LETTER O WITH DIAERESIS
90	<-:>	<U00F7>	DIVISION SIGN
91	<o//>	<U00F8>	LATIN SMALL LETTER O WITH STROKE
92	<u!>	<U00F9>	LATIN SMALL LETTER U WITH GRAVE
93	<u'>	<U00FA>	LATIN SMALL LETTER U WITH ACUTE
94	<u/>>	<U00FB>	LATIN SMALL LETTER U WITH CIRCUMFLEX
95	<u:>	<U00FC>	LATIN SMALL LETTER U WITH DIAERESIS
96	<y'>	<U00FD>	LATIN SMALL LETTER Y WITH ACUTE
97	<th>	<U00FE>	LATIN SMALL LETTER THORN (Icelandic)
98	<y:>	<U00FF>	LATIN SMALL LETTER Y WITH DIAERESIS
99	<A->	<U0100>	LATIN CAPITAL LETTER A WITH MACRON
100	<a->	<U0101>	LATIN SMALL LETTER A WITH MACRON
101	<A(>	<U0102>	LATIN CAPITAL LETTER A WITH BREVE
102	<a(>	<U0103>	LATIN SMALL LETTER A WITH BREVE
103	<A; >	<U0104>	LATIN CAPITAL LETTER A WITH OGONEK
104	<a; >	<U0105>	LATIN SMALL LETTER A WITH OGONEK
105	<C'>	<U0106>	LATIN CAPITAL LETTER C WITH ACUTE
106	<c'>	<U0107>	LATIN SMALL LETTER C WITH ACUTE
107	<C/>>	<U0108>	LATIN CAPITAL LETTER C WITH CIRCUMFLEX
108	<c/>>	<U0109>	LATIN SMALL LETTER C WITH CIRCUMFLEX
109	<C.>	<U010A>	LATIN CAPITAL LETTER C WITH DOT ABOVE
110	<c.>	<U010B>	LATIN SMALL LETTER C WITH DOT ABOVE
111	<C<>	<U010C>	LATIN CAPITAL LETTER C WITH CARON
112	<c<>	<U010D>	LATIN SMALL LETTER C WITH CARON
113	<D<>	<U010E>	LATIN CAPITAL LETTER D WITH CARON
114	<d<>	<U010F>	LATIN SMALL LETTER D WITH CARON
115	<D//>	<U0110>	LATIN CAPITAL LETTER D WITH STROKE
116	<d//>	<U0111>	LATIN SMALL LETTER D WITH STROKE
117	<E->	<U0112>	LATIN CAPITAL LETTER E WITH MACRON
118	<e->	<U0113>	LATIN SMALL LETTER E WITH MACRON
119	<E(>	<U0114>	LATIN CAPITAL LETTER E WITH BREVE
120	<e(>	<U0115>	LATIN SMALL LETTER E WITH BREVE
121	<E.>	<U0116>	LATIN CAPITAL LETTER E WITH DOT ABOVE
122	<e.>	<U0117>	LATIN SMALL LETTER E WITH DOT ABOVE
123	<E; >	<U0118>	LATIN CAPITAL LETTER E WITH OGONEK
124	<e; >	<U0119>	LATIN SMALL LETTER E WITH OGONEK
125	<E<>	<U011A>	LATIN CAPITAL LETTER E WITH CARON
126	<e<>	<U011B>	LATIN SMALL LETTER E WITH CARON
127	<G/>>	<U011C>	LATIN CAPITAL LETTER G WITH CIRCUMFLEX

4	<g/>	<U011D>	LATIN SMALL LETTER G WITH CIRCUMFLEX
4	<G(>	<U011E>	LATIN CAPITAL LETTER G WITH BREVE
4	<g(>	<U011F>	LATIN SMALL LETTER G WITH BREVE
4	<G.>	<U0120>	LATIN CAPITAL LETTER G WITH DOT ABOVE
4	<g.>	<U0121>	LATIN SMALL LETTER G WITH DOT ABOVE
4	<G.>	<U0122>	LATIN CAPITAL LETTER G WITH CEDILLA
4	<g.>	<U0123>	LATIN SMALL LETTER G WITH CEDILLA
4	<H/>	<U0124>	LATIN CAPITAL LETTER H WITH CIRCUMFLEX
4	<h/>	<U0125>	LATIN SMALL LETTER H WITH CIRCUMFLEX
4	<H//>	<U0126>	LATIN CAPITAL LETTER H WITH STROKE
4	<h//>	<U0127>	LATIN SMALL LETTER H WITH STROKE
4	<I?>	<U0128>	LATIN CAPITAL LETTER I WITH TILDE
4	<i?>	<U0129>	LATIN SMALL LETTER I WITH TILDE
4	<I->	<U012A>	LATIN CAPITAL LETTER I WITH MACRON
4	<i->	<U012B>	LATIN SMALL LETTER I WITH MACRON
4	<I(>	<U012C>	LATIN CAPITAL LETTER I WITH BREVE
4	<i(>	<U012D>	LATIN SMALL LETTER I WITH BREVE
4	<I; >	<U012E>	LATIN CAPITAL LETTER I WITH OGONEK
4	<i; >	<U012F>	LATIN SMALL LETTER I WITH OGONEK
4	<I.>	<U0130>	LATIN CAPITAL LETTER I WITH DOT ABOVE
4	<i.>	<U0131>	LATIN SMALL LETTER DOTLESS I
4	<IJ>	<U0132>	LATIN CAPITAL LIGATURE IJ
4	<ij>	<U0133>	LATIN SMALL LIGATURE IJ
4	<J/>	<U0134>	LATIN CAPITAL LETTER J WITH CIRCUMFLEX
4	<j/>	<U0135>	LATIN SMALL LETTER J WITH CIRCUMFLEX
4	<K.>	<U0136>	LATIN CAPITAL LETTER K WITH CEDILLA
4	<k.>	<U0137>	LATIN SMALL LETTER K WITH CEDILLA
4	<kk>	<U0138>	LATIN SMALL LETTER KRA (Greenlandic)
4	<L'>	<U0139>	LATIN CAPITAL LETTER L WITH ACUTE
4	<l'>	<U013A>	LATIN SMALL LETTER L WITH ACUTE
4	<L.>	<U013B>	LATIN CAPITAL LETTER L WITH CEDILLA
4	<l.>	<U013C>	LATIN SMALL LETTER L WITH CEDILLA
4	<L<>	<U013D>	LATIN CAPITAL LETTER L WITH CARON
4	<l<>	<U013E>	LATIN SMALL LETTER L WITH CARON
4	<L.>	<U013F>	LATIN CAPITAL LETTER L WITH MIDDLE DOT
4	<l.>	<U0140>	LATIN SMALL LETTER L WITH MIDDLE DOT
4	<L//>	<U0141>	LATIN CAPITAL LETTER L WITH STROKE
4	<l//>	<U0142>	LATIN SMALL LETTER L WITH STROKE
4	<N'>	<U0143>	LATIN CAPITAL LETTER N WITH ACUTE
4	<n'>	<U0144>	LATIN SMALL LETTER N WITH ACUTE
4	<N.>	<U0145>	LATIN CAPITAL LETTER N WITH CEDILLA
4	<n.>	<U0146>	LATIN SMALL LETTER N WITH CEDILLA
4	<N<>	<U0147>	LATIN CAPITAL LETTER N WITH CARON
4	<n<>	<U0148>	LATIN SMALL LETTER N WITH CARON
4	<'n>	<U0149>	LATIN SMALL LETTER N PRECEDED BY APOSTROPHE
4	<NG>	<U014A>	LATIN CAPITAL LETTER ENG (Sami)
4	<ng>	<U014B>	LATIN SMALL LETTER ENG (Sami)
4	<O->	<U014C>	LATIN CAPITAL LETTER O WITH MACRON
4	<o->	<U014D>	LATIN SMALL LETTER O WITH MACRON
4	<O(>	<U014E>	LATIN CAPITAL LETTER O WITH BREVE
4	<o(>	<U014F>	LATIN SMALL LETTER O WITH BREVE
4	<O" >	<U0150>	LATIN CAPITAL LETTER O WITH DOUBLE ACUTE
4	<o" >	<U0151>	LATIN SMALL LETTER O WITH DOUBLE ACUTE
4	<OE>	<U0152>	LATIN CAPITAL LIGATURE OE
4	<oe>	<U0153>	LATIN SMALL LIGATURE OE
4	<R'>	<U0154>	LATIN CAPITAL LETTER R WITH ACUTE
4	<r'>	<U0155>	LATIN SMALL LETTER R WITH ACUTE
4	<R.>	<U0156>	LATIN CAPITAL LETTER R WITH CEDILLA
4	<r.>	<U0157>	LATIN SMALL LETTER R WITH CEDILLA
4	<R<>	<U0158>	LATIN CAPITAL LETTER R WITH CARON
4	<r<>	<U0159>	LATIN SMALL LETTER R WITH CARON
4	<S'>	<U015A>	LATIN CAPITAL LETTER S WITH ACUTE
4	<s'>	<U015B>	LATIN SMALL LETTER S WITH ACUTE
4	<S/>	<U015C>	LATIN CAPITAL LETTER S WITH CIRCUMFLEX
4	<s/>	<U015D>	LATIN SMALL LETTER S WITH CIRCUMFLEX
4	<S.>	<U015E>	LATIN CAPITAL LETTER S WITH CEDILLA
4	<s.>	<U015F>	LATIN SMALL LETTER S WITH CEDILLA
4	<S<>	<U0160>	LATIN CAPITAL LETTER S WITH CARON
4	<s<>	<U0161>	LATIN SMALL LETTER S WITH CARON
4	<T.>	<U0162>	LATIN CAPITAL LETTER T WITH CEDILLA
4	<t.>	<U0163>	LATIN SMALL LETTER T WITH CEDILLA
4	<T<>	<U0164>	LATIN CAPITAL LETTER T WITH CARON
4	<t<>	<U0165>	LATIN SMALL LETTER T WITH CARON
4	<T//>	<U0166>	LATIN CAPITAL LETTER T WITH STROKE
4	<t//>	<U0167>	LATIN SMALL LETTER T WITH STROKE
4	<U?>	<U0168>	LATIN CAPITAL LETTER U WITH TILDE
4	<u?>	<U0169>	LATIN SMALL LETTER U WITH TILDE
4	<U->	<U016A>	LATIN CAPITAL LETTER U WITH MACRON
4	<u->	<U016B>	LATIN SMALL LETTER U WITH MACRON
4	<U(>	<U016C>	LATIN CAPITAL LETTER U WITH BREVE
4	<u(>	<U016D>	LATIN SMALL LETTER U WITH BREVE
4	<U0>	<U016E>	LATIN CAPITAL LETTER U WITH RING ABOVE
4	<u0>	<U016F>	LATIN SMALL LETTER U WITH RING ABOVE
4	<U" >	<U0170>	LATIN CAPITAL LETTER U WITH DOUBLE ACUTE
4	<u" >	<U0171>	LATIN SMALL LETTER U WITH DOUBLE ACUTE
4	<U; >	<U0172>	LATIN CAPITAL LETTER U WITH OGONEK
4	<u; >	<U0173>	LATIN SMALL LETTER U WITH OGONEK
4	<W/>	<U0174>	LATIN CAPITAL LETTER W WITH CIRCUMFLEX

42	<w/>>	<U0175>	LATIN SMALL LETTER W WITH CIRCUMFLEX
43	<Y/>>	<U0176>	LATIN CAPITAL LETTER Y WITH CIRCUMFLEX
44	<y/>>	<U0177>	LATIN SMALL LETTER Y WITH CIRCUMFLEX
45	<Y'>	<U0178>	LATIN CAPITAL LETTER Y WITH DIAERESIS
46	<Z'>	<U0179>	LATIN CAPITAL LETTER Z WITH ACUTE
47	<z'>	<U017A>	LATIN SMALL LETTER Z WITH ACUTE
48	<Z.>	<U017B>	LATIN CAPITAL LETTER Z WITH DOT ABOVE
49	<z.>	<U017C>	LATIN SMALL LETTER Z WITH DOT ABOVE
50	<Z<>	<U017D>	LATIN CAPITAL LETTER Z WITH CARON
51	<z<>	<U017E>	LATIN SMALL LETTER Z WITH CARON
52	<sl>	<U017F>	LATIN SMALL LETTER LONG S
53	<b//>	<U0180>	LATIN SMALL LETTER B WITH STROKE
54	<B2>	<U0181>	LATIN CAPITAL LETTER B WITH HOOK
55	<C2>	<U0187>	LATIN CAPITAL LETTER C WITH HOOK
56	<c2>	<U0188>	LATIN SMALL LETTER C WITH HOOK
57	<F2>	<U0191>	LATIN CAPITAL LETTER F WITH HOOK
58	<f2>	<U0192>	LATIN SMALL LETTER F WITH HOOK
59	<K2>	<U0198>	LATIN CAPITAL LETTER K WITH HOOK
60	<k2>	<U0199>	LATIN SMALL LETTER K WITH HOOK
61	<O9>	<U01A0>	LATIN CAPITAL LETTER O WITH HORN
62	<o9>	<U01A1>	LATIN SMALL LETTER O WITH HORN
63	<OI>	<U01A2>	LATIN CAPITAL LETTER OI
64	<oi>	<U01A3>	LATIN SMALL LETTER OI
65	<yr>	<U01A6>	LATIN LETTER YR
66	<U9>	<U01AF>	LATIN CAPITAL LETTER U WITH HORN
67	<u9>	<U01B0>	LATIN SMALL LETTER U WITH HORN
68	<Z//>	<U01B5>	LATIN CAPITAL LETTER Z WITH STROKE
69	<z//>	<U01B6>	LATIN SMALL LETTER Z WITH STROKE
70	<ED>	<U01B7>	LATIN CAPITAL LETTER EZH
71	<DZ<>	<U01C4>	LATIN CAPITAL LETTER DZ WITH CARON
72	<Dz<>	<U01C5>	LATIN CAPITAL LETTER D WITH SMALL LETTER Z WITH CARON
73	<dz<>	<U01C6>	LATIN SMALL LETTER DZ WITH CARON
74	<LJ3>	<U01C7>	LATIN CAPITAL LETTER LJ
75	<Lj3>	<U01C8>	LATIN CAPITAL LETTER L WITH SMALL LETTER J
76	<lj3>	<U01C9>	LATIN SMALL LETTER LJ
77	<NJ3>	<U01CA>	LATIN CAPITAL LETTER NJ
78	<Nj3>	<U01CB>	LATIN CAPITAL LETTER N WITH SMALL LETTER J
79	<nj3>	<U01CC>	LATIN SMALL LETTER NJ
80	<A<>	<U01CD>	LATIN CAPITAL LETTER A WITH CARON
81	<a<>	<U01CE>	LATIN SMALL LETTER A WITH CARON
82	<I<>	<U01CF>	LATIN CAPITAL LETTER I WITH CARON
83	<i<>	<U01D0>	LATIN SMALL LETTER I WITH CARON
84	<O<>	<U01D1>	LATIN CAPITAL LETTER O WITH CARON
85	<o<>	<U01D2>	LATIN SMALL LETTER O WITH CARON
86	<U<>	<U01D3>	LATIN CAPITAL LETTER U WITH CARON
87	<u<>	<U01D4>	LATIN SMALL LETTER U WITH CARON
88	<U: ->	<U01D5>	LATIN CAPITAL LETTER U WITH DIAERESIS AND MACRON
89	<u: ->	<U01D6>	LATIN SMALL LETTER U WITH DIAERESIS AND MACRON
90	<U: ' >	<U01D7>	LATIN CAPITAL LETTER U WITH DIAERESIS AND ACUTE
91	<u: ' >	<U01D8>	LATIN SMALL LETTER U WITH DIAERESIS AND ACUTE
92	<U: <>	<U01D9>	LATIN CAPITAL LETTER U WITH DIAERESIS AND CARON
93	<u: <>	<U01DA>	LATIN SMALL LETTER U WITH DIAERESIS AND CARON
94	<U: ! >	<U01DB>	LATIN CAPITAL LETTER U WITH DIAERESIS AND GRAVE
95	<u: ! >	<U01DC>	LATIN SMALL LETTER U WITH DIAERESIS AND GRAVE
96	<e1>	<U01DD>	LATIN SMALL LETTER TURNED E
97	<A1>	<U01DE>	LATIN CAPITAL LETTER A WITH DIAERESIS AND MACRON
98	<a1>	<U01DF>	LATIN SMALL LETTER A WITH DIAERESIS AND MACRON
99	<A7>	<U01E0>	LATIN CAPITAL LETTER A WITH DOT ABOVE AND MACRON
100	<a7>	<U01E1>	LATIN SMALL LETTER A WITH DOT ABOVE AND MACRON
101	<A3>	<U01E2>	LATIN CAPITAL LETTER AE WITH MACRON (ash)
102	<a3>	<U01E3>	LATIN SMALL LETTER AE WITH MACRON (ash)
103	<G//>	<U01E4>	LATIN CAPITAL LETTER G WITH STROKE
104	<g//>	<U01E5>	LATIN SMALL LETTER G WITH STROKE
105	<G<>	<U01E6>	LATIN CAPITAL LETTER G WITH CARON
106	<g<>	<U01E7>	LATIN SMALL LETTER G WITH CARON
107	<K<>	<U01E8>	LATIN CAPITAL LETTER K WITH CARON
108	<k<>	<U01E9>	LATIN SMALL LETTER K WITH CARON
109	<O; >	<U01EA>	LATIN CAPITAL LETTER O WITH OGONEK
110	<o; >	<U01EB>	LATIN SMALL LETTER O WITH OGONEK
111	<O1>	<U01EC>	LATIN CAPITAL LETTER O WITH OGONEK AND MACRON
112	<o1>	<U01ED>	LATIN SMALL LETTER O WITH OGONEK AND MACRON
113	<EZ>	<U01EE>	LATIN CAPITAL LETTER EZH WITH CARON
114	<ez>	<U01EF>	LATIN SMALL LETTER EZH WITH CARON
115	<j<>	<U01F0>	LATIN SMALL LETTER J WITH CARON
116	<DZ3>	<U01F1>	LATIN CAPITAL LETTER DZ
117	<Dz3>	<U01F2>	LATIN CAPITAL LETTER D WITH SMALL LETTER Z
118	<dz3>	<U01F3>	LATIN SMALL LETTER DZ
119	<G'>	<U01F4>	LATIN CAPITAL LETTER G WITH ACUTE
120	<g'>	<U01F5>	LATIN SMALL LETTER G WITH ACUTE
121	<AA'>	<U01FA>	LATIN CAPITAL LETTER A WITH RING ABOVE AND ACUTE
122	<aa'>	<U01FB>	LATIN SMALL LETTER A WITH RING ABOVE AND ACUTE
123	<AE'>	<U01FC>	LATIN CAPITAL LETTER AE WITH ACUTE (ash)
124	<ae'>	<U01FD>	LATIN SMALL LETTER AE WITH ACUTE (ash)
125	<O//'>	<U01FE>	LATIN CAPITAL LETTER O WITH STROKE AND ACUTE
126	<o//'>	<U01FF>	LATIN SMALL LETTER O WITH STROKE AND ACUTE
127	<A! ! >	<U0200>	LATIN CAPITAL LETTER A WITH DOUBLE GRAVE
128	<a! ! >	<U0201>	LATIN SMALL LETTER A WITH DOUBLE GRAVE
129	<A>	<U0202>	LATIN CAPITAL LETTER A WITH INVERTED BREVE

<a>	<U0203>	LATIN SMALL LETTER A WITH INVERTED BREVE
<E!!>	<U0204>	LATIN CAPITAL LETTER E WITH DOUBLE GRAVE
<e!!>	<U0205>	LATIN SMALL LETTER E WITH DOUBLE GRAVE
<E>	<U0206>	LATIN CAPITAL LETTER E WITH INVERTED BREVE
<e>	<U0207>	LATIN SMALL LETTER E WITH INVERTED BREVE
<I!!>	<U0208>	LATIN CAPITAL LETTER I WITH DOUBLE GRAVE
<i!!>	<U0209>	LATIN SMALL LETTER I WITH DOUBLE GRAVE
<I>	<U020A>	LATIN CAPITAL LETTER I WITH INVERTED BREVE
<i>	<U020B>	LATIN SMALL LETTER I WITH INVERTED BREVE
<O!!>	<U020C>	LATIN CAPITAL LETTER O WITH DOUBLE GRAVE
<o!!>	<U020D>	LATIN SMALL LETTER O WITH DOUBLE GRAVE
<O>	<U020E>	LATIN CAPITAL LETTER O WITH INVERTED BREVE
<o>	<U020F>	LATIN SMALL LETTER O WITH INVERTED BREVE
<R!!>	<U0210>	LATIN CAPITAL LETTER R WITH DOUBLE GRAVE
<r!!>	<U0211>	LATIN SMALL LETTER R WITH DOUBLE GRAVE
<R>	<U0212>	LATIN CAPITAL LETTER R WITH INVERTED BREVE
<r>	<U0213>	LATIN SMALL LETTER R WITH INVERTED BREVE
<U!!>	<U0214>	LATIN CAPITAL LETTER U WITH DOUBLE GRAVE
<u!!>	<U0215>	LATIN SMALL LETTER U WITH DOUBLE GRAVE
<U>	<U0216>	LATIN CAPITAL LETTER U WITH INVERTED BREVE
<u>	<U0217>	LATIN SMALL LETTER U WITH INVERTED BREVE
<r1>	<U027C>	LATIN SMALL LETTER R WITH LONG LEG
<ed>	<U0292>	LATIN SMALL LETTER EZH
< ;S>	<U02BB>	MODIFIER LETTER TURNED COMMA
<l />>	<U02C6>	MODIFIER LETTER CIRCUMFLEX ACCENT
<' <>	<U02C7>	CARON (Mandarin Chinese third tone)
<l ->	<U02C9>	MODIFIER LETTER MACRON (Mandarin Chinese first tone)
<l !>	<U02CB>	MODIFIER LETTER GRAVE ACCENT (Mandarin Chinese fourth tone)
<' (>	<U02D8>	BREVE
<' .>	<U02D9>	DOT ABOVE (Mandarin Chinese light tone)
<' 0>	<U02DA>	RING ABOVE
<' ;>	<U02DB>	OGONEK
<l ?>	<U02DC>	SMALL TILDE
<' " >	<U02DD>	DOUBLE ACUTE ACCENT
<' G>	<U0374>	GREEK NUMERAL SIGN (Dexia keraia)
<, G>	<U0375>	GREEK LOWER NUMERAL SIGN (Aristeri keraia)
<j3>	<U037A>	GREEK YPOGEGRAMMENI
<?%>	<U037E>	GREEK QUESTION MARK (Erotimatiko)
<' * >	<U0384>	GREEK TONOS
<' %>	<U0385>	GREEK DIALYTIKA TONOS
<A%>	<U0386>	GREEK CAPITAL LETTER ALPHA WITH TONOS
<. * >	<U0387>	GREEK ANO TELEIA
<E%>	<U0388>	GREEK CAPITAL LETTER EPSILON WITH TONOS
<Y%>	<U0389>	GREEK CAPITAL LETTER ETA WITH TONOS
<I%>	<U038A>	GREEK CAPITAL LETTER IOTA WITH TONOS
<O%>	<U038C>	GREEK CAPITAL LETTER OMICRON WITH TONOS
<U%>	<U038E>	GREEK CAPITAL LETTER UPSILON WITH TONOS
<W%>	<U038F>	GREEK CAPITAL LETTER OMEGA WITH TONOS
<i3>	<U0390>	GREEK SMALL LETTER IOTA WITH DIALYTIKA AND TONOS
<A* >	<U0391>	GREEK CAPITAL LETTER ALPHA
<B* >	<U0392>	GREEK CAPITAL LETTER BETA
<G* >	<U0393>	GREEK CAPITAL LETTER GAMMA
<D* >	<U0394>	GREEK CAPITAL LETTER DELTA
<E* >	<U0395>	GREEK CAPITAL LETTER EPSILON
<Z* >	<U0396>	GREEK CAPITAL LETTER ZETA
<Y* >	<U0397>	GREEK CAPITAL LETTER ETA
<H* >	<U0398>	GREEK CAPITAL LETTER THETA
<I* >	<U0399>	GREEK CAPITAL LETTER IOTA
<K* >	<U039A>	GREEK CAPITAL LETTER KAPPA
<L* >	<U039B>	GREEK CAPITAL LETTER LAMDA
<M* >	<U039C>	GREEK CAPITAL LETTER MU
<N* >	<U039D>	GREEK CAPITAL LETTER NU
<C* >	<U039E>	GREEK CAPITAL LETTER XI
<O* >	<U039F>	GREEK CAPITAL LETTER OMICRON
<P* >	<U03A0>	GREEK CAPITAL LETTER PI
<R* >	<U03A1>	GREEK CAPITAL LETTER RHO
<S* >	<U03A3>	GREEK CAPITAL LETTER SIGMA
<T* >	<U03A4>	GREEK CAPITAL LETTER TAU
<U* >	<U03A5>	GREEK CAPITAL LETTER UPSILON
<F* >	<U03A6>	GREEK CAPITAL LETTER PHI
<X* >	<U03A7>	GREEK CAPITAL LETTER CHI
<Q* >	<U03A8>	GREEK CAPITAL LETTER PSI
<W* >	<U03A9>	GREEK CAPITAL LETTER OMEGA
<J* >	<U03AA>	GREEK CAPITAL LETTER IOTA WITH DIALYTIKA
<V* >	<U03AB>	GREEK CAPITAL LETTER UPSILON WITH DIALYTIKA
<a%>	<U03AC>	GREEK SMALL LETTER ALPHA WITH TONOS
<e%>	<U03AD>	GREEK SMALL LETTER EPSILON WITH TONOS
<y%>	<U03AE>	GREEK SMALL LETTER ETA WITH TONOS
<i%>	<U03AF>	GREEK SMALL LETTER IOTA WITH TONOS
<u3>	<U03B0>	GREEK SMALL LETTER UPSILON WITH DIALYTIKA AND TONOS
<a* >	<U03B1>	GREEK SMALL LETTER ALPHA
<b* >	<U03B2>	GREEK SMALL LETTER BETA
<g* >	<U03B3>	GREEK SMALL LETTER GAMMA
<d* >	<U03B4>	GREEK SMALL LETTER DELTA
<e* >	<U03B5>	GREEK SMALL LETTER EPSILON
<z* >	<U03B6>	GREEK SMALL LETTER ZETA
<y* >	<U03B7>	GREEK SMALL LETTER ETA
<h* >	<U03B8>	GREEK SMALL LETTER THETA

44	<i*>	<U03B9>	GREEK SMALL LETTER IOTA
44	<k*>	<U03BA>	GREEK SMALL LETTER KAPPA
44	<l*>	<U03BB>	GREEK SMALL LETTER LAMDA
44	<m*>	<U03BC>	GREEK SMALL LETTER MU
44	<n*>	<U03BD>	GREEK SMALL LETTER NU
44	<c*>	<U03BE>	GREEK SMALL LETTER XI
44	<o*>	<U03BF>	GREEK SMALL LETTER OMICRON
44	<p*>	<U03C0>	GREEK SMALL LETTER PI
44	<r*>	<U03C1>	GREEK SMALL LETTER RHO
44	<*s>	<U03C2>	GREEK SMALL LETTER FINAL SIGMA
44	<s*>	<U03C3>	GREEK SMALL LETTER SIGMA
44	<t*>	<U03C4>	GREEK SMALL LETTER TAU
44	<u*>	<U03C5>	GREEK SMALL LETTER UPSILON
44	<f*>	<U03C6>	GREEK SMALL LETTER PHI
44	<x*>	<U03C7>	GREEK SMALL LETTER CHI
44	<q*>	<U03C8>	GREEK SMALL LETTER PSI
44	<w*>	<U03C9>	GREEK SMALL LETTER OMEGA
44	<j*>	<U03CA>	GREEK SMALL LETTER IOTA WITH DIALYTIKA
44	<v*>	<U03CB>	GREEK SMALL LETTER UPSILON WITH DIALYTIKA
44	<o%>	<U03CC>	GREEK SMALL LETTER OMICRON WITH TONOS
44	<u%>	<U03CD>	GREEK SMALL LETTER UPSILON WITH TONOS
44	<w%>	<U03CE>	GREEK SMALL LETTER OMEGA WITH TONOS
44	<b3>	<U03D0>	GREEK BETA SYMBOL
44	<T3>	<U03DA>	GREEK LETTER STIGMA
44	<M3>	<U03DC>	GREEK LETTER DIGAMMA
44	<K3>	<U03DE>	GREEK LETTER KOPPA
44	<P3>	<U03E0>	GREEK LETTER SAMPI
44	<IO>	<U0401>	CYRILLIC CAPITAL LETTER IO
44	<D%>	<U0402>	CYRILLIC CAPITAL LETTER DJE (Serbocroatian)
44	<G%>	<U0403>	CYRILLIC CAPITAL LETTER GJE
44	<IE>	<U0404>	CYRILLIC CAPITAL LETTER UKRAINIAN IE
44	<DS>	<U0405>	CYRILLIC CAPITAL LETTER DZE
44	<II>	<U0406>	CYRILLIC CAPITAL LETTER BYELORUSSIAN-UKRAINIAN I
44	<YI>	<U0407>	CYRILLIC CAPITAL LETTER YI (Ukrainian)
44	<J%>	<U0408>	CYRILLIC CAPITAL LETTER JE
44	<LJ>	<U0409>	CYRILLIC CAPITAL LETTER LJE
44	<NJ>	<U040A>	CYRILLIC CAPITAL LETTER NJE
44	<Ts>	<U040B>	CYRILLIC CAPITAL LETTER TSHE (Serbocroatian)
44	<KJ>	<U040C>	CYRILLIC CAPITAL LETTER KJE
44	<V%>	<U040E>	CYRILLIC CAPITAL LETTER SHORT U (Byelorussian)
44	<DZ>	<U040F>	CYRILLIC CAPITAL LETTER DZHE
44	<A=>	<U0410>	CYRILLIC CAPITAL LETTER A
44	<B=>	<U0411>	CYRILLIC CAPITAL LETTER BE
44	<V=>	<U0412>	CYRILLIC CAPITAL LETTER VE
44	<G=>	<U0413>	CYRILLIC CAPITAL LETTER GHE
44	<D=>	<U0414>	CYRILLIC CAPITAL LETTER DE
44	<E=>	<U0415>	CYRILLIC CAPITAL LETTER IE
44	<Z%>	<U0416>	CYRILLIC CAPITAL LETTER ZHE
44	<Z=>	<U0417>	CYRILLIC CAPITAL LETTER ZE
44	<I=>	<U0418>	CYRILLIC CAPITAL LETTER I
44	<J=>	<U0419>	CYRILLIC CAPITAL LETTER SHORT I
44	<K=>	<U041A>	CYRILLIC CAPITAL LETTER KA
44	<L=>	<U041B>	CYRILLIC CAPITAL LETTER EL
44	<M=>	<U041C>	CYRILLIC CAPITAL LETTER EM
44	<N=>	<U041D>	CYRILLIC CAPITAL LETTER EN
44	<O=>	<U041E>	CYRILLIC CAPITAL LETTER O
44	<P=>	<U041F>	CYRILLIC CAPITAL LETTER PE
44	<R=>	<U0420>	CYRILLIC CAPITAL LETTER ER
44	<S=>	<U0421>	CYRILLIC CAPITAL LETTER ES
44	<T=>	<U0422>	CYRILLIC CAPITAL LETTER TE
44	<U=>	<U0423>	CYRILLIC CAPITAL LETTER U
44	<F=>	<U0424>	CYRILLIC CAPITAL LETTER EF
44	<H=>	<U0425>	CYRILLIC CAPITAL LETTER HA
44	<C=>	<U0426>	CYRILLIC CAPITAL LETTER TSE
44	<C%>	<U0427>	CYRILLIC CAPITAL LETTER CHE
44	<S%>	<U0428>	CYRILLIC CAPITAL LETTER SHA
44	<Sc>	<U0429>	CYRILLIC CAPITAL LETTER SHCHA
44	<=">	<U042A>	CYRILLIC CAPITAL LETTER HARD SIGN
44	<Y=>	<U042B>	CYRILLIC CAPITAL LETTER YERU
44	<% ">	<U042C>	CYRILLIC CAPITAL LETTER SOFT SIGN
44	<JE>	<U042D>	CYRILLIC CAPITAL LETTER E
44	<JU>	<U042E>	CYRILLIC CAPITAL LETTER YU
44	<JA>	<U042F>	CYRILLIC CAPITAL LETTER YA
44	<a=>	<U0430>	CYRILLIC SMALL LETTER A
44	<b=>	<U0431>	CYRILLIC SMALL LETTER BE
44	<v=>	<U0432>	CYRILLIC SMALL LETTER VE
44	<g=>	<U0433>	CYRILLIC SMALL LETTER GHE
44	<d=>	<U0434>	CYRILLIC SMALL LETTER DE
44	<e=>	<U0435>	CYRILLIC SMALL LETTER IE
44	<z%>	<U0436>	CYRILLIC SMALL LETTER ZHE
44	<z=>	<U0437>	CYRILLIC SMALL LETTER ZE
44	<i=>	<U0438>	CYRILLIC SMALL LETTER I
44	<j=>	<U0439>	CYRILLIC SMALL LETTER SHORT I
44	<k=>	<U043A>	CYRILLIC SMALL LETTER KA
44	<l=>	<U043B>	CYRILLIC SMALL LETTER EL
44	<m=>	<U043C>	CYRILLIC SMALL LETTER EM
44	<n=>	<U043D>	CYRILLIC SMALL LETTER EN
44	<o=>	<U043E>	CYRILLIC SMALL LETTER O

<p=>	<U043F>	CYRILLIC SMALL LETTER PE
<r=>	<U0440>	CYRILLIC SMALL LETTER ER
<s=>	<U0441>	CYRILLIC SMALL LETTER ES
<t=>	<U0442>	CYRILLIC SMALL LETTER TE
<u=>	<U0443>	CYRILLIC SMALL LETTER U
<f=>	<U0444>	CYRILLIC SMALL LETTER EF
<h=>	<U0445>	CYRILLIC SMALL LETTER HA
<c=>	<U0446>	CYRILLIC SMALL LETTER TSE
<c%>	<U0447>	CYRILLIC SMALL LETTER CHE
<s%>	<U0448>	CYRILLIC SMALL LETTER SHA
<sc>	<U0449>	CYRILLIC SMALL LETTER SHCHA
<='>	<U044A>	CYRILLIC SMALL LETTER HARD SIGN
<y=>	<U044B>	CYRILLIC SMALL LETTER YERU
<% '>	<U044C>	CYRILLIC SMALL LETTER SOFT SIGN
<je>	<U044D>	CYRILLIC SMALL LETTER E
<ju>	<U044E>	CYRILLIC SMALL LETTER YU
<ja>	<U044F>	CYRILLIC SMALL LETTER YA
<io>	<U0451>	CYRILLIC SMALL LETTER IO
<d%>	<U0452>	CYRILLIC SMALL LETTER DJE (Serbocroatian)
<g%>	<U0453>	CYRILLIC SMALL LETTER GJE
<ie>	<U0454>	CYRILLIC SMALL LETTER UKRAINIAN IE
<ds>	<U0455>	CYRILLIC SMALL LETTER DZE
<ii>	<U0456>	CYRILLIC SMALL LETTER BYELORUSSIAN-UKRAINIAN I
<yi>	<U0457>	CYRILLIC SMALL LETTER YI (Ukrainian)
<j%>	<U0458>	CYRILLIC SMALL LETTER JE
<lj>	<U0459>	CYRILLIC SMALL LETTER LJE
<nj>	<U045A>	CYRILLIC SMALL LETTER NJE
<ts>	<U045B>	CYRILLIC SMALL LETTER TSHE (Serbocroatian)
<kj>	<U045C>	CYRILLIC SMALL LETTER KJE
<v%>	<U045E>	CYRILLIC SMALL LETTER SHORT U (Byelorussian)
<dz>	<U045F>	CYRILLIC SMALL LETTER DZHE
<Y3>	<U0462>	CYRILLIC CAPITAL LETTER YAT
<y3>	<U0463>	CYRILLIC SMALL LETTER YAT
<O3>	<U046A>	CYRILLIC CAPITAL LETTER BIG YUS
<o3>	<U046B>	CYRILLIC SMALL LETTER BIG YUS
<F3>	<U0472>	CYRILLIC CAPITAL LETTER FITA
<f3>	<U0473>	CYRILLIC SMALL LETTER FITA
<V3>	<U0474>	CYRILLIC CAPITAL LETTER IZHITSA
<v3>	<U0475>	CYRILLIC SMALL LETTER IZHITSA
<C3>	<U0480>	CYRILLIC CAPITAL LETTER KOPPA
<c3>	<U0481>	CYRILLIC SMALL LETTER KOPPA
<G3>	<U0490>	CYRILLIC CAPITAL LETTER GHE WITH UPTURN
<g3>	<U0491>	CYRILLIC SMALL LETTER GHE WITH UPTURN
<A+>	<U05D0>	HEBREW LETTER ALEF
<B+>	<U05D1>	HEBREW LETTER BET
<G+>	<U05D2>	HEBREW LETTER GIMEL
<D+>	<U05D3>	HEBREW LETTER DALET
<H+>	<U05D4>	HEBREW LETTER HE
<W+>	<U05D5>	HEBREW LETTER VAV
<Z+>	<U05D6>	HEBREW LETTER ZAYIN
<X+>	<U05D7>	HEBREW LETTER HET
<Tj>	<U05D8>	HEBREW LETTER TET
<J+>	<U05D9>	HEBREW LETTER YOD
<K%>	<U05DA>	HEBREW LETTER FINAL KAF
<K+>	<U05DB>	HEBREW LETTER KAF
<L+>	<U05DC>	HEBREW LETTER LAMED
<M%>	<U05DD>	HEBREW LETTER FINAL MEM
<M+>	<U05DE>	HEBREW LETTER MEM
<N%>	<U05DF>	HEBREW LETTER FINAL NUN
<N+>	<U05E0>	HEBREW LETTER NUN
<S+>	<U05E1>	HEBREW LETTER SAMEKH
<E+>	<U05E2>	HEBREW LETTER AYIN
<P%>	<U05E3>	HEBREW LETTER FINAL PE
<P+>	<U05E4>	HEBREW LETTER PE
<Zj>	<U05E5>	HEBREW LETTER FINAL TSADI
<ZJ>	<U05E6>	HEBREW LETTER TSADI
<Q+>	<U05E7>	HEBREW LETTER QOF
<R+>	<U05E8>	HEBREW LETTER RESH
<Sh>	<U05E9>	HEBREW LETTER SHIN
<T+>	<U05EA>	HEBREW LETTER TAV
<,+>	<U060C>	ARABIC COMMA
< ;+>	<U061B>	ARABIC SEMICOLON
<?+>	<U061F>	ARABIC QUESTION MARK
<H'>	<U0621>	ARABIC LETTER HAMZA
<aM>	<U0622>	ARABIC LETTER ALEF WITH MADDA ABOVE
<aH>	<U0623>	ARABIC LETTER ALEF WITH HAMZA ABOVE
<wH>	<U0624>	ARABIC LETTER WAW WITH HAMZA ABOVE
<aH>	<U0625>	ARABIC LETTER ALEF WITH HAMZA BELOW
<yH>	<U0626>	ARABIC LETTER YEH WITH HAMZA ABOVE
<a+>	<U0627>	ARABIC LETTER ALEF
<b+>	<U0628>	ARABIC LETTER BEH
<tm>	<U0629>	ARABIC LETTER TEH MARBUTA
<t+>	<U062A>	ARABIC LETTER TEH
<tk>	<U062B>	ARABIC LETTER THEH
<g+>	<U062C>	ARABIC LETTER JEEM
<hk>	<U062D>	ARABIC LETTER HAH
<x+>	<U062E>	ARABIC LETTER KHAH
<d+>	<U062F>	ARABIC LETTER DAL

<dk>	<U0630>	ARABIC LETTER THAL
<r+>	<U0631>	ARABIC LETTER REH
<z+>	<U0632>	ARABIC LETTER ZAIN
<s+>	<U0633>	ARABIC LETTER SEEN
<sn>	<U0634>	ARABIC LETTER SHEEN
<c+>	<U0635>	ARABIC LETTER SAD
<dd>	<U0636>	ARABIC LETTER DAD
<tj>	<U0637>	ARABIC LETTER TAH
<zH>	<U0638>	ARABIC LETTER ZAH
<e+>	<U0639>	ARABIC LETTER AIN
<i+>	<U063A>	ARABIC LETTER GHAIN
<+>	<U0640>	ARABIC TATWEEL
<f+>	<U0641>	ARABIC LETTER FEH
<q+>	<U0642>	ARABIC LETTER QAF
<k+>	<U0643>	ARABIC LETTER KAF
<l+>	<U0644>	ARABIC LETTER LAM
<m+>	<U0645>	ARABIC LETTER MEEM
<n+>	<U0646>	ARABIC LETTER NOON
<h+>	<U0647>	ARABIC LETTER HEH
<w+>	<U0648>	ARABIC LETTER WAW
<j+>	<U0649>	ARABIC LETTER ALEF MAKSURA
<y+>	<U064A>	ARABIC LETTER YEH
<:+>	<U064B>	ARABIC FATHATAN
<" +>	<U064C>	ARABIC DAMMATAN
<=+>	<U064D>	ARABIC KASRATAN
</ +>	<U064E>	ARABIC FATHA
<' +>	<U064F>	ARABIC DAMMA
<1+>	<U0650>	ARABIC KASRA
<3+>	<U0651>	ARABIC SHADDA
<0+>	<U0652>	ARABIC SUKUN
<0a>	<U0660>	ARABIC-INDIC DIGIT ZERO
<1a>	<U0661>	ARABIC-INDIC DIGIT ONE
<2a>	<U0662>	ARABIC-INDIC DIGIT TWO
<3a>	<U0663>	ARABIC-INDIC DIGIT THREE
<4a>	<U0664>	ARABIC-INDIC DIGIT FOUR
<5a>	<U0665>	ARABIC-INDIC DIGIT FIVE
<6a>	<U0666>	ARABIC-INDIC DIGIT SIX
<7a>	<U0667>	ARABIC-INDIC DIGIT SEVEN
<8a>	<U0668>	ARABIC-INDIC DIGIT EIGHT
<9a>	<U0669>	ARABIC-INDIC DIGIT NINE
<aS>	<U0670>	ARABIC LETTER SUPERSCRIPIT ALEF
<p+>	<U067E>	ARABIC LETTER PEH
<hH>	<U0681>	ARABIC LETTER HAH WITH HAMZA ABOVE
<tc>	<U0686>	ARABIC LETTER TCHEH
<zj>	<U0698>	ARABIC LETTER JEH
<v+>	<U06A4>	ARABIC LETTER VEH
<gf>	<U06AF>	ARABIC LETTER GAF
<A-0>	<U1E00>	LATIN CAPITAL LETTER A WITH RING BELOW
<a-0>	<U1E01>	LATIN SMALL LETTER A WITH RING BELOW
<B.>	<U1E02>	LATIN CAPITAL LETTER B WITH DOT ABOVE
<b.>	<U1E03>	LATIN SMALL LETTER B WITH DOT ABOVE
<B-.>	<U1E04>	LATIN CAPITAL LETTER B WITH DOT BELOW
<b-.>	<U1E05>	LATIN SMALL LETTER B WITH DOT BELOW
<B_>	<U1E06>	LATIN CAPITAL LETTER B WITH LINE BELOW
<b_>	<U1E07>	LATIN SMALL LETTER B WITH LINE BELOW
<C,'>	<U1E08>	LATIN CAPITAL LETTER C WITH CEDILLA AND ACUTE
<c,'>	<U1E09>	LATIN SMALL LETTER C WITH CEDILLA AND ACUTE
<D.>	<U1E0A>	LATIN CAPITAL LETTER D WITH DOT ABOVE
<d.>	<U1E0B>	LATIN SMALL LETTER D WITH DOT ABOVE
<D-.>	<U1E0C>	LATIN CAPITAL LETTER D WITH DOT BELOW
<d-.>	<U1E0D>	LATIN SMALL LETTER D WITH DOT BELOW
<D_>	<U1E0E>	LATIN CAPITAL LETTER D WITH LINE BELOW
<d_>	<U1E0F>	LATIN SMALL LETTER D WITH LINE BELOW
<D.>	<U1E10>	LATIN CAPITAL LETTER D WITH CEDILLA
<d.>	<U1E11>	LATIN SMALL LETTER D WITH CEDILLA
<D- />>	<U1E12>	LATIN CAPITAL LETTER D WITH CIRCUMFLEX BELOW
<d- />>	<U1E13>	LATIN SMALL LETTER D WITH CIRCUMFLEX BELOW
<E-!>	<U1E14>	LATIN CAPITAL LETTER E WITH MACRON AND GRAVE
<e-!>	<U1E15>	LATIN SMALL LETTER E WITH MACRON AND GRAVE
<E-'>	<U1E16>	LATIN CAPITAL LETTER E WITH MACRON AND ACUTE
<e-'>	<U1E17>	LATIN SMALL LETTER E WITH MACRON AND ACUTE
<E- />>	<U1E18>	LATIN CAPITAL LETTER E WITH CIRCUMFLEX BELOW
<e- />>	<U1E19>	LATIN SMALL LETTER E WITH CIRCUMFLEX BELOW
<E-?>	<U1E1A>	LATIN CAPITAL LETTER E WITH TILDE BELOW
<e-?>	<U1E1B>	LATIN SMALL LETTER E WITH TILDE BELOW
<E,(>	<U1E1C>	LATIN CAPITAL LETTER E WITH CEDILLA AND BREVE
<e,(>	<U1E1D>	LATIN SMALL LETTER E WITH CEDILLA AND BREVE
<F.>	<U1E1E>	LATIN CAPITAL LETTER F WITH DOT ABOVE
<f.>	<U1E1F>	LATIN SMALL LETTER F WITH DOT ABOVE
<G->	<U1E20>	LATIN CAPITAL LETTER G WITH MACRON
<g->	<U1E21>	LATIN SMALL LETTER G WITH MACRON
<H.>	<U1E22>	LATIN CAPITAL LETTER H WITH DOT ABOVE
<h.>	<U1E23>	LATIN SMALL LETTER H WITH DOT ABOVE
<H-.>	<U1E24>	LATIN CAPITAL LETTER H WITH DOT BELOW
<h-.>	<U1E25>	LATIN SMALL LETTER H WITH DOT BELOW
<H:>	<U1E26>	LATIN CAPITAL LETTER H WITH DIAERESIS
<h:>	<U1E27>	LATIN SMALL LETTER H WITH DIAERESIS
<H,>	<U1E28>	LATIN CAPITAL LETTER H WITH CEDILLA

4672	<h,>	<U1E29>	LATIN SMALL LETTER H WITH CEDILLA
4673	<H-(>	<U1E2A>	LATIN CAPITAL LETTER H WITH BREVE BELOW
4674	<h->	<U1E2B>	LATIN SMALL LETTER H WITH BREVE BELOW
4675	<I-?>	<U1E2C>	LATIN CAPITAL LETTER I WITH TILDE BELOW
4676	<i-?>	<U1E2D>	LATIN SMALL LETTER I WITH TILDE BELOW
4677	<I:'>	<U1E2E>	LATIN CAPITAL LETTER I WITH DIAERESIS AND ACUTE
4678	<i:'>	<U1E2F>	LATIN SMALL LETTER I WITH DIAERESIS AND ACUTE
4679	<K'>	<U1E30>	LATIN CAPITAL LETTER K WITH ACUTE
4680	<k'>	<U1E31>	LATIN SMALL LETTER K WITH ACUTE
4681	<K-.>	<U1E32>	LATIN CAPITAL LETTER K WITH DOT BELOW
4682	<k-.>	<U1E33>	LATIN SMALL LETTER K WITH DOT BELOW
4683	<K.>	<U1E34>	LATIN CAPITAL LETTER K WITH LINE BELOW
4684	<k.>	<U1E35>	LATIN SMALL LETTER K WITH LINE BELOW
4685	<L-.>	<U1E36>	LATIN CAPITAL LETTER L WITH DOT BELOW
4686	<l-.>	<U1E37>	LATIN SMALL LETTER L WITH DOT BELOW
4687	<L--.>	<U1E38>	LATIN CAPITAL LETTER L WITH DOT BELOW AND MACRON
4688	<l--.>	<U1E39>	LATIN SMALL LETTER L WITH DOT BELOW AND MACRON
4689	<L.>	<U1E3A>	LATIN CAPITAL LETTER L WITH LINE BELOW
4690	<l.>	<U1E3B>	LATIN SMALL LETTER L WITH LINE BELOW
4691	<L-/>>	<U1E3C>	LATIN CAPITAL LETTER L WITH CIRCUMFLEX BELOW
4692	<l-/>>	<U1E3D>	LATIN SMALL LETTER L WITH CIRCUMFLEX BELOW
4693	<M'>	<U1E3E>	LATIN CAPITAL LETTER M WITH ACUTE
4694	<m'>	<U1E3F>	LATIN SMALL LETTER M WITH ACUTE
4695	<M.>	<U1E40>	LATIN CAPITAL LETTER M WITH DOT ABOVE
4696	<m.>	<U1E41>	LATIN SMALL LETTER M WITH DOT ABOVE
4697	<M-.>	<U1E42>	LATIN CAPITAL LETTER M WITH DOT BELOW
4698	<m-.>	<U1E43>	LATIN SMALL LETTER M WITH DOT BELOW
4699	<N.>	<U1E44>	LATIN CAPITAL LETTER N WITH DOT ABOVE
4700	<n.>	<U1E45>	LATIN SMALL LETTER N WITH DOT ABOVE
4701	<N-.>	<U1E46>	LATIN CAPITAL LETTER N WITH DOT BELOW
4702	<n-.>	<U1E47>	LATIN SMALL LETTER N WITH DOT BELOW
4703	<N.>	<U1E48>	LATIN CAPITAL LETTER N WITH LINE BELOW
4704	<n.>	<U1E49>	LATIN SMALL LETTER N WITH LINE BELOW
4705	<N-/>>	<U1E4A>	LATIN CAPITAL LETTER N WITH CIRCUMFLEX BELOW
4706	<n-/>>	<U1E4B>	LATIN SMALL LETTER N WITH CIRCUMFLEX BELOW
4707	<O?>'>	<U1E4C>	LATIN CAPITAL LETTER O WITH TILDE AND ACUTE
4708	<o?>'>	<U1E4D>	LATIN SMALL LETTER O WITH TILDE AND ACUTE
4709	<O?:>'>	<U1E4E>	LATIN CAPITAL LETTER O WITH TILDE AND DIAERESIS
4710	<o?:>'>	<U1E4F>	LATIN SMALL LETTER O WITH TILDE AND DIAERESIS
4711	<O-!>	<U1E50>	LATIN CAPITAL LETTER O WITH MACRON AND GRAVE
4712	<o-!>	<U1E51>	LATIN SMALL LETTER O WITH MACRON AND GRAVE
4713	<O-'>	<U1E52>	LATIN CAPITAL LETTER O WITH MACRON AND ACUTE
4714	<o-'>	<U1E53>	LATIN SMALL LETTER O WITH MACRON AND ACUTE
4715	<P'>	<U1E54>	LATIN CAPITAL LETTER P WITH ACUTE
4716	<p'>	<U1E55>	LATIN SMALL LETTER P WITH ACUTE
4717	<P.>	<U1E56>	LATIN CAPITAL LETTER P WITH DOT ABOVE
4718	<p.>	<U1E57>	LATIN SMALL LETTER P WITH DOT ABOVE
4719	<R.>	<U1E58>	LATIN CAPITAL LETTER R WITH DOT ABOVE
4720	<r.>	<U1E59>	LATIN SMALL LETTER R WITH DOT ABOVE
4721	<R-.>	<U1E5A>	LATIN CAPITAL LETTER R WITH DOT BELOW
4722	<r-.>	<U1E5B>	LATIN SMALL LETTER R WITH DOT BELOW
4723	<R--.>	<U1E5C>	LATIN CAPITAL LETTER R WITH DOT BELOW AND MACRON
4724	<r--.>	<U1E5D>	LATIN SMALL LETTER R WITH DOT BELOW AND MACRON
4725	<R.>	<U1E5E>	LATIN CAPITAL LETTER R WITH LINE BELOW
4726	<r.>	<U1E5F>	LATIN SMALL LETTER R WITH LINE BELOW
4727	<S.>	<U1E60>	LATIN CAPITAL LETTER S WITH DOT ABOVE
4728	<s.>	<U1E61>	LATIN SMALL LETTER S WITH DOT ABOVE
4729	<S-.>	<U1E62>	LATIN CAPITAL LETTER S WITH DOT BELOW
4730	<s-.>	<U1E63>	LATIN SMALL LETTER S WITH DOT BELOW
4731	<S'>.>	<U1E64>	LATIN CAPITAL LETTER S WITH ACUTE AND DOT ABOVE
4732	<s'>.>	<U1E65>	LATIN SMALL LETTER S WITH ACUTE AND DOT ABOVE
4733	<S<.>	<U1E66>	LATIN CAPITAL LETTER S WITH CARON AND DOT ABOVE
4734	<s<.>	<U1E67>	LATIN SMALL LETTER S WITH CARON AND DOT ABOVE
4735	<S-.->	<U1E68>	LATIN CAPITAL LETTER S WITH DOT BELOW AND DOT ABOVE
4736	<s-.->	<U1E69>	LATIN SMALL LETTER S WITH DOT BELOW AND DOT ABOVE
4737	<T.>	<U1E6A>	LATIN CAPITAL LETTER T WITH DOT ABOVE
4738	<t.>	<U1E6B>	LATIN SMALL LETTER T WITH DOT ABOVE
4739	<T-.>	<U1E6C>	LATIN CAPITAL LETTER T WITH DOT BELOW
4740	<t-.>	<U1E6D>	LATIN SMALL LETTER T WITH DOT BELOW
4741	<T.>	<U1E6E>	LATIN CAPITAL LETTER T WITH LINE BELOW
4742	<t.>	<U1E6F>	LATIN SMALL LETTER T WITH LINE BELOW
4743	<T-/>>	<U1E70>	LATIN CAPITAL LETTER T WITH CIRCUMFLEX BELOW
4744	<t-/>>	<U1E71>	LATIN SMALL LETTER T WITH CIRCUMFLEX BELOW
4745	<U--:>	<U1E72>	LATIN CAPITAL LETTER U WITH DIAERESIS BELOW
4746	<u--:>	<U1E73>	LATIN SMALL LETTER U WITH DIAERESIS BELOW
4747	<U-?>	<U1E74>	LATIN CAPITAL LETTER U WITH TILDE BELOW
4748	<u-?>	<U1E75>	LATIN SMALL LETTER U WITH TILDE BELOW
4749	<U-/>>	<U1E76>	LATIN CAPITAL LETTER U WITH CIRCUMFLEX BELOW
4750	<u-/>>	<U1E77>	LATIN SMALL LETTER U WITH CIRCUMFLEX BELOW
4751	<U?>'>	<U1E78>	LATIN CAPITAL LETTER U WITH TILDE AND ACUTE
4752	<u?>'>	<U1E79>	LATIN SMALL LETTER U WITH TILDE AND ACUTE
4753	<U-:>	<U1E7A>	LATIN CAPITAL LETTER U WITH MACRON AND DIAERESIS
4754	<u-:>	<U1E7B>	LATIN SMALL LETTER U WITH MACRON AND DIAERESIS
4755	<V?>	<U1E7C>	LATIN CAPITAL LETTER V WITH TILDE
4756	<v?>	<U1E7D>	LATIN SMALL LETTER V WITH TILDE
4757	<V-.>	<U1E7E>	LATIN CAPITAL LETTER V WITH DOT BELOW
4758	<v-.>	<U1E7F>	LATIN SMALL LETTER V WITH DOT BELOW
4759	<W!>	<U1E80>	LATIN CAPITAL LETTER W WITH GRAVE

4760	<w!>	<U1E81>	LATIN SMALL LETTER W WITH GRAVE
4761	<W'>	<U1E82>	LATIN CAPITAL LETTER W WITH ACUTE
4762	<w'>	<U1E83>	LATIN SMALL LETTER W WITH ACUTE
4763	<W:>	<U1E84>	LATIN CAPITAL LETTER W WITH DIAERESIS
4764	<w:>	<U1E85>	LATIN SMALL LETTER W WITH DIAERESIS
4765	<W.>	<U1E86>	LATIN CAPITAL LETTER W WITH DOT ABOVE
4766	<w.>	<U1E87>	LATIN SMALL LETTER W WITH DOT ABOVE
4767	<W-.>	<U1E88>	LATIN CAPITAL LETTER W WITH DOT BELOW
4768	<w-.>	<U1E89>	LATIN SMALL LETTER W WITH DOT BELOW
4769	<X.>	<U1E8A>	LATIN CAPITAL LETTER X WITH DOT ABOVE
4770	<x.>	<U1E8B>	LATIN SMALL LETTER X WITH DOT ABOVE
4771	<X:>	<U1E8C>	LATIN CAPITAL LETTER X WITH DIAERESIS
4772	<x:>	<U1E8D>	LATIN SMALL LETTER X WITH DIAERESIS
4773	<Y.>	<U1E8E>	LATIN CAPITAL LETTER Y WITH DOT ABOVE
4774	<y.>	<U1E8F>	LATIN SMALL LETTER Y WITH DOT ABOVE
4775	<Z/>>	<U1E90>	LATIN CAPITAL LETTER Z WITH CIRCUMFLEX
4776	<z/>>	<U1E91>	LATIN SMALL LETTER Z WITH CIRCUMFLEX
4777	<Z-.>	<U1E92>	LATIN CAPITAL LETTER Z WITH DOT BELOW
4778	<z-.>	<U1E93>	LATIN SMALL LETTER Z WITH DOT BELOW
4779	<Z_>	<U1E94>	LATIN CAPITAL LETTER Z WITH LINE BELOW
4780	<z_>	<U1E95>	LATIN SMALL LETTER Z WITH LINE BELOW
4781	<A-.>	<U1EA0>	LATIN CAPITAL LETTER A WITH DOT BELOW
4782	<a-.>	<U1EA1>	LATIN SMALL LETTER A WITH DOT BELOW
4783	<A2>	<U1EA2>	LATIN CAPITAL LETTER A WITH HOOK ABOVE
4784	<a2>	<U1EA3>	LATIN SMALL LETTER A WITH HOOK ABOVE
4785	<A/>'>	<U1EA4>	LATIN CAPITAL LETTER A WITH CIRCUMFLEX AND ACUTE
4786	<a/>'>	<U1EA5>	LATIN SMALL LETTER A WITH CIRCUMFLEX AND ACUTE
4787	<A/>!>	<U1EA6>	LATIN CAPITAL LETTER A WITH CIRCUMFLEX AND GRAVE
4788	<a/>!>	<U1EA7>	LATIN SMALL LETTER A WITH CIRCUMFLEX AND GRAVE
4789	<A/>2>	<U1EA8>	LATIN CAPITAL LETTER A WITH CIRCUMFLEX AND HOOK ABOVE
4790	<a/>2>	<U1EA9>	LATIN SMALL LETTER A WITH CIRCUMFLEX AND HOOK ABOVE
4791	<A/>?>	<U1EAA>	LATIN CAPITAL LETTER A WITH CIRCUMFLEX AND TILDE
4792	<a/>?>	<U1EAB>	LATIN SMALL LETTER A WITH CIRCUMFLEX AND TILDE
4793	<A/>-.>	<U1EAC>	LATIN CAPITAL LETTER A WITH CIRCUMFLEX AND DOT BELOW
4794	<a/>-.>	<U1EAD>	LATIN SMALL LETTER A WITH CIRCUMFLEX AND DOT BELOW
4795	<A'>	<U1EAE>	LATIN CAPITAL LETTER A WITH BREVE AND ACUTE
4796	<a'>	<U1EAF>	LATIN SMALL LETTER A WITH BREVE AND ACUTE
4797	<A!>	<U1EB0>	LATIN CAPITAL LETTER A WITH BREVE AND GRAVE
4798	<a!>	<U1EB1>	LATIN SMALL LETTER A WITH BREVE AND GRAVE
4799	<A(2)>	<U1EB2>	LATIN CAPITAL LETTER A WITH BREVE AND HOOK ABOVE
4800	<a(2)>	<U1EB3>	LATIN SMALL LETTER A WITH BREVE AND HOOK ABOVE
4801	<A(?)>	<U1EB4>	LATIN CAPITAL LETTER A WITH BREVE AND TILDE
4802	<a(?)>	<U1EB5>	LATIN SMALL LETTER A WITH BREVE AND TILDE
4803	<A(-.>	<U1EB6>	LATIN CAPITAL LETTER A WITH BREVE AND DOT BELOW
4804	<a(-.>	<U1EB7>	LATIN SMALL LETTER A WITH BREVE AND DOT BELOW
4805	<E-.>	<U1EB8>	LATIN CAPITAL LETTER E WITH DOT BELOW
4806	<e-.>	<U1EB9>	LATIN SMALL LETTER E WITH DOT BELOW
4807	<E2>	<U1EBA>	LATIN CAPITAL LETTER E WITH HOOK ABOVE
4808	<e2>	<U1EBB>	LATIN SMALL LETTER E WITH HOOK ABOVE
4809	<E?>	<U1EBC>	LATIN CAPITAL LETTER E WITH TILDE
4810	<e?>	<U1EBD>	LATIN SMALL LETTER E WITH TILDE
4811	<E/>'>	<U1EBE>	LATIN CAPITAL LETTER E WITH CIRCUMFLEX AND ACUTE
4812	<e/>'>	<U1EBF>	LATIN SMALL LETTER E WITH CIRCUMFLEX AND ACUTE
4813	<E/>!>	<U1EC0>	LATIN CAPITAL LETTER E WITH CIRCUMFLEX AND GRAVE
4814	<e/>!>	<U1EC1>	LATIN SMALL LETTER E WITH CIRCUMFLEX AND GRAVE
4815	<E/>2>	<U1EC2>	LATIN CAPITAL LETTER E WITH CIRCUMFLEX AND HOOK ABOVE
4816	<e/>2>	<U1EC3>	LATIN SMALL LETTER E WITH CIRCUMFLEX AND HOOK ABOVE
4817	<E/>?>	<U1EC4>	LATIN CAPITAL LETTER E WITH CIRCUMFLEX AND TILDE
4818	<e/>?>	<U1EC5>	LATIN SMALL LETTER E WITH CIRCUMFLEX AND TILDE
4819	<E/>-.>	<U1EC6>	LATIN CAPITAL LETTER E WITH CIRCUMFLEX AND DOT BELOW
4820	<e/>-.>	<U1EC7>	LATIN SMALL LETTER E WITH CIRCUMFLEX AND DOT BELOW
4821	<I2>	<U1EC8>	LATIN CAPITAL LETTER I WITH HOOK ABOVE
4822	<i2>	<U1EC9>	LATIN SMALL LETTER I WITH HOOK ABOVE
4823	<I-.>	<U1ECA>	LATIN CAPITAL LETTER I WITH DOT BELOW
4824	<i-.>	<U1ECB>	LATIN SMALL LETTER I WITH DOT BELOW
4825	<O-.>	<U1ECC>	LATIN CAPITAL LETTER O WITH DOT BELOW
4826	<o-.>	<U1ECD>	LATIN SMALL LETTER O WITH DOT BELOW
4827	<O2>	<U1ECE>	LATIN CAPITAL LETTER O WITH HOOK ABOVE
4828	<o2>	<U1ECF>	LATIN SMALL LETTER O WITH HOOK ABOVE
4829	<O/>'>	<U1ED0>	LATIN CAPITAL LETTER O WITH CIRCUMFLEX AND ACUTE
4830	<o/>'>	<U1ED1>	LATIN SMALL LETTER O WITH CIRCUMFLEX AND ACUTE
4831	<O/>!>	<U1ED2>	LATIN CAPITAL LETTER O WITH CIRCUMFLEX AND GRAVE
4832	<o/>!>	<U1ED3>	LATIN SMALL LETTER O WITH CIRCUMFLEX AND GRAVE
4833	<O/>2>	<U1ED4>	LATIN CAPITAL LETTER O WITH CIRCUMFLEX AND HOOK ABOVE
4834	<o/>2>	<U1ED5>	LATIN SMALL LETTER O WITH CIRCUMFLEX AND HOOK ABOVE
4835	<O/>?>	<U1ED6>	LATIN CAPITAL LETTER O WITH CIRCUMFLEX AND TILDE
4836	<o/>?>	<U1ED7>	LATIN SMALL LETTER O WITH CIRCUMFLEX AND TILDE
4837	<O/>-.>	<U1ED8>	LATIN CAPITAL LETTER O WITH CIRCUMFLEX AND DOT BELOW
4838	<o/>-.>	<U1ED9>	LATIN SMALL LETTER O WITH CIRCUMFLEX AND DOT BELOW
4839	<O9'>	<U1EDA>	LATIN CAPITAL LETTER O WITH HORN AND ACUTE
4840	<o9'>	<U1EDB>	LATIN SMALL LETTER O WITH HORN AND ACUTE
4841	<O9!>	<U1EDC>	LATIN CAPITAL LETTER O WITH HORN AND GRAVE
4842	<o9!>	<U1EDD>	LATIN SMALL LETTER O WITH HORN AND GRAVE
4843	<O92>	<U1EDE>	LATIN CAPITAL LETTER O WITH HORN AND HOOK ABOVE
4844	<o92>	<U1EDF>	LATIN SMALL LETTER O WITH HORN AND HOOK ABOVE
4845	<O9?>	<U1EE0>	LATIN CAPITAL LETTER O WITH HORN AND TILDE
4846	<o9?>	<U1EE1>	LATIN SMALL LETTER O WITH HORN AND TILDE
4847	<O9-.>	<U1EE2>	LATIN CAPITAL LETTER O WITH HORN AND DOT BELOW

<o9-.>	<U1EE3>	LATIN SMALL LETTER O WITH HORN AND DOT BELOW
<U-.>	<U1EE4>	LATIN CAPITAL LETTER U WITH DOT BELOW
<u-.>	<U1EE5>	LATIN SMALL LETTER U WITH DOT BELOW
<U2>	<U1EE6>	LATIN CAPITAL LETTER U WITH HOOK ABOVE
<u2>	<U1EE7>	LATIN SMALL LETTER U WITH HOOK ABOVE
<U9'>	<U1EE8>	LATIN CAPITAL LETTER U WITH HORN AND ACUTE
<u9'>	<U1EE9>	LATIN SMALL LETTER U WITH HORN AND ACUTE
<U9!>	<U1EEA>	LATIN CAPITAL LETTER U WITH HORN AND GRAVE
<u9!>	<U1EEB>	LATIN SMALL LETTER U WITH HORN AND GRAVE
<U92>	<U1EEC>	LATIN CAPITAL LETTER U WITH HORN AND HOOK ABOVE
<u92>	<U1EED>	LATIN SMALL LETTER U WITH HORN AND HOOK ABOVE
<U9?>	<U1EEE>	LATIN CAPITAL LETTER U WITH HORN AND TILDE
<u9?>	<U1EEF>	LATIN SMALL LETTER U WITH HORN AND TILDE
<U9-.>	<U1EF0>	LATIN CAPITAL LETTER U WITH HORN AND DOT BELOW
<u9-.>	<U1EF1>	LATIN SMALL LETTER U WITH HORN AND DOT BELOW
<Y!>	<U1EF2>	LATIN CAPITAL LETTER Y WITH GRAVE
<y!>	<U1EF3>	LATIN SMALL LETTER Y WITH GRAVE
<Y-.>	<U1EF4>	LATIN CAPITAL LETTER Y WITH DOT BELOW
<y-.>	<U1EF5>	LATIN SMALL LETTER Y WITH DOT BELOW
<Y2>	<U1EF6>	LATIN CAPITAL LETTER Y WITH HOOK ABOVE
<y2>	<U1EF7>	LATIN SMALL LETTER Y WITH HOOK ABOVE
<Y?>	<U1EF8>	LATIN CAPITAL LETTER Y WITH TILDE
<y?>	<U1EF9>	LATIN SMALL LETTER Y WITH TILDE
<a*,>	<U1F00>	GREEK SMALL LETTER ALPHA WITH PSILI
<a*;>	<U1F01>	GREEK SMALL LETTER ALPHA WITH DASIA
<a*;!>	<U1F02>	GREEK SMALL LETTER ALPHA WITH PSILI AND VARIA
<a*;!>	<U1F03>	GREEK SMALL LETTER ALPHA WITH DASIA AND VARIA
<a*,'>	<U1F04>	GREEK SMALL LETTER ALPHA WITH PSILI AND OXIA
<a*,'>	<U1F05>	GREEK SMALL LETTER ALPHA WITH DASIA AND OXIA
<a*,>	<U1F06>	GREEK SMALL LETTER ALPHA WITH PSILI AND PERISPOMENI
<a*,>	<U1F07>	GREEK SMALL LETTER ALPHA WITH DASIA AND PERISPOMENI
<A*,>	<U1F08>	GREEK CAPITAL LETTER ALPHA WITH PSILI
<A*;>	<U1F09>	GREEK CAPITAL LETTER ALPHA WITH DASIA
<A*;!>	<U1F0A>	GREEK CAPITAL LETTER ALPHA WITH PSILI AND VARIA
<A*;!>	<U1F0B>	GREEK CAPITAL LETTER ALPHA WITH DASIA AND VARIA
<A*,'>	<U1F0C>	GREEK CAPITAL LETTER ALPHA WITH PSILI AND OXIA
<A*,'>	<U1F0D>	GREEK CAPITAL LETTER ALPHA WITH DASIA AND OXIA
<A*,>	<U1F0E>	GREEK CAPITAL LETTER ALPHA WITH PSILI AND PERISPOMENI
<A*,>	<U1F0F>	GREEK CAPITAL LETTER ALPHA WITH DASIA AND PERISPOMENI
<e*,>	<U1F10>	GREEK SMALL LETTER EPSILON WITH PSILI
<e*;>	<U1F11>	GREEK SMALL LETTER EPSILON WITH DASIA
<e*;!>	<U1F12>	GREEK SMALL LETTER EPSILON WITH PSILI AND VARIA
<e*;!>	<U1F13>	GREEK SMALL LETTER EPSILON WITH DASIA AND VARIA
<e*,'>	<U1F14>	GREEK SMALL LETTER EPSILON WITH PSILI AND OXIA
<e*,'>	<U1F15>	GREEK SMALL LETTER EPSILON WITH DASIA AND OXIA
<E*,>	<U1F18>	GREEK CAPITAL LETTER EPSILON WITH PSILI
<E*;>	<U1F19>	GREEK CAPITAL LETTER EPSILON WITH DASIA
<E*;!>	<U1F1A>	GREEK CAPITAL LETTER EPSILON WITH PSILI AND VARIA
<E*;!>	<U1F1B>	GREEK CAPITAL LETTER EPSILON WITH DASIA AND VARIA
<E*,'>	<U1F1C>	GREEK CAPITAL LETTER EPSILON WITH PSILI AND OXIA
<E*,'>	<U1F1D>	GREEK CAPITAL LETTER EPSILON WITH DASIA AND OXIA
<y*,>	<U1F20>	GREEK SMALL LETTER ETA WITH PSILI
<y*;>	<U1F21>	GREEK SMALL LETTER ETA WITH DASIA
<y*;!>	<U1F22>	GREEK SMALL LETTER ETA WITH PSILI AND VARIA
<y*;!>	<U1F23>	GREEK SMALL LETTER ETA WITH DASIA AND VARIA
<y*,'>	<U1F24>	GREEK SMALL LETTER ETA WITH PSILI AND OXIA
<y*,'>	<U1F25>	GREEK SMALL LETTER ETA WITH DASIA AND OXIA
<y*,>	<U1F26>	GREEK SMALL LETTER ETA WITH PSILI AND PERISPOMENI
<y*,>	<U1F27>	GREEK SMALL LETTER ETA WITH DASIA AND PERISPOMENI
<Y*,>	<U1F28>	GREEK CAPITAL LETTER ETA WITH PSILI
<Y*;>	<U1F29>	GREEK CAPITAL LETTER ETA WITH DASIA
<Y*;!>	<U1F2A>	GREEK CAPITAL LETTER ETA WITH PSILI AND VARIA
<Y*;!>	<U1F2B>	GREEK CAPITAL LETTER ETA WITH DASIA AND VARIA
<Y*,'>	<U1F2C>	GREEK CAPITAL LETTER ETA WITH PSILI AND OXIA
<Y*,'>	<U1F2D>	GREEK CAPITAL LETTER ETA WITH DASIA AND OXIA
<Y*,>	<U1F2E>	GREEK CAPITAL LETTER ETA WITH PSILI AND PERISPOMENI
<Y*,>	<U1F2F>	GREEK CAPITAL LETTER ETA WITH DASIA AND PERISPOMENI
<i*,>	<U1F30>	GREEK SMALL LETTER IOTA WITH PSILI
<i*;>	<U1F31>	GREEK SMALL LETTER IOTA WITH DASIA
<i*;!>	<U1F32>	GREEK SMALL LETTER IOTA WITH PSILI AND VARIA
<i*;!>	<U1F33>	GREEK SMALL LETTER IOTA WITH DASIA AND VARIA
<i*,'>	<U1F34>	GREEK SMALL LETTER IOTA WITH PSILI AND OXIA
<i*,'>	<U1F35>	GREEK SMALL LETTER IOTA WITH DASIA AND OXIA
<i*,>	<U1F36>	GREEK SMALL LETTER IOTA WITH PSILI AND PERISPOMENI
<i*,>	<U1F37>	GREEK SMALL LETTER IOTA WITH DASIA AND PERISPOMENI
<I*,>	<U1F38>	GREEK CAPITAL LETTER IOTA WITH PSILI
<I*;>	<U1F39>	GREEK CAPITAL LETTER IOTA WITH DASIA
<I*;!>	<U1F3A>	GREEK CAPITAL LETTER IOTA WITH PSILI AND VARIA
<I*;!>	<U1F3B>	GREEK CAPITAL LETTER IOTA WITH DASIA AND VARIA
<I*,'>	<U1F3C>	GREEK CAPITAL LETTER IOTA WITH PSILI AND OXIA
<I*,'>	<U1F3D>	GREEK CAPITAL LETTER IOTA WITH DASIA AND OXIA
<I*,>	<U1F3E>	GREEK CAPITAL LETTER IOTA WITH PSILI AND PERISPOMENI
<I*,>	<U1F3F>	GREEK CAPITAL LETTER IOTA WITH DASIA AND PERISPOMENI
<o*,>	<U1F40>	GREEK SMALL LETTER OMICRON WITH PSILI
<o*;>	<U1F41>	GREEK SMALL LETTER OMICRON WITH DASIA
<o*;!>	<U1F42>	GREEK SMALL LETTER OMICRON WITH PSILI AND VARIA
<o*;!>	<U1F43>	GREEK SMALL LETTER OMICRON WITH DASIA AND VARIA
<o*,'>	<U1F44>	GREEK SMALL LETTER OMICRON WITH PSILI AND OXIA

<o* ;'>	<U1F45>	GREEK SMALL LETTER OMICRON WITH DASIA AND OXIA
<O* ,>	<U1F48>	GREEK CAPITAL LETTER OMICRON WITH PSILI
<O* ;>	<U1F49>	GREEK CAPITAL LETTER OMICRON WITH DASIA
<O* ,!>	<U1F4A>	GREEK CAPITAL LETTER OMICRON WITH PSILI AND VARIA
<O* ;!>	<U1F4B>	GREEK CAPITAL LETTER OMICRON WITH DASIA AND VARIA
<O* ,'>	<U1F4C>	GREEK CAPITAL LETTER OMICRON WITH PSILI AND OXIA
<O* ;'>	<U1F4D>	GREEK CAPITAL LETTER OMICRON WITH DASIA AND OXIA
<u* ,>	<U1F50>	GREEK SMALL LETTER UPSILON WITH PSILI
<u* ;>	<U1F51>	GREEK SMALL LETTER UPSILON WITH DASIA
<u* ,!>	<U1F52>	GREEK SMALL LETTER UPSILON WITH PSILI AND VARIA
<u* ;!>	<U1F53>	GREEK SMALL LETTER UPSILON WITH DASIA AND VARIA
<u* ,'>	<U1F54>	GREEK SMALL LETTER UPSILON WITH PSILI AND OXIA
<u* ;'>	<U1F55>	GREEK SMALL LETTER UPSILON WITH DASIA AND OXIA
<u* ,?>	<U1F56>	GREEK SMALL LETTER UPSILON WITH PSILI AND PERISPOMENI
<u* ;?>	<U1F57>	GREEK SMALL LETTER UPSILON WITH DASIA AND PERISPOMENI
<U* ;>	<U1F59>	GREEK CAPITAL LETTER UPSILON WITH DASIA
<U* ;!>	<U1F5B>	GREEK CAPITAL LETTER UPSILON WITH DASIA AND VARIA
<U* ;'>	<U1F5D>	GREEK CAPITAL LETTER UPSILON WITH DASIA AND OXIA
<U* ;?>	<U1F5F>	GREEK CAPITAL LETTER UPSILON WITH DASIA AND PERISPOMENI
<w* ,>	<U1F60>	GREEK SMALL LETTER OMEGA WITH PSILI
<w* ;>	<U1F61>	GREEK SMALL LETTER OMEGA WITH DASIA
<w* ,!>	<U1F62>	GREEK SMALL LETTER OMEGA WITH PSILI AND VARIA
<w* ;!>	<U1F63>	GREEK SMALL LETTER OMEGA WITH DASIA AND VARIA
<w* ,'>	<U1F64>	GREEK SMALL LETTER OMEGA WITH PSILI AND OXIA
<w* ;'>	<U1F65>	GREEK SMALL LETTER OMEGA WITH DASIA AND OXIA
<w* ,?>	<U1F66>	GREEK SMALL LETTER OMEGA WITH PSILI AND PERISPOMENI
<w* ;?>	<U1F67>	GREEK SMALL LETTER OMEGA WITH DASIA AND PERISPOMENI
<W* ,>	<U1F68>	GREEK CAPITAL LETTER OMEGA WITH PSILI
<W* ;>	<U1F69>	GREEK CAPITAL LETTER OMEGA WITH DASIA
<W* ,!>	<U1F6A>	GREEK CAPITAL LETTER OMEGA WITH PSILI AND VARIA
<W* ;!>	<U1F6B>	GREEK CAPITAL LETTER OMEGA WITH DASIA AND VARIA
<W* ,'>	<U1F6C>	GREEK CAPITAL LETTER OMEGA WITH PSILI AND OXIA
<W* ;'>	<U1F6D>	GREEK CAPITAL LETTER OMEGA WITH DASIA AND OXIA
<W* ,?>	<U1F6E>	GREEK CAPITAL LETTER OMEGA WITH PSILI AND PERISPOMENI
<W* ;?>	<U1F6F>	GREEK CAPITAL LETTER OMEGA WITH DASIA AND PERISPOMENI
<a* !>	<U1F70>	GREEK SMALL LETTER ALPHA WITH VARIA
<a* ;>	<U1F71>	GREEK SMALL LETTER ALPHA WITH OXIA
<e* !>	<U1F72>	GREEK SMALL LETTER EPSILON WITH VARIA
<e* ;>	<U1F73>	GREEK SMALL LETTER EPSILON WITH OXIA
<y* !>	<U1F74>	GREEK SMALL LETTER ETA WITH VARIA
<y* ;>	<U1F75>	GREEK SMALL LETTER ETA WITH OXIA
<i* !>	<U1F76>	GREEK SMALL LETTER IOTA WITH VARIA
<i* ;>	<U1F77>	GREEK SMALL LETTER IOTA WITH OXIA
<o* !>	<U1F78>	GREEK SMALL LETTER OMICRON WITH VARIA
<o* ;>	<U1F79>	GREEK SMALL LETTER OMICRON WITH OXIA
<u* !>	<U1F7A>	GREEK SMALL LETTER UPSILON WITH VARIA
<u* ;>	<U1F7B>	GREEK SMALL LETTER UPSILON WITH OXIA
<w* !>	<U1F7C>	GREEK SMALL LETTER OMEGA WITH VARIA
<w* ;>	<U1F7D>	GREEK SMALL LETTER OMEGA WITH OXIA
<a* ,j>	<U1F80>	GREEK SMALL LETTER ALPHA WITH PSILI AND YPOGEGRAMMENI
<a* ;j>	<U1F81>	GREEK SMALL LETTER ALPHA WITH DASIA AND YPOGEGRAMMENI
<a* ,!j>	<U1F82>	GREEK SMALL LETTER ALPHA WITH PSILI AND VARIA AND YPOGEGRAMMENI
<a* ;!j>	<U1F83>	GREEK SMALL LETTER ALPHA WITH DASIA AND VARIA AND YPOGEGRAMMENI
<a* ,'j>	<U1F84>	GREEK SMALL LETTER ALPHA WITH PSILI AND OXIA AND YPOGEGRAMMENI
<a* ;'j>	<U1F85>	GREEK SMALL LETTER ALPHA WITH DASIA AND OXIA AND YPOGEGRAMMENI
<a* ,?j>	<U1F86>	GREEK SMALL LETTER ALPHA WITH PSILI AND PERISPOMENI AND YPOGEGRAMMENI
<a* ;?j>	<U1F87>	GREEK SMALL LETTER ALPHA WITH DASIA AND PERISPOMENI AND YPOGEGRAMMENI
<A* ,J>	<U1F88>	GREEK CAPITAL LETTER ALPHA WITH PSILI AND PROSGEGRAMMENI
<A* ;J>	<U1F89>	GREEK CAPITAL LETTER ALPHA WITH DASIA AND PROSGEGRAMMENI
<A* ,!J>	<U1F8A>	GREEK CAPITAL LETTER ALPHA WITH PSILI AND VARIA AND PROSGEGRAMMENI
<A* ;!J>	<U1F8B>	GREEK CAPITAL LETTER ALPHA WITH DASIA AND VARIA AND PROSGEGRAMMENI
<A* ,'J>	<U1F8C>	GREEK CAPITAL LETTER ALPHA WITH PSILI AND OXIA AND PROSGEGRAMMENI
<A* ;'J>	<U1F8D>	GREEK CAPITAL LETTER ALPHA WITH DASIA AND OXIA AND PROSGEGRAMMENI
<A* ,?J>	<U1F8E>	GREEK CAPITAL LETTER ALPHA WITH PSILI AND PERISPOMENI AND PROSGEGRAMMENI
<A* ;?J>	<U1F8F>	GREEK CAPITAL LETTER ALPHA WITH DASIA AND PERISPOMENI AND PROSGEGRAMMENI
<y* ,j>	<U1F90>	GREEK SMALL LETTER ETA WITH PSILI AND YPOGEGRAMMENI
<y* ;j>	<U1F91>	GREEK SMALL LETTER ETA WITH DASIA AND YPOGEGRAMMENI
<y* ,!j>	<U1F92>	GREEK SMALL LETTER ETA WITH PSILI AND VARIA AND YPOGEGRAMMENI
<y* ;!j>	<U1F93>	GREEK SMALL LETTER ETA WITH DASIA AND VARIA AND YPOGEGRAMMENI
<y* ,'j>	<U1F94>	GREEK SMALL LETTER ETA WITH PSILI AND OXIA AND YPOGEGRAMMENI
<y* ;'j>	<U1F95>	GREEK SMALL LETTER ETA WITH DASIA AND OXIA AND YPOGEGRAMMENI
<y* ,?j>	<U1F96>	GREEK SMALL LETTER ETA WITH PSILI AND PERISPOMENI AND YPOGEGRAMMENI
<y* ;?j>	<U1F97>	GREEK SMALL LETTER ETA WITH DASIA AND PERISPOMENI AND YPOGEGRAMMENI
<Y* ,J>	<U1F98>	GREEK CAPITAL LETTER ETA WITH PSILI AND PROSGEGRAMMENI
<Y* ;J>	<U1F99>	GREEK CAPITAL LETTER ETA WITH DASIA AND PROSGEGRAMMENI
<Y* ,!J>	<U1F9A>	GREEK CAPITAL LETTER ETA WITH PSILI AND VARIA AND PROSGEGRAMMENI
<Y* ;!J>	<U1F9B>	GREEK CAPITAL LETTER ETA WITH DASIA AND VARIA AND PROSGEGRAMMENI
<Y* ,'J>	<U1F9C>	GREEK CAPITAL LETTER ETA WITH PSILI AND OXIA AND PROSGEGRAMMENI
<Y* ;'J>	<U1F9D>	GREEK CAPITAL LETTER ETA WITH DASIA AND OXIA AND PROSGEGRAMMENI
<Y* ,?J>	<U1F9E>	GREEK CAPITAL LETTER ETA WITH PSILI AND PERISPOMENI AND PROSGEGRAMMENI
<Y* ;?J>	<U1F9F>	GREEK CAPITAL LETTER ETA WITH DASIA AND PERISPOMENI AND PROSGEGRAMMENI
<w* ,j>	<U1FA0>	GREEK SMALL LETTER OMEGA WITH PSILI AND YPOGEGRAMMENI
<w* ;j>	<U1FA1>	GREEK SMALL LETTER OMEGA WITH DASIA AND YPOGEGRAMMENI
<w* ,!j>	<U1FA2>	GREEK SMALL LETTER OMEGA WITH PSILI AND VARIA AND YPOGEGRAMMENI
<w* ;!j>	<U1FA3>	GREEK SMALL LETTER OMEGA WITH DASIA AND VARIA AND YPOGEGRAMMENI
<w* ,'j>	<U1FA4>	GREEK SMALL LETTER OMEGA WITH PSILI AND OXIA AND YPOGEGRAMMENI
<w* ;'j>	<U1FA5>	GREEK SMALL LETTER OMEGA WITH DASIA AND OXIA AND YPOGEGRAMMENI
<w* ,?j>	<U1FA6>	GREEK SMALL LETTER OMEGA WITH PSILI AND PERISPOMENI AND YPOGEGRAMMENI

<w*?j>	<U1FA7>	GREEK SMALL LETTER OMEGA WITH DASIA AND PERISPOMENI AND YPOGEGRAMMENI
<W*,J>	<U1FA8>	GREEK CAPITAL LETTER OMEGA WITH PSILI AND PROSGEGRAMMENI
<W*!J>	<U1FA9>	GREEK CAPITAL LETTER OMEGA WITH DASIA AND PROSGEGRAMMENI
<W*!J>	<U1FAA>	GREEK CAPITAL LETTER OMEGA WITH PSILI AND VARIA AND PROSGEGRAMMENI
<W*!J>	<U1FAB>	GREEK CAPITAL LETTER OMEGA WITH DASIA AND VARIA AND PROSGEGRAMMENI
<W*,'J>	<U1FAC>	GREEK CAPITAL LETTER OMEGA WITH PSILI AND OXIA AND PROSGEGRAMMENI
<W*;'J>	<U1FAD>	GREEK CAPITAL LETTER OMEGA WITH DASIA AND OXIA AND PROSGEGRAMMENI
<W*?J>	<U1FAE>	GREEK CAPITAL LETTER OMEGA WITH PSILI AND PERISPOMENI AND PROSGEGRAMMENI
<W*?J>	<U1FAF>	GREEK CAPITAL LETTER OMEGA WITH DASIA AND PERISPOMENI AND PROSGEGRAMMENI
<a*!>	<U1FB0>	GREEK SMALL LETTER ALPHA WITH VRACHY
<a*->	<U1FB1>	GREEK SMALL LETTER ALPHA WITH MACRON
<a*!j>	<U1FB2>	GREEK SMALL LETTER ALPHA WITH VARIA AND YPOGEGRAMMENI
<a*j>	<U1FB3>	GREEK SMALL LETTER ALPHA WITH YPOGEGRAMMENI
<a*j>	<U1FB4>	GREEK SMALL LETTER ALPHA WITH OXIA AND YPOGEGRAMMENI
<a*?>	<U1FB6>	GREEK SMALL LETTER ALPHA WITH PERISPOMENI
<a*?j>	<U1FB7>	GREEK SMALL LETTER ALPHA WITH PERISPOMENI AND YPOGEGRAMMENI
<A*!>	<U1FB8>	GREEK CAPITAL LETTER ALPHA WITH VRACHY
<A*->	<U1FB9>	GREEK CAPITAL LETTER ALPHA WITH MACRON
<A*!>	<U1FBA>	GREEK CAPITAL LETTER ALPHA WITH VARIA
<A*'>	<U1FBB>	GREEK CAPITAL LETTER ALPHA WITH OXIA
<A*J>	<U1FBC>	GREEK CAPITAL LETTER ALPHA WITH PROSGEGRAMMENI
<)*>	<U1FBD>	GREEK KORONIS
<J3>	<U1FBE>	GREEK PROSGEGRAMMENI
<,,>	<U1FBF>	GREEK PSILI
<?*>	<U1FC0>	GREEK PERISPOMENI
<?:>	<U1FC1>	GREEK DIALYTIKA AND PERISPOMENI
<y*!j>	<U1FC2>	GREEK SMALL LETTER ETA WITH VARIA AND YPOGEGRAMMENI
<y*j>	<U1FC3>	GREEK SMALL LETTER ETA WITH YPOGEGRAMMENI
<y*!j>	<U1FC4>	GREEK SMALL LETTER ETA WITH OXIA AND YPOGEGRAMMENI
<y*?>	<U1FC6>	GREEK SMALL LETTER ETA WITH PERISPOMENI
<y*?j>	<U1FC7>	GREEK SMALL LETTER ETA WITH PERISPOMENI AND YPOGEGRAMMENI
<E*!!>	<U1FC8>	GREEK CAPITAL LETTER EPSILON WITH VARIA
<E*!>	<U1FC9>	GREEK CAPITAL LETTER EPSILON WITH OXIA
<Y*!>	<U1FCA>	GREEK CAPITAL LETTER ETA WITH VARIA
<Y*'>	<U1FCB>	GREEK CAPITAL LETTER ETA WITH OXIA
<Y*J>	<U1FCC>	GREEK CAPITAL LETTER ETA WITH PROSGEGRAMMENI
<,!>	<U1FCD>	GREEK PSILI AND VARIA
<,/'>	<U1FCE>	GREEK PSILI AND OXIA
<?,>	<U1FCF>	GREEK PSILI AND PERISPOMENI
<i*!>	<U1FD0>	GREEK SMALL LETTER IOTA WITH VRACHY
<i*->	<U1FD1>	GREEK SMALL LETTER IOTA WITH MACRON
<i*!>	<U1FD2>	GREEK SMALL LETTER IOTA WITH DIALYTIKA AND VARIA
<i*!'>	<U1FD3>	GREEK SMALL LETTER IOTA WITH DIALYTIKA AND OXIA
<i*?>	<U1FD6>	GREEK SMALL LETTER IOTA WITH PERISPOMENI
<i*?>	<U1FD7>	GREEK SMALL LETTER IOTA WITH DIALYTIKA AND PERISPOMENI
<I*!>	<U1FD8>	GREEK CAPITAL LETTER IOTA WITH VRACHY
<I*->	<U1FD9>	GREEK CAPITAL LETTER IOTA WITH MACRON
<I*!>	<U1FDA>	GREEK CAPITAL LETTER IOTA WITH VARIA
<I*!>	<U1FDB>	GREEK CAPITAL LETTER IOTA WITH OXIA
<!>	<U1FDD>	GREEK DASIA AND VARIA
<!'>	<U1FDE>	GREEK DASIA AND OXIA
<?;>	<U1FDF>	GREEK DASIA AND PERISPOMENI
<u*!>	<U1FE0>	GREEK SMALL LETTER UPSILON WITH VRACHY
<u*->	<U1FE1>	GREEK SMALL LETTER UPSILON WITH MACRON
<u*!>	<U1FE2>	GREEK SMALL LETTER UPSILON WITH DIALYTIKA AND VARIA
<u*!'>	<U1FE3>	GREEK SMALL LETTER UPSILON WITH DIALYTIKA AND OXIA
<r*,>	<U1FE4>	GREEK SMALL LETTER RHO WITH PSILI
<r*!>	<U1FE5>	GREEK SMALL LETTER RHO WITH DASIA
<u*?>	<U1FE6>	GREEK SMALL LETTER UPSILON WITH PERISPOMENI
<u*?>	<U1FE7>	GREEK SMALL LETTER UPSILON WITH DIALYTIKA AND PERISPOMENI
<U*!>	<U1FE8>	GREEK CAPITAL LETTER UPSILON WITH VRACHY
<U*->	<U1FE9>	GREEK CAPITAL LETTER UPSILON WITH MACRON
<U*!>	<U1FEA>	GREEK CAPITAL LETTER UPSILON WITH VARIA
<U*!>	<U1FEB>	GREEK CAPITAL LETTER UPSILON WITH OXIA
<R*!>	<U1FEC>	GREEK CAPITAL LETTER RHO WITH DASIA
<!:>	<U1FED>	GREEK DIALYTIKA AND VARIA
<:'>	<U1FEE>	GREEK DIALYTIKA AND OXIA
<!*>	<U1FEF>	GREEK VARIA
<w*!j>	<U1FF2>	GREEK SMALL LETTER OMEGA WITH VARIA AND YPOGEGRAMMENI
<w*j>	<U1FF3>	GREEK SMALL LETTER OMEGA WITH YPOGEGRAMMENI
<w*!j>	<U1FF4>	GREEK SMALL LETTER OMEGA WITH OXIA AND YPOGEGRAMMENI
<w*?>	<U1FF6>	GREEK SMALL LETTER OMEGA WITH PERISPOMENI
<w*?j>	<U1FF7>	GREEK SMALL LETTER OMEGA WITH PERISPOMENI AND YPOGEGRAMMENI
<O*!>	<U1FF8>	GREEK CAPITAL LETTER OMICRON WITH VARIA
<O*'>	<U1FF9>	GREEK CAPITAL LETTER OMICRON WITH OXIA
<W*!>	<U1FFA>	GREEK CAPITAL LETTER OMEGA WITH VARIA
<W*'>	<U1FFB>	GREEK CAPITAL LETTER OMEGA WITH OXIA
<W*J>	<U1FFC>	GREEK CAPITAL LETTER OMEGA WITH PROSGEGRAMMENI
<//*>	<U1FFD>	GREEK OXIA
<!;>	<U1FFE>	GREEK DASIA
<1N>	<U2002>	EN SPACE
<1M>	<U2003>	EM SPACE
<3M>	<U2004>	THREE-PER-EM SPACE
<4M>	<U2005>	FOUR-PER-EM SPACE
<6M>	<U2006>	SIX-PER-EM SPACE
<LR>	<U200E>	LEFT-TO-RIGHT MARK
<RL>	<U200F>	RIGHT-TO-LEFT MARK
<1T>	<U2009>	THIN SPACE

<1H>	<U200A>	HAIR SPACE
<-1>	<U2010>	HYPHEN
<-N>	<U2013>	EN DASH
<-M>	<U2014>	EM DASH
<-3>	<U2015>	HORIZONTAL BAR
< 2>	<U2016>	DOUBLE VERTICAL LINE
<=2>	<U2017>	DOUBLE LOW LINE
<'6>	<U2018>	LEFT SINGLE QUOTATION MARK
<'9>	<U2019>	RIGHT SINGLE QUOTATION MARK
<.9>	<U201A>	SINGLE LOW-9 QUOTATION MARK
<9'>	<U201B>	SINGLE HIGH-REVERSED-9 QUOTATION MARK
<"6>	<U201C>	LEFT DOUBLE QUOTATION MARK
<"9>	<U201D>	RIGHT DOUBLE QUOTATION MARK
<:9>	<U201E>	DOUBLE LOW-9 QUOTATION MARK
<9">	<U201F>	DOUBLE HIGH-REVERSED-9 QUOTATION MARK
<//->	<U2020>	DAGGER
<//=>	<U2021>	DOUBLE DAGGER
<sb>	<U2022>	BULLET
<3b>	<U2023>	TRIANGULAR BULLET
<..>	<U2025>	TWO DOT LEADER
<.3>	<U2026>	HORIZONTAL ELLIPSIS
<.->	<U2027>	HYPHENATION POINT
<linesep>	<U2028>	LINE SEPARATOR
<parsep>	<U2029>	PARAGRAPH SEPARATOR
<%0>	<U2030>	PER MILLE SIGN
<1'>	<U2032>	PRIME
<2'>	<U2033>	DOUBLE PRIME
<3'>	<U2034>	TRIPLE PRIME
<1">	<U2035>	REVERSED PRIME
<2">	<U2036>	REVERSED DOUBLE PRIME
<3">	<U2037>	REVERSED TRIPLE PRIME
<Ca>	<U2038>	CARET
<<1>	<U2039>	SINGLE LEFT-POINTING ANGLE QUOTATION MARK
</>1>	<U203A>	SINGLE RIGHT-POINTING ANGLE QUOTATION MARK
<:X>	<U203B>	REFERENCE MARK
<!*2>	<U203C>	DOUBLE EXCLAMATION MARK
<' ->	<U203E>	OVERLINE
<-b>	<U2043>	HYPHEN BULLET
<///f>	<U2044>	FRACTION SLASH
<0S>	<U2070>	SUPERSCRIPIT ZERO
<4S>	<U2074>	SUPERSCRIPIT FOUR
<5S>	<U2075>	SUPERSCRIPIT FIVE
<6S>	<U2076>	SUPERSCRIPIT SIX
<7S>	<U2077>	SUPERSCRIPIT SEVEN
<8S>	<U2078>	SUPERSCRIPIT EIGHT
<9S>	<U2079>	SUPERSCRIPIT NINE
<+S>	<U207A>	SUPERSCRIPIT PLUS SIGN
<-S>	<U207B>	SUPERSCRIPIT MINUS
<=S>	<U207C>	SUPERSCRIPIT EQUALS SIGN
<(S>	<U207D>	SUPERSCRIPIT LEFT PARENTHESIS
<)S>	<U207E>	SUPERSCRIPIT RIGHT PARENTHESIS
<nS>	<U207F>	SUPERSCRIPIT LATIN SMALL LETTER N
<0s>	<U2080>	SUBSCRIPT ZERO
<1s>	<U2081>	SUBSCRIPT ONE
<2s>	<U2082>	SUBSCRIPT TWO
<3s>	<U2083>	SUBSCRIPT THREE
<4s>	<U2084>	SUBSCRIPT FOUR
<5s>	<U2085>	SUBSCRIPT FIVE
<6s>	<U2086>	SUBSCRIPT SIX
<7s>	<U2087>	SUBSCRIPT SEVEN
<8s>	<U2088>	SUBSCRIPT EIGHT
<9s>	<U2089>	SUBSCRIPT NINE
<+s>	<U208A>	SUBSCRIPT PLUS SIGN
<-s>	<U208B>	SUBSCRIPT MINUS
<=s>	<U208C>	SUBSCRIPT EQUALS SIGN
<(s>	<U208D>	SUBSCRIPT LEFT PARENTHESIS
<)s>	<U208E>	SUBSCRIPT RIGHT PARENTHESIS
<Ff>	<U20A3>	FRENCH FRANC SIGN
	<U20A4>	LIRA SIGN
<Pt>	<U20A7>	PESETA SIGN
<W=>	<U20A9>	WON SIGN
<"7>	<U20D1>	COMBINING RIGHT HARPOON ABOVE
<oC>	<U2103>	DEGREE CELSIUS
<co>	<U2105>	CARE OF
<oF>	<U2109>	DEGREE FAHRENHEIT
<N0>	<U2116>	NUMERO SIGN
<PO>	<U2117>	SOUND RECORDING COPYRIGHT
<Rx>	<U211E>	PRESCRIPTION TAKE
<SM>	<U2120>	SERVICE MARK
<TM>	<U2122>	TRADE MARK SIGN
<Om>	<U2126>	OHM SIGN
<AO>	<U212B>	ANGSTROM SIGN
<Est>	<U212E>	ESTIMATED SYMBOL
<13>	<U2153>	VULGAR FRACTION ONE THIRD
<23>	<U2154>	VULGAR FRACTION TWO THIRDS
<15>	<U2155>	VULGAR FRACTION ONE FIFTH
<25>	<U2156>	VULGAR FRACTION TWO FIFTHS
<35>	<U2157>	VULGAR FRACTION THREE FIFTHS

5200	<45>	<U2158>	VULGAR FRACTION FOUR FIFTHS
5201	<16>	<U2159>	VULGAR FRACTION ONE SIXTH
5202	<56>	<U215A>	VULGAR FRACTION FIVE SIXTHS
5203	<18>	<U215B>	VULGAR FRACTION ONE EIGHTH
5204	<38>	<U215C>	VULGAR FRACTION THREE EIGHTHS
5205	<58>	<U215D>	VULGAR FRACTION FIVE EIGHTHS
5206	<78>	<U215E>	VULGAR FRACTION SEVEN EIGHTHS
5207	<1R>	<U2160>	ROMAN NUMERAL ONE
5208	<2R>	<U2161>	ROMAN NUMERAL TWO
5209	<3R>	<U2162>	ROMAN NUMERAL THREE
5210	<4R>	<U2163>	ROMAN NUMERAL FOUR
5211	<5R>	<U2164>	ROMAN NUMERAL FIVE
5212	<6R>	<U2165>	ROMAN NUMERAL SIX
5213	<7R>	<U2166>	ROMAN NUMERAL SEVEN
5214	<8R>	<U2167>	ROMAN NUMERAL EIGHT
5215	<9R>	<U2168>	ROMAN NUMERAL NINE
5216	<aR>	<U2169>	ROMAN NUMERAL TEN
5217	 	<U216A>	ROMAN NUMERAL ELEVEN
5218	<cR>	<U216B>	ROMAN NUMERAL TWELVE
5219	<50R>	<U216C>	ROMAN NUMERAL FIFTY
5220	<100R>	<U216D>	ROMAN NUMERAL ONE HUNDRED
5221	<500R>	<U216E>	ROMAN NUMERAL FIVE HUNDRED
5222	<1000R>	<U216F>	ROMAN NUMERAL ONE THOUSAND
5223	<1r>	<U2170>	SMALL ROMAN NUMERAL ONE
5224	<2r>	<U2171>	SMALL ROMAN NUMERAL TWO
5225	<3r>	<U2172>	SMALL ROMAN NUMERAL THREE
5226	<4r>	<U2173>	SMALL ROMAN NUMERAL FOUR
5227	<5r>	<U2174>	SMALL ROMAN NUMERAL FIVE
5228	<6r>	<U2175>	SMALL ROMAN NUMERAL SIX
5229	<7r>	<U2176>	SMALL ROMAN NUMERAL SEVEN
5230	<8r>	<U2177>	SMALL ROMAN NUMERAL EIGHT
5231	<9r>	<U2178>	SMALL ROMAN NUMERAL NINE
5232	<ar>	<U2179>	SMALL ROMAN NUMERAL TEN
5233	 	<U217A>	SMALL ROMAN NUMERAL ELEVEN
5234	<cr>	<U217B>	SMALL ROMAN NUMERAL TWELVE
5235	<50r>	<U217C>	SMALL ROMAN NUMERAL FIFTY
5236	<100r>	<U217D>	SMALL ROMAN NUMERAL ONE HUNDRED
5237	<500r>	<U217E>	SMALL ROMAN NUMERAL FIVE HUNDRED
5238	<1000r>	<U217F>	SMALL ROMAN NUMERAL ONE THOUSAND
5239	<1000RCD>	<U2180>	ROMAN NUMERAL ONE THOUSAND C D
5240	<5000R>	<U2181>	ROMAN NUMERAL FIVE THOUSAND
5241	<10000R>	<U2182>	ROMAN NUMERAL TEN THOUSAND
5242	<<->	<U2190>	LEFTWARDS ARROW
5243	<-!>	<U2191>	UPWARDS ARROW
5244	<-/>>	<U2192>	RIGHTWARDS ARROW
5245	<-v>	<U2193>	DOWNWARDS ARROW
5246	<</>>	<U2194>	LEFT RIGHT ARROW
5247	<UD>	<U2195>	UP DOWN ARROW
5248	<<!>	<U2196>	NORTH WEST ARROW
5249	</////>>	<U2197>	NORTH EAST ARROW
5250	<!>/>>	<U2198>	SOUTH EAST ARROW
5251	<</////>	<U2199>	SOUTH WEST ARROW
5252	<UD->	<U21A8>	UP DOWN ARROW WITH BASE
5253	</>V>	<U21C0>	RIGHTWARDS HARPOON WITH BARB UPWARDS
5254	<<=>	<U21D0>	LEFTWARDS DOUBLE ARROW
5255	<=>/>>	<U21D2>	RIGHTWARDS DOUBLE ARROW
5256	<==>	<U21D4>	LEFT RIGHT DOUBLE ARROW
5257	<FA>	<U2200>	FOR ALL
5258	<dP>	<U2202>	PARTIAL DIFFERENTIAL
5259	<TE>	<U2203>	THERE EXISTS
5260	<//0>	<U2205>	EMPTY SET
5261	<DE>	<U2206>	INCREMENT
5262	<NB>	<U2207>	NABLA
5263	<(->	<U2208>	ELEMENT OF
5264	<(-)>	<U220B>	CONTAINS AS MEMBER
5265	<FP>	<U220E>	END OF PROOF
5266	<*P>	<U220F>	N-ARY PRODUCT
5267	<+Z>	<U2211>	N-ARY SUMMATION
5268	<-2>	<U2212>	MINUS SIGN
5269	<-+>	<U2213>	MINUS-OR-PLUS SIGN
5270	<.+>	<U2214>	DOT PLUS
5271	<*->	<U2217>	ASTERISK OPERATOR
5272	<Ob>	<U2218>	RING OPERATOR
5273	<Sb>	<U2219>	BULLET OPERATOR
5274	<RT>	<U221A>	SQUARE ROOT
5275	<0(>	<U221D>	PROPORTIONAL TO
5276	<00>	<U221E>	INFINITY
5277	<-L>	<U221F>	RIGHT ANGLE
5278	<-V>	<U2220>	ANGLE
5279	<PP>	<U2225>	PARALLEL TO
5280	<AN>	<U2227>	LOGICAL AND
5281	<OR>	<U2228>	LOGICAL OR
5282	<(U>	<U2229>	INTERSECTION
5283	<)U>	<U222A>	UNION
5284	<In>	<U222B>	INTEGRAL
5285	<DI>	<U222C>	DOUBLE INTEGRAL
5286	<Io>	<U222E>	CONTOUR INTEGRAL
5287	<. :>	<U2234>	THEREFORE

<: >	<U2235>	BECAUSE
<:R>	<U2236>	RATIO
<::>	<U2237>	PROPORTION
<?1>	<U223C>	TILDE OPERATOR
<CG>	<U223E>	INVERTED LAZY S
<?->	<U2243>	ASYMPTOTICALLY EQUAL TO
<?=>	<U2245>	APPROXIMATELY EQUAL TO
<?2>	<U2248>	ALMOST EQUAL TO
<=?>	<U224C>	ALL EQUAL TO
<HI>	<U2253>	IMAGE OF OR APPROXIMATELY EQUAL TO
<!=>	<U2260>	NOT EQUAL TO
<=3>	<U2261>	IDENTICAL TO
<=<>	<U2264>	LESS-THAN OR EQUAL TO
</>=>	<U2265>	GREATER-THAN OR EQUAL TO
<<*>	<U226A>	MUCH LESS-THAN
<*/>>	<U226B>	MUCH GREATER-THAN
<!<>	<U226E>	NOT LESS-THAN
<!/>>	<U226F>	NOT GREATER-THAN
<(C>	<U2282>	SUBSET OF
<)C>	<U2283>	SUPERSET OF
<(_>	<U2286>	SUBSET OF OR EQUAL TO
<)_>	<U2287>	SUPERSET OF OR EQUAL TO
<0.>	<U2299>	CIRCLED DOT OPERATOR
<02>	<U229A>	CIRCLED RING OPERATOR
<-T>	<U22A5>	UP TACK
<.P>	<U22C5>	DOT OPERATOR
<:3>	<U22EE>	VERTICAL ELLIPSIS
<Eh>	<U2302>	HOUSE
<<7>	<U2308>	LEFT CEILING
</>7>	<U2309>	RIGHT CEILING
<7<>	<U230A>	LEFT FLOOR
<7/>>	<U230B>	RIGHT FLOOR
<NI>	<U2310>	REVERSED NOT SIGN
<(A>	<U2312>	ARC
<TR>	<U2315>	TELEPHONE RECORDER
<88>	<U2318>	PLACE OF INTEREST SIGN
<Iu>	<U2320>	TOP HALF INTEGRAL
<Il>	<U2321>	BOTTOM HALF INTEGRAL
<<//>	<U2329>	LEFT-POINTING ANGLE BRACKET
<///>>	<U232A>	RIGHT-POINTING ANGLE BRACKET
<Vs>	<U2423>	OPEN BOX
<1h>	<U2440>	OCR HOOK
<3h>	<U2441>	OCR CHAIR
<2h>	<U2442>	OCR FORK
<4h>	<U2443>	OCR INVERTED FORK
<1j>	<U2446>	OCR BRANCH BANK IDENTIFICATION
<2j>	<U2447>	OCR AMOUNT OF CHECK
<3j>	<U2448>	OCR DASH
<4j>	<U2449>	OCR CUSTOMER ACCOUNT NUMBER
<1-o>	<U2460>	CIRCLED DIGIT ONE
<2-o>	<U2461>	CIRCLED DIGIT TWO
<3-o>	<U2462>	CIRCLED DIGIT THREE
<4-o>	<U2463>	CIRCLED DIGIT FOUR
<5-o>	<U2464>	CIRCLED DIGIT FIVE
<6-o>	<U2465>	CIRCLED DIGIT SIX
<7-o>	<U2466>	CIRCLED DIGIT SEVEN
<8-o>	<U2467>	CIRCLED DIGIT EIGHT
<9-o>	<U2468>	CIRCLED DIGIT NINE
<10-o>	<U2469>	CIRCLED NUMBER TEN
<11-o>	<U246A>	CIRCLED NUMBER ELEVEN
<12-o>	<U246B>	CIRCLED NUMBER TWELVE
<13-o>	<U246C>	CIRCLED NUMBER THIRTEEN
<14-o>	<U246D>	CIRCLED NUMBER FOURTEEN
<15-o>	<U246E>	CIRCLED NUMBER FIFTEEN
<16-o>	<U246F>	CIRCLED NUMBER SIXTEEN
<17-o>	<U2470>	CIRCLED NUMBER SEVENTEEN
<18-o>	<U2471>	CIRCLED NUMBER EIGHTEEN
<19-o>	<U2472>	CIRCLED NUMBER NINETEEN
<20-o>	<U2473>	CIRCLED NUMBER TWENTY
<(1)>	<U2474>	PARENTHESIZED DIGIT ONE
<(2)>	<U2475>	PARENTHESIZED DIGIT TWO
<(3)>	<U2476>	PARENTHESIZED DIGIT THREE
<(4)>	<U2477>	PARENTHESIZED DIGIT FOUR
<(5)>	<U2478>	PARENTHESIZED DIGIT FIVE
<(6)>	<U2479>	PARENTHESIZED DIGIT SIX
<(7)>	<U247A>	PARENTHESIZED DIGIT SEVEN
<(8)>	<U247B>	PARENTHESIZED DIGIT EIGHT
<(9)>	<U247C>	PARENTHESIZED DIGIT NINE
<(10)>	<U247D>	PARENTHESIZED NUMBER TEN
<(11)>	<U247E>	PARENTHESIZED NUMBER ELEVEN
<(12)>	<U247F>	PARENTHESIZED NUMBER TWELVE
<(13)>	<U2480>	PARENTHESIZED NUMBER THIRTEEN
<(14)>	<U2481>	PARENTHESIZED NUMBER FOURTEEN
<(15)>	<U2482>	PARENTHESIZED NUMBER FIFTEEN
<(16)>	<U2483>	PARENTHESIZED NUMBER SIXTEEN
<(17)>	<U2484>	PARENTHESIZED NUMBER SEVENTEEN
<(18)>	<U2485>	PARENTHESIZED NUMBER EIGHTEEN
<(19)>	<U2486>	PARENTHESIZED NUMBER NINETEEN

<(20)>	<U2487>	PARENTHESIZED NUMBER TWENTY
<1.>	<U2488>	DIGIT ONE FULL STOP
<2.>	<U2489>	DIGIT TWO FULL STOP
<3.>	<U248A>	DIGIT THREE FULL STOP
<4.>	<U248B>	DIGIT FOUR FULL STOP
<5.>	<U248C>	DIGIT FIVE FULL STOP
<6.>	<U248D>	DIGIT SIX FULL STOP
<7.>	<U248E>	DIGIT SEVEN FULL STOP
<8.>	<U248F>	DIGIT EIGHT FULL STOP
<9.>	<U2490>	DIGIT NINE FULL STOP
<10.>	<U2491>	NUMBER TEN FULL STOP
<11.>	<U2492>	NUMBER ELEVEN FULL STOP
<12.>	<U2493>	NUMBER TWELVE FULL STOP
<13.>	<U2494>	NUMBER THIRTEEN FULL STOP
<14.>	<U2495>	NUMBER FOURTEEN FULL STOP
<15.>	<U2496>	NUMBER FIFTEEN FULL STOP
<16.>	<U2497>	NUMBER SIXTEEN FULL STOP
<17.>	<U2498>	NUMBER SEVENTEEN FULL STOP
<18.>	<U2499>	NUMBER EIGHTEEN FULL STOP
<19.>	<U249A>	NUMBER NINETEEN FULL STOP
<20.>	<U249B>	NUMBER TWENTY FULL STOP
<(a)>	<U249C>	PARENTHESIZED LATIN SMALL LETTER A
<(b)>	<U249D>	PARENTHESIZED LATIN SMALL LETTER B
<(c)>	<U249E>	PARENTHESIZED LATIN SMALL LETTER C
<(d)>	<U249F>	PARENTHESIZED LATIN SMALL LETTER D
<(e)>	<U24A0>	PARENTHESIZED LATIN SMALL LETTER E
<(f)>	<U24A1>	PARENTHESIZED LATIN SMALL LETTER F
<(g)>	<U24A2>	PARENTHESIZED LATIN SMALL LETTER G
<(h)>	<U24A3>	PARENTHESIZED LATIN SMALL LETTER H
<(i)>	<U24A4>	PARENTHESIZED LATIN SMALL LETTER I
<(j)>	<U24A5>	PARENTHESIZED LATIN SMALL LETTER J
<(k)>	<U24A6>	PARENTHESIZED LATIN SMALL LETTER K
<(l)>	<U24A7>	PARENTHESIZED LATIN SMALL LETTER L
<(m)>	<U24A8>	PARENTHESIZED LATIN SMALL LETTER M
<(n)>	<U24A9>	PARENTHESIZED LATIN SMALL LETTER N
<(o)>	<U24AA>	PARENTHESIZED LATIN SMALL LETTER O
<(p)>	<U24AB>	PARENTHESIZED LATIN SMALL LETTER P
<(q)>	<U24AC>	PARENTHESIZED LATIN SMALL LETTER Q
<(r)>	<U24AD>	PARENTHESIZED LATIN SMALL LETTER R
<(s)>	<U24AE>	PARENTHESIZED LATIN SMALL LETTER S
<(t)>	<U24AF>	PARENTHESIZED LATIN SMALL LETTER T
<(u)>	<U24B0>	PARENTHESIZED LATIN SMALL LETTER U
<(v)>	<U24B1>	PARENTHESIZED LATIN SMALL LETTER V
<(w)>	<U24B2>	PARENTHESIZED LATIN SMALL LETTER W
<(x)>	<U24B3>	PARENTHESIZED LATIN SMALL LETTER X
<(y)>	<U24B4>	PARENTHESIZED LATIN SMALL LETTER Y
<(z)>	<U24B5>	PARENTHESIZED LATIN SMALL LETTER Z
<A-o>	<U24B6>	CIRCLED LATIN CAPITAL LETTER A
<B-o>	<U24B7>	CIRCLED LATIN CAPITAL LETTER B
<C-o>	<U24B8>	CIRCLED LATIN CAPITAL LETTER C
<D-o>	<U24B9>	CIRCLED LATIN CAPITAL LETTER D
<E-o>	<U24BA>	CIRCLED LATIN CAPITAL LETTER E
<F-o>	<U24BB>	CIRCLED LATIN CAPITAL LETTER F
<G-o>	<U24BC>	CIRCLED LATIN CAPITAL LETTER G
<H-o>	<U24BD>	CIRCLED LATIN CAPITAL LETTER H
<I-o>	<U24BE>	CIRCLED LATIN CAPITAL LETTER I
<J-o>	<U24BF>	CIRCLED LATIN CAPITAL LETTER J
<K-o>	<U24C0>	CIRCLED LATIN CAPITAL LETTER K
<L-o>	<U24C1>	CIRCLED LATIN CAPITAL LETTER L
<M-o>	<U24C2>	CIRCLED LATIN CAPITAL LETTER M
<N-o>	<U24C3>	CIRCLED LATIN CAPITAL LETTER N
<O-o>	<U24C4>	CIRCLED LATIN CAPITAL LETTER O
<P-o>	<U24C5>	CIRCLED LATIN CAPITAL LETTER P
<Q-o>	<U24C6>	CIRCLED LATIN CAPITAL LETTER Q
<R-o>	<U24C7>	CIRCLED LATIN CAPITAL LETTER R
<S-o>	<U24C8>	CIRCLED LATIN CAPITAL LETTER S
<T-o>	<U24C9>	CIRCLED LATIN CAPITAL LETTER T
<U-o>	<U24CA>	CIRCLED LATIN CAPITAL LETTER U
<V-o>	<U24CB>	CIRCLED LATIN CAPITAL LETTER V
<W-o>	<U24CC>	CIRCLED LATIN CAPITAL LETTER W
<X-o>	<U24CD>	CIRCLED LATIN CAPITAL LETTER X
<Y-o>	<U24CE>	CIRCLED LATIN CAPITAL LETTER Y
<Z-o>	<U24CF>	CIRCLED LATIN CAPITAL LETTER Z
<a-o>	<U24D0>	CIRCLED LATIN SMALL LETTER A
<b-o>	<U24D1>	CIRCLED LATIN SMALL LETTER B
<c-o>	<U24D2>	CIRCLED LATIN SMALL LETTER C
<d-o>	<U24D3>	CIRCLED LATIN SMALL LETTER D
<e-o>	<U24D4>	CIRCLED LATIN SMALL LETTER E
<f-o>	<U24D5>	CIRCLED LATIN SMALL LETTER F
<g-o>	<U24D6>	CIRCLED LATIN SMALL LETTER G
<h-o>	<U24D7>	CIRCLED LATIN SMALL LETTER H
<i-o>	<U24D8>	CIRCLED LATIN SMALL LETTER I
<j-o>	<U24D9>	CIRCLED LATIN SMALL LETTER J
<k-o>	<U24DA>	CIRCLED LATIN SMALL LETTER K
<l-o>	<U24DB>	CIRCLED LATIN SMALL LETTER L
<m-o>	<U24DC>	CIRCLED LATIN SMALL LETTER M
<n-o>	<U24DD>	CIRCLED LATIN SMALL LETTER N
<o-o>	<U24DE>	CIRCLED LATIN SMALL LETTER O

<HH>	<U2550>	BOX DRAWINGS DOUBLE HORIZONTAL
<VV>	<U2551>	BOX DRAWINGS DOUBLE VERTICAL
<dR>	<U2552>	BOX DRAWINGS DOWN SINGLE AND RIGHT DOUBLE
<Dr>	<U2553>	BOX DRAWINGS DOWN DOUBLE AND RIGHT SINGLE
<DR>	<U2554>	BOX DRAWINGS DOUBLE DOWN AND RIGHT
<dL>	<U2555>	BOX DRAWINGS DOWN SINGLE AND LEFT DOUBLE
<DL>	<U2556>	BOX DRAWINGS DOWN DOUBLE AND LEFT SINGLE
<LD>	<U2557>	BOX DRAWINGS DOUBLE DOWN AND LEFT
<uR>	<U2558>	BOX DRAWINGS UP SINGLE AND RIGHT DOUBLE
<Ur>	<U2559>	BOX DRAWINGS UP DOUBLE AND RIGHT SINGLE
<UR>	<U255A>	BOX DRAWINGS DOUBLE UP AND RIGHT
	<U255B>	BOX DRAWINGS UP SINGLE AND LEFT DOUBLE
	<U255C>	BOX DRAWINGS UP DOUBLE AND LEFT SINGLE
	<U255D>	BOX DRAWINGS DOUBLE UP AND LEFT
<vR>	<U255E>	BOX DRAWINGS VERTICAL SINGLE AND RIGHT DOUBLE
<VR>	<U255F>	BOX DRAWINGS VERTICAL DOUBLE AND RIGHT SINGLE
<VR>	<U2560>	BOX DRAWINGS DOUBLE VERTICAL AND RIGHT
<vL>	<U2561>	BOX DRAWINGS VERTICAL SINGLE AND LEFT DOUBLE
<VL>	<U2562>	BOX DRAWINGS VERTICAL DOUBLE AND LEFT SINGLE
<VL>	<U2563>	BOX DRAWINGS DOUBLE VERTICAL AND LEFT
<dH>	<U2564>	BOX DRAWINGS DOWN SINGLE AND HORIZONTAL DOUBLE
<DH>	<U2565>	BOX DRAWINGS DOWN DOUBLE AND HORIZONTAL SINGLE
<DH>	<U2566>	BOX DRAWINGS DOUBLE DOWN AND HORIZONTAL
<uH>	<U2567>	BOX DRAWINGS UP SINGLE AND HORIZONTAL DOUBLE
<Uh>	<U2568>	BOX DRAWINGS UP DOUBLE AND HORIZONTAL SINGLE
<UH>	<U2569>	BOX DRAWINGS DOUBLE UP AND HORIZONTAL
<vH>	<U256A>	BOX DRAWINGS VERTICAL SINGLE AND HORIZONTAL DOUBLE
<Vh>	<U256B>	BOX DRAWINGS VERTICAL DOUBLE AND HORIZONTAL SINGLE
<VH>	<U256C>	BOX DRAWINGS DOUBLE VERTICAL AND HORIZONTAL
<FD>	<U2571>	BOX DRAWINGS LIGHT DIAGONAL UPPER RIGHT TO LOWER LEFT
<BD>	<U2572>	BOX DRAWINGS LIGHT DIAGONAL UPPER LEFT TO LOWER RIGHT
<TB>	<U2580>	UPPER HALF BLOCK
<LB>	<U2584>	LOWER HALF BLOCK
<FB>	<U2588>	FULL BLOCK
<lB>	<U258C>	LEFT HALF BLOCK
<RB>	<U2590>	RIGHT HALF BLOCK
<.S>	<U2591>	LIGHT SHADE
<:S>	<U2592>	MEDIUM SHADE
<?S>	<U2593>	DARK SHADE
<fS>	<U25A0>	BLACK SQUARE
<oS>	<U25A1>	WHITE SQUARE
<RO>	<U25A2>	WHITE SQUARE WITH ROUNDED CORNERS
<Rr>	<U25A3>	WHITE SQUARE CONTAINING BLACK SMALL SQUARE
<RF>	<U25A4>	SQUARE WITH HORIZONTAL FILL
<RY>	<U25A5>	SQUARE WITH VERTICAL FILL
<RH>	<U25A6>	SQUARE WITH ORTHOGONAL CROSSHATCH FILL
<RZ>	<U25A7>	SQUARE WITH UPPER LEFT TO LOWER RIGHT FILL
<RK>	<U25A8>	SQUARE WITH UPPER RIGHT TO LOWER LEFT FILL
<RX>	<U25A9>	SQUARE WITH DIAGONAL CROSSHATCH FILL
<sB>	<U25AA>	BLACK SMALL SQUARE
<SR>	<U25AC>	BLACK RECTANGLE
<Or>	<U25AD>	WHITE RECTANGLE
<UT>	<U25B2>	BLACK UP-POINTING TRIANGLE
<uT>	<U25B3>	WHITE UP-POINTING TRIANGLE
<Tr>	<U25B7>	WHITE RIGHT-POINTING TRIANGLE
<PR>	<U25BA>	BLACK RIGHT-POINTING POINTER
<dt>	<U25BC>	BLACK DOWN-POINTING TRIANGLE
<dT>	<U25BD>	WHITE DOWN-POINTING TRIANGLE
<TL>	<U25C1>	WHITE LEFT-POINTING TRIANGLE
<PL>	<U25C4>	BLACK LEFT-POINTING POINTER
<Db>	<U25C6>	BLACK DIAMOND
<Dw>	<U25C7>	WHITE DIAMOND
<LZ>	<U25CA>	LOZENGE
<Om>	<U25CB>	WHITE CIRCLE
<Oo>	<U25CE>	BULLSEYE
<Om>	<U25CF>	BLACK CIRCLE
	<U25D0>	CIRCLE WITH LEFT HALF BLACK
<OR>	<U25D1>	CIRCLE WITH RIGHT HALF BLACK
<Sn>	<U25D8>	INVERSE BULLET
<Ic>	<U25D9>	INVERSE WHITE CIRCLE
<Fd>	<U25E2>	BLACK LOWER RIGHT TRIANGLE
<Bd>	<U25E3>	BLACK LOWER LEFT TRIANGLE
<Ci>	<U25EF>	LARGE CIRCLE
<*2>	<U2605>	BLACK STAR
<*1>	<U2606>	WHITE STAR
<TEL>	<U260E>	BLACK TELEPHONE
<tel>	<U260F>	WHITE TELEPHONE
<<H>	<U261C>	WHITE LEFT POINTING INDEX
</>H>	<U261E>	WHITE RIGHT POINTING INDEX
<Ou>	<U263A>	WHITE SMILING FACE
<OU>	<U263B>	BLACK SMILING FACE
<SU>	<U263C>	WHITE SUN WITH RAYS
<Fm>	<U2640>	FEMALE SIGN
<Ml>	<U2642>	MALE SIGN
<cS>	<U2660>	BLACK SPADE SUIT
<cH>	<U2661>	WHITE HEART SUIT
<cD>	<U2662>	WHITE DIAMOND SUIT
<cC>	<U2663>	BLACK CLUB SUIT

5700	<cS->	<U2664>	WHITE SPADE SUIT
5701	<cH->	<U2665>	BLACK HEART SUIT
5702	<cD->	<U2666>	BLACK DIAMOND SUIT
5703	<cC->	<U2667>	WHITE CLUB SUIT
5704	<Md>	<U2669>	QUARTER NOTE
5705	<M8>	<U266A>	EIGHTH NOTE
5706	<M2>	<U266B>	BEAMED EIGHTH NOTES
5707	<M16>	<U266C>	BEAMED SIXTEENTH NOTES
5708	<Mb>	<U266D>	MUSIC FLAT SIGN
5709	<Mx>	<U266E>	MUSIC NATURAL SIGN
5710	<MX>	<U266F>	MUSIC SHARP SIGN
5711	<OK>	<U2713>	CHECK MARK
5712	<XX>	<U2717>	BALLOT X
5713	<-X>	<U2720>	MALTESE CROSS
5714	<IS>	<U3000>	IDEOGRAPHIC SPACE
5715	< , _>	<U3001>	IDEOGRAPHIC COMMA
5716	< . _>	<U3002>	IDEOGRAPHIC FULL STOP
5717	<+ ">	<U3003>	DITTO MARK
5718	<JIS>	<U3004>	JAPANESE INDUSTRIAL STANDARD SYMBOL
5719	<* _>	<U3005>	IDEOGRAPHIC ITERATION MARK
5720	< ; _>	<U3006>	IDEOGRAPHIC CLOSING MARK
5721	<0 _>	<U3007>	IDEOGRAPHIC NUMBER ZERO
5722	<<+>	<U300A>	LEFT DOUBLE ANGLE BRACKET
5723	</>+>	<U300B>	RIGHT DOUBLE ANGLE BRACKET
5724	<<'>	<U300C>	LEFT CORNER BRACKET
5725	</>'>	<U300D>	RIGHT CORNER BRACKET
5726	<<">	<U300E>	LEFT WHITE CORNER BRACKET
5727	</>">	<U300F>	RIGHT WHITE CORNER BRACKET
5728	<(">	<U3010>	LEFT BLACK LENTICULAR BRACKET
5729	<)">	<U3011>	RIGHT BLACK LENTICULAR BRACKET
5730	<=T>	<U3012>	POSTAL MARK
5731	<=_>	<U3013>	GETA MARK
5732	<('>	<U3014>	LEFT TORTOISE SHELL BRACKET
5733	<)'>	<U3015>	RIGHT TORTOISE SHELL BRACKET
5734	<('I>	<U3016>	LEFT WHITE LENTICULAR BRACKET
5735	<)'I>	<U3017>	RIGHT WHITE LENTICULAR BRACKET
5736	<-?>	<U301C>	WAVE DASH
5737	<=T:)>	<U3020>	POSTAL MARK FACE
5738	<A5>	<U3041>	HIRAGANA LETTER SMALL A
5739	<a5>	<U3042>	HIRAGANA LETTER A
5740	<I5>	<U3043>	HIRAGANA LETTER SMALL I
5741	<i5>	<U3044>	HIRAGANA LETTER I
5742	<U5>	<U3045>	HIRAGANA LETTER SMALL U
5743	<u5>	<U3046>	HIRAGANA LETTER U
5744	<E5>	<U3047>	HIRAGANA LETTER SMALL E
5745	<e5>	<U3048>	HIRAGANA LETTER E
5746	<O5>	<U3049>	HIRAGANA LETTER SMALL O
5747	<o5>	<U304A>	HIRAGANA LETTER O
5748	<ka>	<U304B>	HIRAGANA LETTER KA
5749	<ga>	<U304C>	HIRAGANA LETTER GA
5750	<ki>	<U304D>	HIRAGANA LETTER KI
5751	<gi>	<U304E>	HIRAGANA LETTER GI
5752	<ku>	<U304F>	HIRAGANA LETTER KU
5753	<gu>	<U3050>	HIRAGANA LETTER GU
5754	<ke>	<U3051>	HIRAGANA LETTER KE
5755	<ge>	<U3052>	HIRAGANA LETTER GE
5756	<ko>	<U3053>	HIRAGANA LETTER KO
5757	<go>	<U3054>	HIRAGANA LETTER GO
5758	<sa>	<U3055>	HIRAGANA LETTER SA
5759	<za>	<U3056>	HIRAGANA LETTER ZA
5760	<si>	<U3057>	HIRAGANA LETTER SI
5761	<zi>	<U3058>	HIRAGANA LETTER ZI
5762	<su>	<U3059>	HIRAGANA LETTER SU
5763	<zu>	<U305A>	HIRAGANA LETTER ZU
5764	<se>	<U305B>	HIRAGANA LETTER SE
5765	<ze>	<U305C>	HIRAGANA LETTER ZE
5766	<so>	<U305D>	HIRAGANA LETTER SO
5767	<zo>	<U305E>	HIRAGANA LETTER ZO
5768	<ta>	<U305F>	HIRAGANA LETTER TA
5769	<da>	<U3060>	HIRAGANA LETTER DA
5770	<ti>	<U3061>	HIRAGANA LETTER TI
5771	<di>	<U3062>	HIRAGANA LETTER DI
5772	<tU>	<U3063>	HIRAGANA LETTER SMALL TU
5773	<tu>	<U3064>	HIRAGANA LETTER TU
5774	<du>	<U3065>	HIRAGANA LETTER DU
5775	<te>	<U3066>	HIRAGANA LETTER TE
5776	<de>	<U3067>	HIRAGANA LETTER DE
5777	<to>	<U3068>	HIRAGANA LETTER TO
5778	<do>	<U3069>	HIRAGANA LETTER DO
5779	<na>	<U306A>	HIRAGANA LETTER NA
5780	<ni>	<U306B>	HIRAGANA LETTER NI
5781	<nu>	<U306C>	HIRAGANA LETTER NU
5782	<ne>	<U306D>	HIRAGANA LETTER NE
5783	<no>	<U306E>	HIRAGANA LETTER NO
5784	<ha>	<U306F>	HIRAGANA LETTER HA
5785	<ba>	<U3070>	HIRAGANA LETTER BA
5786	<pa>	<U3071>	HIRAGANA LETTER PA
5787	<hi>	<U3072>	HIRAGANA LETTER HI

572	<bi>	<U3073>	HIRAGANA LETTER BI
573	<pi>	<U3074>	HIRAGANA LETTER PI
574	<hu>	<U3075>	HIRAGANA LETTER HU
575	<bu>	<U3076>	HIRAGANA LETTER BU
576	<pu>	<U3077>	HIRAGANA LETTER PU
577	<he>	<U3078>	HIRAGANA LETTER HE
578	<be>	<U3079>	HIRAGANA LETTER BE
579	<pe>	<U307A>	HIRAGANA LETTER PE
580	<ho>	<U307B>	HIRAGANA LETTER HO
581	<bo>	<U307C>	HIRAGANA LETTER BO
582	<po>	<U307D>	HIRAGANA LETTER PO
583	<ma>	<U307E>	HIRAGANA LETTER MA
584	<mi>	<U307F>	HIRAGANA LETTER MI
585	<mu>	<U3080>	HIRAGANA LETTER MU
586	<me>	<U3081>	HIRAGANA LETTER ME
587	<mo>	<U3082>	HIRAGANA LETTER MO
588	<yA>	<U3083>	HIRAGANA LETTER SMALL YA
589	<ya>	<U3084>	HIRAGANA LETTER YA
590	<yU>	<U3085>	HIRAGANA LETTER SMALL YU
591	<yu>	<U3086>	HIRAGANA LETTER YU
592	<yO>	<U3087>	HIRAGANA LETTER SMALL YO
593	<yo>	<U3088>	HIRAGANA LETTER YO
594	<ra>	<U3089>	HIRAGANA LETTER RA
595	<ri>	<U308A>	HIRAGANA LETTER RI
596	<ru>	<U308B>	HIRAGANA LETTER RU
597	<re>	<U308C>	HIRAGANA LETTER RE
598	<ro>	<U308D>	HIRAGANA LETTER RO
599	<wA>	<U308E>	HIRAGANA LETTER SMALL WA
600	<wa>	<U308F>	HIRAGANA LETTER WA
601	<wi>	<U3090>	HIRAGANA LETTER WI
602	<we>	<U3091>	HIRAGANA LETTER WE
603	<wo>	<U3092>	HIRAGANA LETTER WO
604	<n5>	<U3093>	HIRAGANA LETTER N
605	<vu>	<U3094>	HIRAGANA LETTER VU
606	<"5>	<U309B>	KATAKANA-HIRAGANA VOICED SOUND MARK
607	<05>	<U309C>	KATAKANA-HIRAGANA SEMI-VOICED SOUND MARK
608	<*5>	<U309D>	HIRAGANA ITERATION MARK
609	<+5>	<U309E>	HIRAGANA VOICED ITERATION MARK
610	<a6>	<U30A1>	KATAKANA LETTER SMALL A
611	<A6>	<U30A2>	KATAKANA LETTER A
612	<i6>	<U30A3>	KATAKANA LETTER SMALL I
613	<I6>	<U30A4>	KATAKANA LETTER I
614	<u6>	<U30A5>	KATAKANA LETTER SMALL U
615	<U6>	<U30A6>	KATAKANA LETTER U
616	<e6>	<U30A7>	KATAKANA LETTER SMALL E
617	<E6>	<U30A8>	KATAKANA LETTER E
618	<o6>	<U30A9>	KATAKANA LETTER SMALL O
619	<O6>	<U30AA>	KATAKANA LETTER O
620	<Ka>	<U30AB>	KATAKANA LETTER KA
621	<Ga>	<U30AC>	KATAKANA LETTER GA
622	<Ki>	<U30AD>	KATAKANA LETTER KI
623	<Gi>	<U30AE>	KATAKANA LETTER GI
624	<Ku>	<U30AF>	KATAKANA LETTER KU
625	<Gu>	<U30B0>	KATAKANA LETTER GU
626	<Ke>	<U30B1>	KATAKANA LETTER KE
627	<Ge>	<U30B2>	KATAKANA LETTER GE
628	<Ko>	<U30B3>	KATAKANA LETTER KO
629	<Go>	<U30B4>	KATAKANA LETTER GO
630	<Sa>	<U30B5>	KATAKANA LETTER SA
631	<Za>	<U30B6>	KATAKANA LETTER ZA
632	<Si>	<U30B7>	KATAKANA LETTER SI
633	<Zi>	<U30B8>	KATAKANA LETTER ZI
634	<Su>	<U30B9>	KATAKANA LETTER SU
635	<Zu>	<U30BA>	KATAKANA LETTER ZU
636	<Se>	<U30BB>	KATAKANA LETTER SE
637	<Ze>	<U30BC>	KATAKANA LETTER ZE
638	<So>	<U30BD>	KATAKANA LETTER SO
639	<Zo>	<U30BE>	KATAKANA LETTER ZO
640	<Ta>	<U30BF>	KATAKANA LETTER TA
641	<Da>	<U30C0>	KATAKANA LETTER DA
642	<Ti>	<U30C1>	KATAKANA LETTER TI
643	<Di>	<U30C2>	KATAKANA LETTER DI
644	<TU>	<U30C3>	KATAKANA LETTER SMALL TU
645	<Tu>	<U30C4>	KATAKANA LETTER TU
646	<Du>	<U30C5>	KATAKANA LETTER DU
647	<Te>	<U30C6>	KATAKANA LETTER TE
648	<De>	<U30C7>	KATAKANA LETTER DE
649	<To>	<U30C8>	KATAKANA LETTER TO
650	<Do>	<U30C9>	KATAKANA LETTER DO
651	<Na>	<U30CA>	KATAKANA LETTER NA
652	<Ni>	<U30CB>	KATAKANA LETTER NI
653	<Nu>	<U30CC>	KATAKANA LETTER NU
654	<Ne>	<U30CD>	KATAKANA LETTER NE
655	<No>	<U30CE>	KATAKANA LETTER NO
656	<Ha>	<U30CF>	KATAKANA LETTER HA
657	<Ba>	<U30D0>	KATAKANA LETTER BA
658	<Pa>	<U30D1>	KATAKANA LETTER PA
659	<Hi>	<U30D2>	KATAKANA LETTER HI

<Bi>	<U30D3>	KATAKANA LETTER BI
<Pi>	<U30D4>	KATAKANA LETTER PI
<Hu>	<U30D5>	KATAKANA LETTER HU
<Bu>	<U30D6>	KATAKANA LETTER BU
<Pu>	<U30D7>	KATAKANA LETTER PU
<He>	<U30D8>	KATAKANA LETTER HE
<Be>	<U30D9>	KATAKANA LETTER BE
<Pe>	<U30DA>	KATAKANA LETTER PE
<Ho>	<U30DB>	KATAKANA LETTER HO
<Bo>	<U30DC>	KATAKANA LETTER BO
<Po>	<U30DD>	KATAKANA LETTER PO
<Ma>	<U30DE>	KATAKANA LETTER MA
<Mi>	<U30DF>	KATAKANA LETTER MI
<Mu>	<U30E0>	KATAKANA LETTER MU
<Me>	<U30E1>	KATAKANA LETTER ME
<Mo>	<U30E2>	KATAKANA LETTER MO
<YA>	<U30E3>	KATAKANA LETTER SMALL YA
<Ya>	<U30E4>	KATAKANA LETTER YA
<YU>	<U30E5>	KATAKANA LETTER SMALL YU
<Yu>	<U30E6>	KATAKANA LETTER YU
<YO>	<U30E7>	KATAKANA LETTER SMALL YO
<Yo>	<U30E8>	KATAKANA LETTER YO
<Ra>	<U30E9>	KATAKANA LETTER RA
<Ri>	<U30EA>	KATAKANA LETTER RI
<Ru>	<U30EB>	KATAKANA LETTER RU
<Re>	<U30EC>	KATAKANA LETTER RE
<Ro>	<U30ED>	KATAKANA LETTER RO
<WA>	<U30EE>	KATAKANA LETTER SMALL WA
<Wa>	<U30EF>	KATAKANA LETTER WA
<Wi>	<U30F0>	KATAKANA LETTER WI
<We>	<U30F1>	KATAKANA LETTER WE
<Wo>	<U30F2>	KATAKANA LETTER WO
<N6>	<U30F3>	KATAKANA LETTER N
<Vu>	<U30F4>	KATAKANA LETTER VU
<KA>	<U30F5>	KATAKANA LETTER SMALL KA
<KE>	<U30F6>	KATAKANA LETTER SMALL KE
<Va>	<U30F7>	KATAKANA LETTER VA
<Vi>	<U30F8>	KATAKANA LETTER VI
<Ve>	<U30F9>	KATAKANA LETTER VE
<Vo>	<U30FA>	KATAKANA LETTER VO
<.6>	<U30FB>	KATAKANA MIDDLE DOT
<-6>	<U30FC>	KATAKANA-HIRAGANA PROLONGED SOUND MARK
<*6>	<U30FD>	KATAKANA ITERATION MARK
<+6>	<U30FE>	KATAKANA VOICED ITERATION MARK
<b4>	<U3105>	BOPOMOFO LETTER B
<p4>	<U3106>	BOPOMOFO LETTER P
<m4>	<U3107>	BOPOMOFO LETTER M
<f4>	<U3108>	BOPOMOFO LETTER F
<d4>	<U3109>	BOPOMOFO LETTER D
<t4>	<U310A>	BOPOMOFO LETTER T
<n4>	<U310B>	BOPOMOFO LETTER N
<l4>	<U310C>	BOPOMOFO LETTER L
<g4>	<U310D>	BOPOMOFO LETTER G
<k4>	<U310E>	BOPOMOFO LETTER K
<h4>	<U310F>	BOPOMOFO LETTER H
<j4>	<U3110>	BOPOMOFO LETTER J
<q4>	<U3111>	BOPOMOFO LETTER Q
<x4>	<U3112>	BOPOMOFO LETTER X
<zh>	<U3113>	BOPOMOFO LETTER ZH
<ch>	<U3114>	BOPOMOFO LETTER CH
<sh>	<U3115>	BOPOMOFO LETTER SH
<r4>	<U3116>	BOPOMOFO LETTER R
<z4>	<U3117>	BOPOMOFO LETTER Z
<c4>	<U3118>	BOPOMOFO LETTER C
<s4>	<U3119>	BOPOMOFO LETTER S
<a4>	<U311A>	BOPOMOFO LETTER A
<o4>	<U311B>	BOPOMOFO LETTER O
<e4>	<U311C>	BOPOMOFO LETTER E
<eh4>	<U311D>	BOPOMOFO LETTER EH
<ai>	<U311E>	BOPOMOFO LETTER AI
<ei>	<U311F>	BOPOMOFO LETTER EI
<au>	<U3120>	BOPOMOFO LETTER AU
<ou>	<U3121>	BOPOMOFO LETTER OU
<an>	<U3122>	BOPOMOFO LETTER AN
<en>	<U3123>	BOPOMOFO LETTER EN
<aN>	<U3124>	BOPOMOFO LETTER ANG
<eN>	<U3125>	BOPOMOFO LETTER ENG
<er>	<U3126>	BOPOMOFO LETTER ER
<i4>	<U3127>	BOPOMOFO LETTER I
<u4>	<U3128>	BOPOMOFO LETTER U
<iu>	<U3129>	BOPOMOFO LETTER IU
<v4>	<U312A>	BOPOMOFO LETTER V
<nG>	<U312B>	BOPOMOFO LETTER NG
<gn>	<U312C>	BOPOMOFO LETTER GN
<(JU)>	<U321C>	PARENTHESIZED HANGUL CIEUC U
<1c>	<U3220>	PARENTHESIZED IDEOGRAPH ONE
<2c>	<U3221>	PARENTHESIZED IDEOGRAPH TWO
<3c>	<U3222>	PARENTHESIZED IDEOGRAPH THREE

<4c>	<U3223>	PARENTHESES IDEOGRAPH FOUR
<5c>	<U3224>	PARENTHESES IDEOGRAPH FIVE
<6c>	<U3225>	PARENTHESES IDEOGRAPH SIX
<7c>	<U3226>	PARENTHESES IDEOGRAPH SEVEN
<8c>	<U3227>	PARENTHESES IDEOGRAPH EIGHT
<9c>	<U3228>	PARENTHESES IDEOGRAPH NINE
<10c>	<U3229>	PARENTHESES IDEOGRAPH TEN
<KSC>	<U327F>	KOREAN STANDARD SYMBOL
<am>	<U33C2>	SQUARE AM
<pm>	<U33D8>	SQUARE PM
<ff>	<UFB00>	LATIN SMALL LIGATURE FF
<fi>	<UFB01>	LATIN SMALL LIGATURE FI
<fl>	<UFB02>	LATIN SMALL LIGATURE FL
<ffi>	<UFB03>	LATIN SMALL LIGATURE FFI
<ffl>	<UFB04>	LATIN SMALL LIGATURE FFL
<St>	<UFB05>	LATIN SMALL LIGATURE LONG S T
<st>	<UFB06>	LATIN SMALL LIGATURE ST
<3+; >	<UFE7D>	ARABIC SHADDA MEDIAL FORM
<aM. >	<UFE82>	ARABIC LETTER ALEF WITH MADDA ABOVE FINAL FORM
<aH. >	<UFE84>	ARABIC LETTER ALEF WITH HAMZA ABOVE FINAL FORM
<aH. >	<UFE88>	ARABIC LETTER ALEF WITH HAMZA BELOW FINAL FORM
<a+ >	<UFE8D>	ARABIC LETTER ALEF ISOLATED FORM
<a. >	<UFE8E>	ARABIC LETTER ALEF FINAL FORM
<b+ >	<UFE8F>	ARABIC LETTER BEH ISOLATED FORM
<b. >	<UFE90>	ARABIC LETTER BEH FINAL FORM
<b, >	<UFE91>	ARABIC LETTER BEH INITIAL FORM
<b, ; >	<UFE92>	ARABIC LETTER BEH MEDIAL FORM
<tm- >	<UFE93>	ARABIC LETTER TEH MARBUTA ISOLATED FORM
<tm. >	<UFE94>	ARABIC LETTER TEH MARBUTA FINAL FORM
<t+ >	<UFE95>	ARABIC LETTER TEH ISOLATED FORM
<t. >	<UFE96>	ARABIC LETTER TEH FINAL FORM
<t, >	<UFE97>	ARABIC LETTER TEH INITIAL FORM
<t, ; >	<UFE98>	ARABIC LETTER TEH MEDIAL FORM
<tk- >	<UFE99>	ARABIC LETTER THEH ISOLATED FORM
<tk. >	<UFE9A>	ARABIC LETTER THEH FINAL FORM
<tk, >	<UFE9B>	ARABIC LETTER THEH INITIAL FORM
<tk, ; >	<UFE9C>	ARABIC LETTER THEH MEDIAL FORM
<g+ >	<UFE9D>	ARABIC LETTER JEEM ISOLATED FORM
<g. >	<UFE9E>	ARABIC LETTER JEEM FINAL FORM
<g, >	<UFE9F>	ARABIC LETTER JEEM INITIAL FORM
<g, ; >	<UFEA0>	ARABIC LETTER JEEM MEDIAL FORM
<hk- >	<UFEA1>	ARABIC LETTER HAH ISOLATED FORM
<hk. >	<UFEA2>	ARABIC LETTER HAH FINAL FORM
<hk, >	<UFEA3>	ARABIC LETTER HAH INITIAL FORM
<hk, ; >	<UFEA4>	ARABIC LETTER HAH MEDIAL FORM
<x+ >	<UFEA5>	ARABIC LETTER KHAH ISOLATED FORM
<x. >	<UFEA6>	ARABIC LETTER KHAH FINAL FORM
<x, >	<UFEA7>	ARABIC LETTER KHAH INITIAL FORM
<x, ; >	<UFEA8>	ARABIC LETTER KHAH MEDIAL FORM
<d+ >	<UFEA9>	ARABIC LETTER DAL ISOLATED FORM
<d. >	<UFEAA>	ARABIC LETTER DAL FINAL FORM
<dk- >	<UFEAB>	ARABIC LETTER THAL ISOLATED FORM
<dk. >	<UFEAC>	ARABIC LETTER THAL FINAL FORM
<r+ >	<UFEAD>	ARABIC LETTER REH ISOLATED FORM
<r. >	<UFEAE>	ARABIC LETTER REH FINAL FORM
<z+ >	<UFEAF>	ARABIC LETTER ZAIN ISOLATED FORM
<z. >	<UFEB0>	ARABIC LETTER ZAIN FINAL FORM
<s+ >	<UFEB1>	ARABIC LETTER SEEN ISOLATED FORM
<s. >	<UFEB2>	ARABIC LETTER SEEN FINAL FORM
<s, >	<UFEB3>	ARABIC LETTER SEEN INITIAL FORM
<s, ; >	<UFEB4>	ARABIC LETTER SEEN MEDIAL FORM
<sn- >	<UFEB5>	ARABIC LETTER SHEEN ISOLATED FORM
<sn. >	<UFEB6>	ARABIC LETTER SHEEN FINAL FORM
<sn, >	<UFEB7>	ARABIC LETTER SHEEN INITIAL FORM
<sn, ; >	<UFEB8>	ARABIC LETTER SHEEN MEDIAL FORM
<c+ >	<UFEB9>	ARABIC LETTER SAD ISOLATED FORM
<c. >	<UFEBA>	ARABIC LETTER SAD FINAL FORM
<c, >	<UFEBB>	ARABIC LETTER SAD INITIAL FORM
<c, ; >	<UFEBC>	ARABIC LETTER SAD MEDIAL FORM
<dd- >	<UFEBD>	ARABIC LETTER DAD ISOLATED FORM
<dd. >	<UFEBE>	ARABIC LETTER DAD FINAL FORM
<dd, >	<UFEBF>	ARABIC LETTER DAD INITIAL FORM
<dd, ; >	<UFEC0>	ARABIC LETTER DAD MEDIAL FORM
<tj- >	<UFEC1>	ARABIC LETTER TAH ISOLATED FORM
<tj. >	<UFEC2>	ARABIC LETTER TAH FINAL FORM
<tj, >	<UFEC3>	ARABIC LETTER TAH INITIAL FORM
<tj, ; >	<UFEC4>	ARABIC LETTER TAH MEDIAL FORM
<zH- >	<UFEC5>	ARABIC LETTER ZAH ISOLATED FORM
<zH. >	<UFEC6>	ARABIC LETTER ZAH FINAL FORM
<zH, >	<UFEC7>	ARABIC LETTER ZAH INITIAL FORM
<zH, ; >	<UFEC8>	ARABIC LETTER ZAH MEDIAL FORM
<e+ >	<UFEC9>	ARABIC LETTER AIN ISOLATED FORM
<e. >	<UFECA>	ARABIC LETTER AIN FINAL FORM
<e, >	<UFECB>	ARABIC LETTER AIN INITIAL FORM
<e, ; >	<UFEC<	ARABIC LETTER AIN MEDIAL FORM
<i+ >	<UFECD>	ARABIC LETTER GHAIN ISOLATED FORM
<i. >	<UFECE>	ARABIC LETTER GHAIN FINAL FORM
<i, >	<UFECF>	ARABIC LETTER GHAIN INITIAL FORM

59	<i+;>	<UFED0>	ARABIC LETTER GHAIN MEDIAL FORM
60	<f+>	<UFED1>	ARABIC LETTER FEH ISOLATED FORM
61	<f+.>	<UFED2>	ARABIC LETTER FEH FINAL FORM
62	<f+,>	<UFED3>	ARABIC LETTER FEH INITIAL FORM
63	<f+;>	<UFED4>	ARABIC LETTER FEH MEDIAL FORM
64	<q+>	<UFED5>	ARABIC LETTER QAF ISOLATED FORM
65	<q+.>	<UFED6>	ARABIC LETTER QAF FINAL FORM
66	<q+,>	<UFED7>	ARABIC LETTER QAF INITIAL FORM
67	<q+;>	<UFED8>	ARABIC LETTER QAF MEDIAL FORM
68	<k+>	<UFED9>	ARABIC LETTER KAF ISOLATED FORM
69	<k+.>	<UFEDA>	ARABIC LETTER KAF FINAL FORM
70	<k+,>	<UFEDB>	ARABIC LETTER KAF INITIAL FORM
71	<k+;>	<UFEDC>	ARABIC LETTER KAF MEDIAL FORM
72	<l+>	<UFEDD>	ARABIC LETTER LAM ISOLATED FORM
73	<l+.>	<UFEDE>	ARABIC LETTER LAM FINAL FORM
74	<l+,>	<UFEDF>	ARABIC LETTER LAM INITIAL FORM
75	<l+;>	<UFEE0>	ARABIC LETTER LAM MEDIAL FORM
76	<m+>	<UFEE1>	ARABIC LETTER MEEM ISOLATED FORM
77	<m+.>	<UFEE2>	ARABIC LETTER MEEM FINAL FORM
78	<m+,>	<UFEE3>	ARABIC LETTER MEEM INITIAL FORM
79	<m+;>	<UFEE4>	ARABIC LETTER MEEM MEDIAL FORM
80	<n+>	<UFEE5>	ARABIC LETTER NOON ISOLATED FORM
81	<n+.>	<UFEE6>	ARABIC LETTER NOON FINAL FORM
82	<n+,>	<UFEE7>	ARABIC LETTER NOON INITIAL FORM
83	<n+;>	<UFEE8>	ARABIC LETTER NOON MEDIAL FORM
84	<h+>	<UFEE9>	ARABIC LETTER HEH ISOLATED FORM
85	<h+.>	<UFEEA>	ARABIC LETTER HEH FINAL FORM
86	<h+,>	<UFEEB>	ARABIC LETTER HEH INITIAL FORM
87	<h+;>	<UFEEC>	ARABIC LETTER HEH MEDIAL FORM
88	<w+>	<UFEEED>	ARABIC LETTER WAW ISOLATED FORM
89	<w+.>	<UFEEEE>	ARABIC LETTER WAW FINAL FORM
90	<j+>	<UFEEF>	ARABIC LETTER ALEF MAKSURA ISOLATED FORM
91	<j+.>	<UFEEF0>	ARABIC LETTER ALEF MAKSURA FINAL FORM
92	<y+>	<UFEEF1>	ARABIC LETTER YEH ISOLATED FORM
93	<y+.>	<UFEEF2>	ARABIC LETTER YEH FINAL FORM
94	<y+,>	<UFEEF3>	ARABIC LETTER YEH INITIAL FORM
95	<y+;>	<UFEEF4>	ARABIC LETTER YEH MEDIAL FORM
96	<lm->	<UFEEF5>	ARABIC LIGATURE LAM WITH ALEF WITH MADDA ABOVE ISOLATED FORM
97	<lm.>	<UFEEF6>	ARABIC LIGATURE LAM WITH ALEF WITH MADDA ABOVE FINAL FORM
98	<lh->	<UFEEF7>	ARABIC LIGATURE LAM WITH ALEF WITH HAMZA ABOVE ISOLATED FORM
99	<lh.>	<UFEEF8>	ARABIC LIGATURE LAM WITH ALEF WITH HAMZA ABOVE FINAL FORM
100	<lh->	<UFEEF9>	ARABIC LIGATURE LAM WITH ALEF WITH HAMZA BELOW ISOLATED FORM
101	<lh.>	<UFEEFA>	ARABIC LIGATURE LAM WITH ALEF WITH HAMZA BELOW FINAL FORM
102	<la->	<UFEEFB>	ARABIC LIGATURE LAM WITH ALEF ISOLATED FORM
103	<la.>	<UFEEFC>	ARABIC LIGATURE LAM WITH ALEF FINAL FORM
104	<H->	<U0023>	NUMBER SIGN
105	<!S>	<U0024>	DOLLAR SIGN
106	<@>	<U0040>	COMMERCIAL AT
107	<Oa>	<U0040>	COMMERCIAL AT
108	<!C>	<U00A2>	CENT SIGN
109	<L->	<U00A3>	POUND SIGN
110	<Xo>	<U00A4>	CURRENCY SIGN
111	<Y->	<U00A5>	YEN SIGN
112	<!B>	<U00A6>	BROKEN BAR
113	<So>	<U00A7>	SECTION SIGN
114	<?!>	<U00AC>	NOT SIGN
115	<9I>	<U00B6>	PILCROW SIGN
116	<_>	<U2500>	BOX DRAWINGS LIGHT HORIZONTAL
117	<=>	<U2501>	BOX DRAWINGS HEAVY HORIZONTAL
118	<_!>	<U2502>	BOX DRAWINGS LIGHT VERTICAL
119	<_V/>>	<U250C>	BOX DRAWINGS LIGHT DOWN AND RIGHT
120	<_V<w>	<U2510>	BOX DRAWINGS LIGHT DOWN AND LEFT
121	<_A/>>	<U2514>	BOX DRAWINGS LIGHT UP AND RIGHT
122	<_A<	<U2518>	BOX DRAWINGS LIGHT UP AND LEFT
123	<_!/>>	<U251C>	BOX DRAWINGS LIGHT VERTICAL AND RIGHT
124	<_!<	<U2524>	BOX DRAWINGS LIGHT VERTICAL AND LEFT
125	<_V->	<U252C>	BOX DRAWINGS LIGHT DOWN AND HORIZONTAL
126	<_A->	<U2534>	BOX DRAWINGS LIGHT UP AND HORIZONTAL
127	<_!->	<U253C>	BOX DRAWINGS LIGHT VERTICAL AND HORIZONTAL
128	<_//>>	<U2571>	BOX DRAWINGS LIGHT DIAGONAL UPPER RIGHT TO LOWER LEFT
129	<_<	<U2572>	BOX DRAWINGS LIGHT DIAGONAL UPPER LEFT TO LOWER RIGHT
130	<_./>>	<U25E2>	BLACK LOWER RIGHT TRIANGLE
131	<_<	<U25E3>	BLACK LOWER LEFT TRIANGLE
132	<_d!>	<U266A>	EIGHTH NOTE

6069 **7 CONFORMANCE** (controversial)

6070

6071 **7.1 FDCC-set**

6072

6073 A FDCC-set description is conforming to this Technical Report if it meets the
6074 requirements in clause 4.

6075

6076 **7.2 FDCC-set category**

6077

6078 Conformance can be claimed for a category description against each of the clauses 4.3
6079 thru 4.12, and then the requirements of clause 4.1 are also met, and a
6080 LC_IDENTIFICATION category as described in clause 4.2 is specified.

6081

6082 **7.3 Charmap**

6083

6084 A charmap description is conforming to this Technical Report if it meets the requirements
6085 in clause 5.

6086

6087 **7.4 Repertoiremap**

6088

6089 A repertoiremap description is conforming to this Technical Report if it meets the
6090 requirements in clause 6.

Annex A
(informative)

Differences from the ISO/IEC 9945-2 standard

This Technical Report originated from the locale and charmap specifications in the ISO/IEC 9945-2 POSIX shell and utilities standard, and it intends to be backwards compatible, so that what is conformant to that standard should also be conformant to this Technical Report.

A number of enhancements have been made and a number of restrictions have been lifted in comparison to the POSIX standard:

A.1 Restrictions removed

1. Dependence on specific meaning of the character NUL as termination of a string (from the C standard) has been removed, to cater for other programming languages than C.

A.2 Enhancements

1. A description of a "repertoiremap" definition was added to facilitate descriptions of FDCC-sets without charmaps, and also to provide binding from a FDCC-set using one set of character names to charmaps using another naming set.

2. The specific POSIX locale has been replaced with the "i18n" FDCC-set, defined on the repertoire on ISO/IEC 10646.

3. Transliteration support has been added in the LC_CTYPE category.

4. Terminology has been aligned with ISO/IEC TR 11017, especially the POSIX term "locale" has been changed to "FDCC-set".

5. A date escape format "%F" has been added for ISO 8601 dates, and another date escape format "%f" has been added for weekday number with Monday being the first day of the week.

6. Added to LC_MONETARY to accommodate differences between local and international formats:

```
int_p_cs_precedes
int_p_sep_by_space
int_n_cs_precedes
int_n_sep_by_space
```

7. Section symbols have been added via the "section-symbol" keyword in the LC_COLLATE category.

8. The "order_start" keyword has got an optional "section-symbol" identifier

9. The keywords "reorder-section-after" and "reorder-section_end" have been introduced to reorder sections.

10. Symbolic ellipses (both decimal and hexadecimal) has been introduced as a notation.

- 6143 11. The "print" CTYPE class includes automatically all "graph" characters.
6144
- 6145 12. The <Uxxxx> and <Uxxxxxxxx> notations have been introduced as predefined
6146 symbolic character names, together with a number of symbolic character names derived
6147 from POSIX and the Internet.
6148
- 6149 13. New categories LC_IDENTIFICATION, LC_XLITERATE, LC_NAME,
6150 LC_ADDRESS, and LC_TELEPHONE, have been introduced.
6151
- 6152 14. The LC_CTYPE has got support for new classes, via the new keywords class and
6153 map, which corresponds to the C standard library functions iswctype() and towctrans()
6154 respectively.
6155
- 6156 15. The "digit" keyword now supports digits for multiple scripts.
6157
- 6158 16. The LC_MONETARY category provides support for multiple currencies, such as the
6159 native currency and the Euro in some European countries.
6160
- 6161 17. The LC_TIME has got a number of enhancements to cater for alternate calendars, and
6162 timezone information may be given.
6163
- 6164 18. The charmap specification has been enhanced to support ISO 2022.

6165
6166
6167
6168
6169
6170
6171
6172
6173
6174
6175
6176
6177
6178
6179
6180
6181
6182
6183
6184
6185
6186
6187
6188
6189
6190
6191
6192
6193
6194
6195
6196
6197
6198
6199
6200
6201
6202
6203
6204
6205
6206
6207
6208
6209
6210
6211
6212
6213
6214
6215
6216

Annex B (informative)

Rationale

B.1 FDCC-set Rationale

The description of FDCC-sets is based on work performed in the UniForum Technical Committee Subcommittee on Internationalisation and POSIX. Wherever appropriate, keywords were taken from the C Standard or the ISO/IEC 9945-2:1993 POSIX standard. The C and POSIX term "locale" has been changed into the term "FDCC-set" from ISO/IEC TR 11017 to align with that specification.

The POSIX utility "localedef" compiles locale sources into object files. The "object" definitions need not be portable, as long as "source" definitions are. Strictly speaking, "source" definitions are portable only between applications using the same character set(s). Such "source" definitions can, if they use symbolic names only, easily be ported between systems using different code sets as long as the characters in the portable character set (ISO 646) have common values between the code sets; this is frequently the case in historical applications. Of course, this requires that the symbolic names used for characters outside the portable character set are identical between character sets.

To avoid confusion between an octal constant and a backreference, the octal, hexadecimal, and decimal constants must contain at least two digits. As single-digit constants are relatively rare, this should not impose any significant hardship. Each of the constants includes "two or more" digits to account for systems in which the byte size is larger than eight bits. For example, an ISO/IEC 10646 system that has defined 16-bit bytes may require six octal, four hexadecimal, and five decimal digits, for some coded characters.

As an international (ISO/IEC) Technical Report this Technical Report should follow the ISO/IEC guidelines, including the ISO/IEC TR 10176. This TR has a rule that characters outside the invariant part of ISO/IEC 646 should not be used in portable specifications. The backslash and the number-sign character are not in the invariant part. As far as general usage of these symbols, they are covered by the "grandfather clause" specifying previous practise in international standards and in the industry such as in specifications from The Open Group, but for newly defined interfaces, ISO has requested that specifications provide alternate representations, and this Technical Report then follows POSIX for backward compatibility. Consequently, while the default escape character remains the backslash, and the default comment character is the number-sign, applications are required to recognize alternative representations, identified in the applicable source text via the "escape_char" and "comment_char" keywords.

B.1.1 LC_IDENTIFICATION Rationale.

The LC_IDENTIFICATION category gives meta-information on the FDCC-set, such as who created it, and what is the level of conformance for each of the FDCC sets.

B.1.2 LC_CTYPE Rationale

The LC_CTYPE category primarily is used to define the encoding-independent aspects of a character set, such as character classification. In addition, certain encoding-dependent characteristics are also defined for an application via the LC_CTYPE category. This

6217 Technical Report does not mandate that the encoding used in the FDCC-set is the same as
 6218 the one used by the application, because an application may decide that it is advantageous
 6219 to define a FDCC-set in a system-wide encoding rather than having multiple, logically
 6220 identical FDCC-sets in different encodings, and to convert from the application encoding
 6221 to the system-wide encoding on usage. Other applications could require encoding-depend-
 6222 ent FDCC-sets. In either case, the LC_CTYPE attributes that are directly dependent on
 6223 the encoding, such as "mb_cur_max" and the display width of characters, are not user-
 6224 specifiable in a locale source, and are consequently not defined as keywords.

6225
 6226 As the LC_CTYPE character classes are based on the C Standard character-class
 6227 definition, the category does not support multicharacter elements. For instance, the
 6228 German character <sharp-s> is traditionally classified as a lowercase letter. There is no
 6229 corresponding uppercase letter; in proper capitalization of German text the <sharp-s> will
 6230 be replaced by SS; i.e., by two characters. This kind of conversion is outside the scope of
 6231 the "toupper" and "tolower" keywords.

6232
 6233 The character classes "digit", "xdigit", "lower", "upper", and "space" have a set of
 6234 automatically included characters. These only need to be specified if the character values
 6235 (i.e. encoding) differs from the application default values. The definition of character class
 6236 "digit" allows alternate digits (e.g., Hindi) to be specified here. The definition of character
 6237 class "xdigit" requires that the characters included in character class "digit" are included
 6238 here also, and allows for different symbols for the hexadecimal digits 10 through 15.

6239
 6240 The "combining" and "combining-level3" classes are an IT-enablement of ISO/IEC 10646
 6241 definitions of combining characters. These can be used to check identifiers for consistence
 6242 with the guidelines given in TR 10176 annex A.

6243 6244 **B.1.3 LC_COLLATE Rationale.**

6245
 6246 The LC_COLLATE category governs the collation order in the FDCC-set, and may thus
 6247 be useful for the processing of the ISO/IEC 14651 string ordering and comparison
 6248 standard, the C Standard strxfrm() and strcoll() functions, as well as a number of ISO/IEC
 6249 9945-2:1993 POSIX utilities.

6250
 6251 The rules governing collation depends to some extent on the use. At least five different
 6252 levels of increasingly complex collation rules can be distinguished:

- 6253
6254
- 6255 (1) Byte/machine code order. This is the historical collation order in the UNIX
 6256 system and many proprietary operating systems. Collation is here done
 6257 character by character, without any regard to context. The primary virtue is that
 6258 it usually is quite fast, and also completely deterministic; it works well when
 6259 the native machine collation sequence matches the user expectations.
 - 6260 (2) Character order. On this level, collation is also done character by character,
 6261 without regard to context. The order between characters is, however, not deter-
 6262 mined by the code values, but on the user's expectations of the correct order
 6263 between characters. In addition, such a (simple) collation order can specify that
 6264 certain characters collate equal (e.g., upper and lowercase letters).
 - 6265 (3) String ordering. On this level, entire strings are compared based on relatively
 6266 straightforward rules. At this level, several "passes" may be required to deter-
 6267 mine the order between two strings. Characters may be ignored in some passes,
 6268 but not in others; the strings may be compared in different directions; and

- 6269 simple string substitutions may be made before strings are compared. This level
 6270 is best described as "dictionary" ordering; it is based on the spelling, not the
 6271 pronunciation, or meaning, of the words.
- (4) 6272 Text search ordering. This is a further refinement of the previous level, best de-
 6273 scribed as "telephone book ordering"; some common homonyms (words spelled
 6274 differently but with same pronunciation) are collated together; numbers are
 6275 collated as if spelled with words, and so on.
- (5) 6276 Semantic level ordering. Words and strings are collated based on their meaning;
 6277 entire words (such as "the") are eliminated, the ordering is not deterministic.
 6278 This may requires special software, and is highly dependent on the intended
 6279 use.

6280
 6281 While the historical collation order formally is at level 1, for the English language it
 6282 corresponds roughly to elements at level 2. The user expects to see the output from the
 6283 "ls" utility sorted very much as it would be in a dictionary. While telephone book ordering
 6284 would be an optimal goal for standard collation, this was ruled out as the order would be
 6285 language dependent. Furthermore, a requirement was that the order must be determined
 6286 solely from the text string and the collation rules; no external information (e.g., "pronu-
 6287 nciation dictionaries") could be required.

6288
 6289 As a result, the goal for the collation support is at level 3. This also matches the re-
 6290 quirements for the Canadian collation order standard, as well as other, known collation
 6291 requirements for alphabetic scripts. It specifically rules out collation based on pronun-
 6292 ciation rules, or based on semantic analysis of the text. The syntax for the LC_COLLATE
 6293 category source is the result of a cooperative effort between representatives for many
 6294 countries and organizations working with international issues, such as UniForum, The
 6295 Open Group, The Unicode Consortium Inc. and ISO, and it meets the requirements for
 6296 level 3, and has been verified to produce the correct result with examples based on
 6297 Canadian and Danish collation order.

6298
 6299 The directives that can be specified in an operand to the order_start keyword are based on
 6300 the requirements specified in several proposed standards and in customary use. The
 6301 following is a rephrasing of rules defined for "lexical ordering in English and French" by
 6302 the Canadian Standards Association (text in brackets is rephrased):

- 6303
 6304 (1) Once special characters (punctuation) have been removed from original strings,
 6305 the ordering is determined by scanning forward (left to right) [disregarding case
 6306 and diacriticals].
- 6307 (2) In case of equivalence, special characters are once again removed from original
 6308 strings and the ordering is determined scanning backward (starting from the
 6309 rightmost character of the string and back), character by character, (disregarding
 6310 case but considering diacriticals).
- 6311 (3) In case of repeated equivalence, special characters are removed again from
 6312 original strings and the ordering is determined scanning forward, character by
 6313 character, (considering both case and diacriticals).
- 6314 (4) If there is still an ordering equivalence after rules (1) through (3) have been
 6315 applied, then only special characters and the position they occupy in the string
 6316 are considered to determine ordering. The string that has a special character in
 6317 the lowest position comes first. If two strings have a special character in the
 6318 same position, the character [with the lowest collation value] comes first. In
 6319 case of equality, the other special characters are considered until there is a
 6320 difference or all special characters have been exhausted.

6321 It is estimated that the Technical Report covers the mechanisms to specify data to cover
6322 the requirements for all European languages, and Cyrillic and Middle Eastern scripts.
6323

6324 The Far East (particularly Japanese/Chinese) collations are often based on contextual
6325 information. In Japan, collations of strings containing CJK characters (ideograms) are
6326 often done considering some related information such as pronunciation, which needs a
6327 bulk dictionary (and some common sense). Such collation, in general, falls outside the
6328 desired goal of this Technical Report, and this Technical Report can support only a
6329 restricted of collations used in Japan. There are, however, several other collation rules
6330 (stroke/radical, or "most common pronunciation") which can be supported with the
6331 mechanism described here. Previous drafts contained a substitute statement, which
6332 performed a regular expression style replacement before string compares. It has been
6333 withdrawn based on balloter objections that it was not required for the types of ordering
6334 this Technical Report is aimed at.
6335

6336 The character (and collating element) order is defined by the order in which characters and
6337 elements are specified between the `order_start` and `order_end` keywords. This character
6338 order is used in range expressions in regular expressions. Weights assigned to the charac-
6339 ters and elements define the collation sequence; in the absence of weights, the character
6340 order is also the collation sequence.
6341

6342 The position keyword was introduced to provide the capability to consider, in a compare,
6343 the relative position of non-IGNORED characters. As an example, consider the two strings
6344 "o-ring" and "or-ing". Assuming the hyphen is IGNORED on the first pass, the two strings
6345 will compare equal, and the position of the hyphen is immaterial. On second pass, all
6346 characters except the hyphen are IGNORED, and in the normal case the two strings would
6347 again compare equal. By taking position into account, the first collates before the second.
6348

6349 This Technical Report adds a number of facilities over the ISO/IEC 9945:1993 POSIX
6350 standard, especially in the support for the ISO/IEC 10646 UCS character set. These
6351 extended facilities are in alignment with the ISO/IEC 14651 sorting standard. In addition
6352 to the facilities provided in ISO/IEC 14651, this specification contains mechanisms to put
6353 data into a FDCC-set environment, and has added facilities to sort sections differently, has
6354 facilities to reuse FDCC-sets in different notations via the "equivalence-symbol" keyword
6355 and tables.
6356

6357 **B.1.3.1 "reorder-after" rationale** 6358

6359 Much work has been done on FDCC-sets, making them quite general. The ISO/IEC 9945-
6360 2:1993 POSIX standard introduced a "copy" command for all categories of the POSIX
6361 locale. This is useful for many purposes and it ensures that two FDCC-sets are equivalent
6362 for this category. A further step in building on previous FDCC-set work is defined in this
6363 Technical Report.
6364

6365 Collating sequences often vary a bit from country to country, and from language to
6366 language, but generally much of the collating sequence is the same. For example the
6367 Danish sequence is for the most part the same as the German or English collation, but for
6368 about a dozen letters it differs. The same can be said for Swedish or Hungarian: generally
6369 the Latin collating sequence is the same, but a few characters are different.
6370

6371 This Technical Report defines a FDCC-set defined on the character repertoire of the
6372 ISO/IEC 10646 standard, in a character set independent way. The intention is that some of

6373 the information from this FDCC-set will be acceptable in many cultures, and that it can
 6374 serve as the basis for modifications in other cultures, to obtain a culturally acceptable
 6375 specification. Using the "reorder-after" construct will also help improve the overview of
 6376 what the changes really are for implementers and other users.
 6377

6378 An example of the use of the "reorder-after" construct is the following. A default
 6379 international ordering for the Latin alphabet may be adequate for Danish, with the
 6380 exception of the collation rules for the letters Û, ü, Æ, æ, Ä, ä, Ø, ø, Ö, ö, Å and å. By
 6381 applying the "reorder-after" construct, the Danish specification can be made more easily
 6382 by copying and reordering the existing international specification, rather than specifying
 6383 collation parameters for all Latin letters (with or without diacritics). There is no obligation
 6384 for Denmark to take this approach, but the "reorder-after" construct provides the
 6385 mechanism for doing so if it is deemed desirable.
 6386

6387 **B.1.3.2 awk script for "reorder-after" construct**

6388
 6389 A script has been written in the "awk" language defined in the POSIX standard ISO/IEC
 6390 9945-2 to implement the "reorder-after" construct. It functions as follows: It reads all of
 6391 the FDCC-set and if in the LC_COLLATE category, it processes the line, else it just
 6392 outputs the line. For the LC_COLLATE category it reads the lines and puts it into a
 6393 double linked list of strings identified by a line number; at the end of the LC_COLLATE
 6394 category all the lines are output. If the line is a "copy" keyword and it reads the file
 6395 referenced, extracting the LC_COLLATE section of the file in to the list of strings. If the
 6396 line is a "reorder-after" keyword, it sets a pointer to be the line number of the symbol to
 6397 of the "reorder-after" keyword. If the line is part of the "reorder-after" specification, it is
 6398 entered into the double linked list at this point, and the previous entry in the double linked
 6399 list for the <collation-element> is removed from the list. A "reorder-end" keyword
 6400 terminates the reordering.
 6401

```

6402 BEGIN { comment = "%"; back[0]= follow[0] = 0; }
6403 /LC_COLLATE/ { coll=1 }
6404 /END LC_COLLATE/ { coll=0; for (lnr= 1; lnr; lnr= follow[lnr]) print c-
6405 ont[lnr] }
6406
6407 { if (coll == 0) print $0 ;
6408   else { if ($1 == "copy") {
6409     file = $2
6410     while (getline < file )
6411       if ( $1 == "LC_COLLATE" ) copy_lc = 1
6412       else if ( $1 == "END" && $2 == "LC_COLLATE" ) copy_lc = 0
6413       else if (copy_lc) {
6414         lnr++
6415         follow[lnr-1] = lnr; back [ lnr ] = lnr-1
6416         cont[lnr] = $0; symb[ $1 ] = lnr
6417       }
6418     close (file )
6419   }
6420   else if ($1 == "reorder-after") { ra=1 ; after = symb [ $2 ] }
6421   else if ($1 == "reorder-end") ra = 0
6422   else {
6423     lnr++
6424     if (ra) follow [ lnr ] = follow [ after ]
6425     if (ra) back [ follow [ after ] ] = lnr
6426     follow[after] = lnr; back [ lnr ] = after
6427     cont[lnr] = $0
6428     if ( ra && $1 != comment && $1 != " ) {
6429       old = symb [ $1 ];
6430       follow [ back [ old ] ] = follow [ old ];
6431       back [ follow [ old ] ] = back [ old ];
6432       symb[ $1 ] = lnr;
6433     }

```

```

6434         after = lnr
6435     }
6436 }
6437 }
6438 B.1.3.3 Sample FDCC-set specification for Danish
6439
6440 escape_char /
6441 comment_char %
6442 repertoiremap "i18nrep"
6443 charset "ISO_8859-1:1987"
6444 % Distribution and use is free, also
6445 % for commercial purposes.
6446
6447 LC_VERSION
6448 title "Danish language FDCC-set for Denmark"
6449 source "Danish Standards Association"
6450 address "Kollegievej 6, DK-2920 Charlottenlund, Danmark"
6451 contact "Keld Simonsen"
6452 email "Keld.Simonsen@dkuug.dk"
6453 tel "+45 - 3996-6101"
6454 fax "+45 - 3996-6202"
6455 language "da"
6456 territory "DK"
6457 revision "4.2"
6458 date "1997-12-22"
6459
6460 category i18n:2000;LC_IDENTIFICATION
6461 category i18n:2000;LC_CTYPE
6462 category i18n:2000;LC_COLLATE
6463 category i18n:2000;LC_TIME
6464 category posix:1993;LC_NUMERIC
6465 category i18n:2000;LC_MONETARY
6466 category posix:1993;LC_MESSAGES
6467 category i18n:2000;LC_XLITERATE
6468 category i18n:2000;LC_NAME
6469 category i18n:2000;LC_ADDRESS
6470 category i18n:2000;LC_TELEPHONE
6471
6472 END LC_VERSION
6473
6474 LC_CTYPE
6475 copy "i18n"
6476 END LC_CTYPE
6477
6478 LC_COLLATE
6479 % The ordering algorithm is in accordance
6480 % with Danish Standard DS 377 (1980)
6481 % and the Danish Orthography Dictionary
6482 % (Retskrivningsordbogen, 2. udgave, 1996).
6483 % It is also in accordance with
6484 % Greenlandic orthography.
6485
6486 collating-element <A-A> from "<A><A>"
6487 collating-element <A-a> from "<A><a>"
6488 collating-element <a-A> from "<a><A>"
6489 collating-element <a-a> from "<a><a>"
6490 collating-symbol <SPECIAL>
6491 copy i18n
6492 reorder-after <CAPITAL>
6493 <CAPITAL>
6494 <CAPITAL-SMALL>
6495 <SMALL-CAPITAL>
6496 <SMALL>
6497 reorder-after <q8>
6498 <kk> <Q>;<SPECIAL>;<SMALL>;IGNORE
6499 reorder-after <t8>
6500 <TH> "<T><H>";"<TH><TH>";"<CAPITAL><CAPITAL>";IGNORE
6501 <th> "<T><H>";"<TH><TH>";"<SMALL><SMALL>";IGNORE
6502 reorder-after <y8>
6503 % <U:> and <U"> are treated as <Y> in Danish

```

```

6504 <U:> <Y>;<U:>;<CAPITAL>;IGNORE
6505 <u:> <Y>;<U:>;<SMALL>;IGNORE
6506 <U"&> <Y>;<U"&>;<CAPITAL>;IGNORE
6507 <u"&> <Y>;<U"&>;<SMALL>;IGNORE
6508 reorder-after <z8>
6509 % <AE> is a separate letter in Danish
6510 <AE> <AE>;<NONE>;<CAPITAL>;IGNORE
6511 <ae> <AE>;<NONE>;<SMALL>;IGNORE
6512 <AE'> <AE>;<ACUTE>;<CAPITAL>;IGNORE
6513 <ae'> <AE>;<ACUTE>;<SMALL>;IGNORE
6514 <A3> <AE>;<MACRON>;<CAPITAL>;IGNORE
6515 <a3> <AE>;<MACRON>;<SMALL>;IGNORE
6516 <A:> <AE>;<SPECIAL>;<CAPITAL>;IGNORE
6517 <a:> <AE>;<SPECIAL>;<SMALL>;IGNORE
6518 % <O//> is a separate letter in Danish
6519 <O//> <O//>;<NONE>;<CAPITAL>;IGNORE
6520 <o//> <O//>;<NONE>;<SMALL>;IGNORE
6521 <O//'> <O//>;<ACUTE>;<CAPITAL>;IGNORE
6522 <o//'> <O//>;<ACUTE>;<SMALL>;IGNORE
6523 <O:> <O//>;<DIAERESIS>;<CAPITAL>;IGNORE
6524 <o:> <O//>;<DIAERESIS>;<SMALL>;IGNORE
6525 <O"> <O//>;<DOUBLE-ACUTE>;<CAPITAL>;IGNORE
6526 <o"> <O//>;<DOUBLE-ACUTE>;<SMALL>;IGNORE
6527 % <AA> is a separate letter in Danish
6528 <AA> <AA>;<NONE>;<CAPITAL>;IGNORE
6529 <aa> <AA>;<NONE>;<SMALL>;IGNORE
6530 <A-A> <AA>;<A-A>;<CAPITAL>;IGNORE
6531 <A-a> <AA>;<A-A>;<CAPITAL-SMALL>;IGNORE
6532 <a-A> <AA>;<A-A>;<SMALL-CAPITAL>;IGNORE
6533 <a-a> <AA>;<A-A>;<SMALL>;IGNORE
6534 <AA'> <AA>;<AA'>;<CAPITAL>;IGNORE
6535 <aa'> <AA>;<AA'>;<SMALL>;IGNORE
6536 reorder-end
6537 END LC_COLLATE
6538
6539 LC_MONETARY
6540 int_curr_symbol " <D><K><K><SP>"
6541 currency_symbol " <k><r>"
6542 mon_decimal_point " <,>"
6543 mon_thousands_sep " <.>"
6544 mon_grouping 3;3
6545 positive_sign ""
6546 negative_sign " <->"
6547 int_frac_digits 2
6548 frac_digits 2
6549 p_cs_precedes 1
6550 p_sep_by_space 2
6551 n_cs_precedes 1
6552 n_sep_by_space 2
6553 p_sign_posn 4
6554 n_sign_posn 4
6555 END LC_MONETARY
6556
6557 LC_NUMERIC
6558 decimal_point " <,>"
6559 thousands_sep " <.>"
6560 grouping 3;3
6561 END LC_NUMERIC
6562
6563 LC_TIME
6564 abday " <m><a><n>" ; /
6565 " <t><i><r>" ; " <o><n><s>" ; /
6566 " <t><o><r>" ; " <f><r><e>" ; /
6567 " <l><o//><r>" ; " <s><o//><n>"
6568 day " <m><a><n><d><a><g>" ; /
6569 " <t><i><r><s><d><a><g>" ; /
6570 " <o><n><s><d><a><g>" ; /
6571 " <t><o><r><s><d><a><g>" ; /
6572 " <f><r><e><d><a><g>" ; /
6573 " <l><o//><r><d><a><g>" /

```

```

6574      " <s><o//><n><d><a><g>" ;
6575 week      7;19971201;4
6576 abmon     "<j><a><n>" ; "<f><e><b>" ; /
6577          "<m><a><r>" ; "<a><p><r>" ; /
6578          "<m><a><j>" ; "<j><u><n>" ; /
6579          "<j><u><l>" ; "<a><u><g>" ; /
6580          "<s><e><p>" ; "<o><k><t>" ; /
6581          "<n><o><v>" ; "<d><e><c>"
6582 mon      "<j><a><n><u><a><r>" ; /
6583          "<f><e><b><r><u><a><r>" ; /
6584          "<m><a><r><t><s>" ; /
6585          "<a><p><r><i><l>" ; /
6586          "<m><a><j>" ; /
6587          "<j><u><n><i>" ; /
6588          "<j><u><l><i>" ; /
6589          "<a><u><g><u><s><t>" ; /
6590          "<s><e><p><t><e><m><b><e><r>" ; /
6591          "<o><k><t><o><b><e><r>" ; /
6592          "<n><o><v><e><m><b><e><r>" ; /
6593          "<d><e><c><e><m><b><e><r>"
6594 d_t_fmt    "<%><a><SP><%><F><SP><%><T><SP><%><Z>"
6595 d_fmt      "<%><O><d><.><SP><%><B><SP><%><Y>"
6596 alt_digits "<0><.>;<1><.>;<2><.>;<3><.>;<4><.>; /
6597          <5><.>;<6><.>;<7><.>;<8><.>;<9><.>; /
6598          <1><0><.>;<1><1><.>;<1><2><.>;<1><3><.>;<1><4><.>; /
6599          <1><5><.>;<1><6><.>;<1><7><.>;<1><8><.>;<1><9><.>; /
6600          <2><0><.>;<2><1><.>;<2><2><.>;<2><3><.>;<2><4><.>; /
6601          <2><5><.>;<2><6><.>;<2><7><.>;<2><8><.>;<2><9><.>; /
6602          <3><0><.>;<3><1><.>"
6603 t_fmt      "<%><T>"
6604 am_pm     "" ; ""
6605 t_fmt_ampm ""
6606 timezone  "<C><E><T><-><1><C><E><T><SP><D><S><T><,><M><3><.><5><.><0> /
6607          <,><M><1><0><.><5><.><0>"
6608 END LC_TIME
6609
6610 LC_MESSAGES
6611 yesexpr   "<<(><1><J><j><Y><y><)/>><.><*>"
6612 noexpr    "<<(><0><N><n><)/>><.><*>"
6613 END LC_MESSAGES
6614
6615 LC_NAME
6616 name_fmt  "<%><p><%><t><%><g><%><t><%><m><%><t><%><f>"
6617 name_gen  ""
6618 name_mr   "<h><r>"
6619 name_mrs  "<f><r><u>"
6620 name_miss "<f><r><o/><k><e><n>"
6621 name_ms   "<f><r>"
6622 END LC_NAME
6623
6624 LC_ADDRESS
6625 country_name "<D><a><n><m><a><r><k>"
6626 country_post "<D><K>"
6627 lang_ab     "<d><a>"
6628 lang_term   "<d><a><n>"
6629 postal_fmt  "<%><a><%><N><%><f><%><N><%><d><%><N><%><b><%><N><%> /
6630          <%><s><SP><%><h><SP><%><e><SP><%><r><%><N> /
6631          <%><C><-><%><z><SP><%><T><%><N><%><C><%><N>"
6632 END LC_ADDRESS
6633
6634 LC_TELEPHONE
6635 tel_int_fmt "<+><%><c><SP><%><a><SP><%><1>"
6636 tel_dom_fmt "<%><1>"
6637 int_select "<0><0>"
6638 int_prefix "<4><5>"
6639 END LC_TELEPHONE
6640
6641
6642

```


6643 **B.1.4 LC_MONETARY Rationale.**

6644

6645 The currency symbol does not appear in LC_MONETARY because it is not defined in the
6646 C Standard's C locale. The C Standard limits the size of decimal points and thousands
6647 delimiters to single-byte values. In FDCC-sets based on multibyte coded character sets this
6648 cannot be enforced, obviously; this Technical Report does not prohibit such characters, but
6649 makes the behaviour unspecified (in the text "In contexts where other standards . . .").

6650

6651 The grouping specification is based on, but not identical to, the C Standard. The "-1"
6652 signals that no further grouping is performed, the equivalent of (CHAR_MAX) in the C
6653 Standard).

6654

6655 The FDCC-set definition is an extension of the C Standard localeconv() specification. In
6656 particular, rules on how currency_symbol is treated are extended to also cover int_
6657 curr_symbol, and p_set_by_space and n_sep_by_space have been augmented with the
6658 value 2, which places a space between the sign and the symbol (if they are adjacent;
6659 otherwise it should be treated as a 0). The following table shows the result of various
6660 combinations:

6661

6662

6663

6664

		p_sep_by_space		
		2	1	0
p_cs_precedes = 1	p_sign_posn = 0	(\$ 1.25)	(\$ 1.25)	(\$1.25)
	p_sign_posn = 1	+ \$1.25	+\$ 1.25	+\$1.25
	p_sign_posn = 2	\$1.25 +	\$ 1.25+	\$1.25+
	p_sign_posn = 3	+ \$1.25	+\$ 1.25	+\$1.25
	p_sign_posn = 4	\$ +1.25	+\$ 1.25	+\$1.25

6665

6666

6667

6668

6669

6670

p_cs_precedes = 0	p_sign_posn = 0	(1.25 \$)	(1.25 \$)	(1.25\$)
	p_sign_posn = 1	+1.25 \$	+1.25 \$	+1.25\$
	p_sign_posn = 2	1.25\$ +	1.25 \$+	1.25\$+
	p_sign_posn = 3	1.25+ \$	1.25 +\$	1.25+\$
	p_sign_posn = 4	1.25\$ +	1.25 \$+	1.25\$+

6671

6672

6673

6674

6675

6676

6677

6678 The following is an example of the interpretation of the mon_grouping keyword.

6679 Assuming that the value to be formatted is 123456789 and the mon_thousands_sep is "",

6680 then the following table shows the result. The third column shows the equivalent C

6681 Standard string that would be used to accommodate this grouping. It is the responsibility

6682 of the utility to perform mappings of the formats in this clause to those used by language

6683 bindings such as the C Standard.

6684

6685

6686

6687

6688

6689

6690

6691

6692

6693

6694

Mon_grouping	Formatted Value	C String
3;-1	123456'789	"\3\177"
3	123'456'789	"\3"
3;2;-1	1234'56'789	"\3\2\177"
3;2	12'34'56'789	"\3\2"
-1	123456789	"177"

In these examples, the octal value of (CHAR_MAX) is 177.

6695 The multiple currency support is specified such that a FDCC-set can be used without
 6696 change during the transition period in a static environment. For example in the case of the
 6697 Euro currency as being employed in a number of European countries, there is no need to
 6698 change the FDCC-set when shifting from one currency to two concurrent currencies; and
 6699 there is no need to change FDCC-set, when changing to the Euro as the only currency.
 6700 Also the same application call can be made to be valid for countries with a single
 6701 currency and countries with dual currencies. The specifications can also be used without
 6702 change of the FDCC-set on an installation, when converting from one national currency to
 6703 another, for example when removing some zeroes to form a new currency.

6704
 6705 The following example illustrates the support for multiple currencies; the example is for
 6706 the Euro in Germany:

```

6707
6708     LC_MONETARY
6709     valid_from           " ";           "19990101"
6710     valid_to            "20020630";     ""
6711     conversion_rate     1/1;           195/100
6712     int_curr_symbol     "<D><E><M><SP>";  "<E><U><R><SP>"
6713     currency_symbol     "<D><M>";      "<E><U><R>"
6714     mon_decimal_point   "<,>"
6715     mon_thousands_sep  "<.>"
6716     mon_grouping       3;3
6717     positive_sign       ""
6718     negative_sign      "<->"
6719     int_frac_digits     2;             2
6720     frac_digits        2;             2
6721     p_cs_precedes      1;             1
6722     p_sep_by_space     2;             2
6723     n_cs_precedes      1;             1
6724     n_sep_by_space     2;             2
6725     p_sign_posn        4;             4
6726     n_sign_posn        4;             4
6727
6728     END LC_MONETARY
  
```

6729 **B.1.5 LC_NUMERIC Rationale.**

6730 See the rationale for LC_MONETARY (B.1.3) for a description of the behaviour of
 6731 grouping.

6732 **B.1.6 LC_TIME Rationale.**

6733 The LC_TIME descriptions of abday, day, and abmon imply a Gregorian style calendar
 6734 (7-day weeks, 12-month years, leap years, etc.). Other calendars can be supported, for
 6735 example calendars with a fixed week length.

6736 In some FDCC-sets the field descriptors for weekday and month names will be given with
 6737 an initial small letter. Programs using these fields may need to adjust the capitalization if
 6738 the output is going to be used at the beginning of a sentence.

6739 The field descriptors corresponding to the optional keywords consist of a modifier
 6740 followed by a traditional field descriptor (for instance %Ex). If the optional keywords are
 6741 not supported by the application or are unspecified for the current FDCC-set, these field
 6742 descriptors are treated as the traditional field descriptor. For instance, assume the
 6743 following keywords:

```

6744     alt_digits "0th";"1st";"2nd";"3rd";"4th";"5th";"6th";"7th";"8th";"9th";"10th"
6745     d_fmt     "The %Od day of %B in %Y"
  
```

6753 On 1776-07-04, the %x field descriptor would result in "The 4th day of July in 1776,"
6754 while 1789-07-14 would come out as "The 14 day of July in 1789." It can be noted that
6755 the above example is for illustrative purposes only; the %o modifier is primarily intended
6756 to provide for Kanji or Hindi digits in date formats. While it is clear that an alternate year
6757 format is required, there is no consensus on the format or the requirements. As a result,
6758 while these keywords are reserved, the details are left unspecified. It is expected that
6759 National Standards Bodies will provide specifications.

6761 **B.1.7 LC_MESSAGES Rationale.**

6762 The LC_MESSAGES category is described in clause 4 as affecting the language used by
6763 utilities for their output. The mechanism used by the application to accomplish this, other
6764 than the responses shown here in the FDCC-set definition, is not specified by this version
6765 of this Technical Report. The ISO internationalization working group ISO/IEC
6766 JTC1/SC22/WG20 is developing an interface that would allow applications (and,
6767 presumably some of the standard utilities) to access messages from various message
6768 catalogs, tailored to a user's LC_MESSAGES value.

6771 **B.1.8 LC_XLITERATE Rationale.**

6772 Transliteration is often language dependent, transliterating one specific language to another
6773 specific language. For example transliteration from Russian to English, and from Serbian
6774 to German would normally be quite different, although the same repertoire of characters
6775 would be transliterated. Even transliteration of two languages using the same script into
6776 one language (for example from Russian to Danish and from Serbian to Danish), or
6777 transliteration of the same language (for example Russian into English or German) may be
6778 different. The language to be transliterated to is identified with the FDCC-set, which may
6779 also be used to identify a specific language to be transliterated from. Transliteration may
6780 also be to a specific repertoire of characters, determined for example by limitations of
6781 displaying equipment, or what the user can intelligibly read. The capabilities here allows
6782 for multiple fallback, so that the specification can be valid for all target character
6783 repertoires, eliminating the need for specific data for each target repertoire.

6786 **B.1.9 LC_NAME Rationale.**

6787 The LC_NAME category gives information to prepare a text for addressing a person, for
6788 example as a part of a postal address on an envelope, or as a saluting line in a letter.
6789 The information is intended to be given to an API that has the various naming information
6790 as parameters and yields a formatted string as the return value.

6791 The "profession" entry is intended for either the general profession of the person in
6792 question, or the job title, for use in letters or as part of the address on an envelope.

6796 **B.1.10 LC_ADDRESS Rationale.**

6797 The LC_ADDRESS category gives information to prepare a text for writing an address,
6798 for example as a part of a postal address on an envelope. The information is intended to
6799 be given to an API that has the various address information as parameters and yields a
6800 formatted string as the return value.

6801
6802
6803
6804

6805 B.1.11 LC_TELEPHONE Rationale.

6806
6807 The LC_TELEPHONE category gives information to prepare a text for writing a telephone
6808 number. The information is intended to be given to an API that has the various
6809 information on a telephone number as parameters and yields a formatted string as the
6810 return value. Both an international and a domestic formatting possibility is available.

6811 B.2 Character Set Rationale.

6812
6813 This Technical Report poses no requirement that multiple character sets or code sets be
6814 supported, leaving this as a marketing differentiation for implementors. Although multiple
6815 charmaps are supported, it is the responsibility of the application to provide the file(s); if
6816 only one is provided, only that one will be accessible.

6817
6818 The character set description text provides the capability to describe character set attributes
6819 (such as collation order or character classes) independent of character set encoding, and
6820 using only the characters in the portable character set. This makes it possible to create
6821 "generic" FDCC-set source texts for all code sets that share the portable character set
6822 (such as the ISO/IEC 8859 family or IBM Extended ASCII).

6823
6824 Applications are free to describe more than one code set in a character set description text.
6825 For example, if an application defines ISO/IEC 8859-1 as the primary code set, and
6826 ISO/IEC 8859-2 as an alternate set, with each character from the alternate code set
6827 preceded in data by a shift code, a character set description text could contain a complete
6828 description of the primary set and those characters from the secondary that are not
6829 identical, the encoding of the latter including the shift code.

6830
6831 Applications are free to choose their own symbolic names, as long as the names identified
6832 by this Technical Report are also defined; this provides support for already existing
6833 "character names".

6834
6835 The charmap was introduced to resolve problems with the portability of, especially,
6836 FDCC-set sources. While the portable character set (in Table 1) is a constant across all
6837 FDCC-sets for a particular application, this is not true for the extended character set.
6838 However, the particular coded character set used for an application does not necessarily
6839 imply different characteristics or collation: on the contrary, these attributes should in many
6840 cases be identical, regardless of codeset. The charmap provides the capability to define a
6841 common FDCC-set definition for multiple codesets (the same FDCC-set source can be
6842 used for codesets with different extended characters; the ability in the charmap to define
6843 "empty" names allows for characters missing in certain codesets).

6844
6845 In addition, some implementors have expressed an interest in using the charmap to define
6846 certain other characteristics of codesets, such as the <mb_cur_max> value for the
6847 particular codeset. (Note that <mb_cur_max> has to be equal to or lower than the C
6848 Standard {MB_LEN_MAX}, which is the application limit). Such extensions are not
6849 described here; but may be added in a later revision of this Technical Report.

6850
6851 The <escape_char> declaration was added at the request of the international community to
6852 ease the creation of portable charmaps on terminals not implementing the default
6853 backslash escape. (This approach was adopted because this is a new interface invented by
6854 ISO/IEC 9945-2:1993 POSIX. Historical interfaces, such as the shell command language
6855 and awk, have not been modified to accommodate this type of terminal.)
6856

6857 The octal number notation was selected to match those of POSIX "awk" and "tr" utilities
6858 and is consistent with that used by the POSIX localedef utility.
6859

6860 The charmap capability implements a facility available at some X/Open compatible
6861 applications. Its prime virtue is to support "generic" collation sequence source definitions.
6862 An implementor or an applications developer can produce a template definition that can be
6863 used to produce several codeset-dependent "compiled" FDCC-set definitions. The facility
6864 also removes any dependency in many source definitions on characters outside the
6865 character set defined in this clause.
6866

6867 The charmap allows specification of more than one encoding of a character. This allows
6868 for encodings that can encode items in more than one way. For example, an item can be
6869 encoded once as a fully composed character and again as a base character plus combining
6870 character. This would allow either representation to be recognized. As only the first
6871 occurrence of the character may be output, this technique could be used to normalize a
6872 character stream.
6873

6874 The ISO 2022 support introduced gives the possibility to refer other definitions via
6875 charmaps, so the full encoding does not have to be replicated. It supports shifting with G0,
6876 G1, G2 and G3 sets, and also general shifting of coded character sets via escape
6877 sequences.
6878

6879

6880 **B.3 Repertoiremap Rationale.**

6881
6882 The repertoiremap was introduced to make FDCC-sets independent of the availability of
6883 charmaps. With the repertoiremap it is possible to use a FDCC-set encoded with one set
6884 of symbolic character names, together with charmaps with other symbolic character
6885 naming schemes, provided there are repertoiremaps available for both naming schemes.
6886

6887 Repertoiremaps are also useful to describe repertoires of characters, to be used for
6888 example for transliteration.

Annex C (informative)

BNF Grammar

C.1 BNF Syntax Rules

The syntax used here is near to ISO/IEC 14977, but "_" is allowed in identifiers, and comma is not used as concatenator, as the items are just concatenated.

Definitions between <angle brackets> make use of terms not defined in this BNF syntax, and assume general English usage.

Other conventions:

* means 0 or more repetitions of a token.

+ means one or more repetitions of a token

Brackets [] indicate optional occurrence of a token.

Comments start with a % on a separate line.

There may be more specifications in the normative text that describes restrictions on the grammar.

C.2 Grammar for FDCC-sets

```
% The following is the overall FDCC-set grammar
FDCC_set_definition = [ global_statement* ] category+ ;
global_statement  = 'escape_char' SP char_symbol EOL
                  | 'comment_char' SP char_symbol end_of_line
                  | 'repertoiremap' SP quoted_string EOL
                  | 'charmap' SP quoted_string EOL ;
category          = lc_identification | lc_ctype | lc_collate
                  | lc_monetary | lc_numeric | lc_time
                  | lc_messages | lc_xliterate | lc_telephone
                  | lc_name | lc_address ;

% The following is the LC_IDENTIFICATION category grammar
lc_ident          = ident_head ident_keyword* ident_tail
                  | ident_head copy_FDCC_set ident_tail ;
ident_head        = 'LC_IDENTIFICATION' EOL ;
ident_keyword     = ident_keyword_string SP quoted_string EOL ;
ident_keyword_string = 'title' | 'source' | 'address' | 'contact'
                      | 'email' | 'tel' | 'fax' | 'language'
                      | 'territory' | 'audience' | 'application'
                      | 'abbreviation' | 'revision' | 'date' ;
ident_tail        = 'END' SP 'LC_IDENTIFICATION' EOL ;

% The following is the LC_CTYPE category grammar
lc_ctype          = ctype_head ctype_keyword* ctype_tail
                  | ctype_head copy_FDCC_set ctype_tail ;
ctype_head        = 'LC_CTYPE' EOL ;
ctype_keyword     = charclass_keyword SP charclass_list EOL
                  | charconv_keyword SP charconv_list EOL
                  | 'width' SP width_list EOL ;
charclass_keyword = 'upper' | 'lower' | 'alpha' | 'digit' |
                    | 'alnum' | 'punct' | 'xdigit' | 'space' |
                    | 'print' | 'graph' | 'blank' | 'cntrl' |
                    | 'outdigit'
                    | 'class' charclass_name semicolon ;
charclass_name    = '"combining"' | '"combining_level3"'
                  | ''' identifier ''' ;
```

```

6951 charclass_list = charclass_list semicolon char_symbol
6952 | charclass_list semicolon ctype_abs_ellipses
6953 semicolon char_symbol
6954 | charclass_list semicolon charsymbol
6955 ctype_symbolic_ellipses charsymbol
6956 | char_symbol ;
6957 width_list = charclass_list ':' number
6958 | width_list semicolon width_list ;
6959 charconv_keyword = 'toupper' | 'tolower'
6960 | 'map' '' identifier '' semicolon ;
6961 charconv_list = charconv_list semicolon charconv_entry
6962 | charconv_entry ;
6963 charconv_entry = '(' char_symbol comma char_symbol ')' ;
6964 ctype_symbolic_ellipses = '...' | '....' ;
6965 ctype_abs_ellipses = '...' ;
6966 ctype_tail = 'END' SP 'LC_TYPE' EOL ;
6967
6968 % The following is the LC_COLLATE category grammar
6969 lc_collate = collate_head collate_keywords collate_tail ;
6970 collate_head = 'LC_COLLATE' EOL ;
6971 collate_keywords = opt_statement* order_statements | delta ;
6972 opt_statement = 'collating-symbol' SP collsymbol_list EOL
6973 | 'collating-element' SP collelement SP 'from'
6974 SP collelem_string EOL
6975 | 'section-symbol' space+ section_symbol EOL
6976 | 'col_weight_max' SP number EOL
6977 | 'symbol-equivalence' SP collsymbol SP
6978 collsymbol EOL
6979 | collation_statement ;
6980 collelem_string = '' char_symbol+ '' ;
6981 order_statements = order_start collation_order order_end ;
6982 order_start = 'order_start' SP order_params EOL ;
6983 order_params = [section_symbol] [semicolon order_opts] ;
6984 order_opts = order_opt [ semicolon order_opt ]* ;
6985 order_opt = opt_word [ comma opt_word ]* ;
6986 opt_word = 'forward' | 'backward' | 'position' ;
6987 section = 'section' SP section_symbol [ SP
6988 collsymbol_list ] EOL ;
6989 collation_order = ( order_start | section |
6990 collation_statement)* ;
6991 collation_statement = collsymbol EOL
6992 | collating_element [ SP weight_list ] EOL ;
6993 collsymbol_list = collsymbol_element
6994 [ semicolon collsymbol_element ]* ;
6995 collsymbol_element = collsymbol
6996 | collsymbol SP ellipses SP collsymbol ;
6997 collating_element = char_symbol | collelement
6998 | ellipses | 'UNDEFINED' ;
6999 weight_list = weight_symbol [ semicolon weight_symbol ]* ;
7000 weight_symbol = <empty>
7001 | char_symbol
7002 | collsymbol
7003 | '' elem_list ''
7004 | '' symb_list '' | 'IGNORE' ;
7005 ellipses = '...' | '...' | '....' ;
7006 order_end = 'order_end' EOL ;
7007 delta = opt_statement*
7008 'copy' SP FDCC_set_name EOL
7009 opt_statement*
7010 reordering_statement* ;
7011 reordering_statement = reorder_after_block
7012 | reorder_section_after_1
7013 | reorder_section_block ;
7014 reorder_after_block = reorder_after (collation_order |
7015 reorder_after)* reorder_end ;
7016 reorder_after = 'reorder-after' SP collsymbol EOL ;
7017 reorder_end = 'reorder-end' EOL ;
7018 reorder_section_block = reorder_section_after_2 section_statement*
7019 reorder_section_end ;
7020 section_statement = section_symbol SP order_opts EOL ;

```

```

7021 reorder_section_after_1 = 'reorder-section-after' SP sectionsymbol SP
7022 collsymbol EOL;
7023 reorder_section_after_2 = 'reorder-section-after' SP collsymbol EOL;
7024 reorder_section_end = 'reorder-section-end' EOL ;
7025 collate_tail = 'END' SP 'LC_COLLATE' EOL
7026 % The following is the LC_MESSAGES category grammar
7027 lc_messages = messages_head messages_keyword* messages_tail
7028 | messages_head copy_FDCC_set messages_tail ;
7029 messages_head = 'LC_MESSAGES' EOL ;
7030 messages_keyword = 'yesexpr' SP '""' extended_reg_expr '""' EOL
7031 | 'yesexpr' SP '""' extended_reg_expr '""' EOL ;
7032 messages_tail = 'END' SP 'LC_MESSAGES' EOL ;
7033
7034
7035 % The following is the LC_MONETARY category grammar
7036 lc_monetary = monetary_head monetary_keyword* monetary_tail
7037 | monetary_head copy_FDCC_set monetary_tail ;
7038 monetary_head = 'LC_MONETARY' EOL ;
7039 monetary_keyword = mon_keyword_string SP quoted_string EOL
7040 | mon_keyword_strings SP mon_string_list EOL
7041 | mon_keyword_char SP mon_number_list EOL
7042 | mon_keyword_date SP mon_date_list EOL
7043 | 'conversion_rate' SP mon_conv_list EOL
7044 | 'mon_grouping' SP mon_group_list EOL ;
7045 mon_keyword_string = 'mon_decimal_point' | 'mon_thousands_sep'
7046 | 'positive_sign' | 'negative_sign' ;
7047 mon_keyword_strings = 'int_curr_symbol' | 'currency_symbol' ;
7048 mon_keyword_char = 'int_frac_digits' | 'frac_digits'
7049 | 'p_cs_precedes' | 'p_sep_by_space'
7050 | 'n_cs_precedes' | 'n_sep_by_space'
7051 | 'int_p_cs_precedes' | 'int_p_sep_by_space'
7052 | 'int_n_cs_precedes' | 'int_n_sep_by_space'
7053 | 'p_sign_posn' | 'n_sign_posn'
7054 | 'int_p_sign_posn' | 'int_n_sign_posn' ;
7055 mon_keyword_date = 'valid_from' | 'valid_to' ;
7056 mon_date_list = mon_date | mon_date_list semicolon mon_date ;
7057 mon_date = '""' 8 * digit '""' ;
7058 mon_group_list = number | mon_group_list semicolon number ;
7059 mon_string_list = quoted_string [ semicolon quoted_string]* ;
7060 mon_number_list = mon_number | mon_number_list semicolon
7061 mon_number ;
7062 mon_number = number | -1 ;
7063 mon_conv_list = mon_pair | mon_conv_list semicolon mon_pair ;
7064 mon_pair = number spaces* '/' spaces* number ;
7065 monetary_tail = 'END' SP 'LC_MONETARY' EOL ;
7066
7067 % The following is the LC_NUMERIC category grammar
7068 lc_numeric = numeric_head numeric_keyword* numeric_tail
7069 | numeric_head copy_FDCC_set numeric_tail ;
7070 numeric_head = 'LC_NUMERIC' EOL ;
7071 numeric_keyword = num_keyword_string SP quoted_string EOL
7072 | num_keyword_grouping SP num_group_list EOL ;
7073 num_keyword_string = 'decimal_point' | 'thousands_sep' ;
7074 num_keyword_grouping = 'grouping' ;
7075 num_group_list = number
7076 | num_group_list semicolon number ;
7077 numeric_tail = 'END' SP 'LC_NUMERIC' EOL ;
7078
7079 % The following is the LC_TIME category grammar
7080 lc_time = time_head time_keyword* time_tail
7081 | time_head copy_FDCC_set time_tail ;
7082 time_head = 'LC_TIME' EOL ;
7083 time_keyword = time_keyword_name SP time_list EOL
7084 | time_keyword_fmt SP quoted_string EOL
7085 | time_keyword_opt SP time_list EOL
7086 | 'week' SP number semicolon mon_date semicolon
7087 number EOL
7088 | time_keyword_num SP number EOL
7089 | 'timezone' SP time_list EOL;
7090 time_keyword_name = 'abday' | 'day' | 'abmon' | 'mon' | 'am_pm' ;

```



```

7091 time_keyword_fmt = 'd_fmt' | 'd_fmt' | 't_fmt' | 't_fmt_ampm';
7092 time_keyword_opt = 'era' | 'era_year' | 'era_d_fmt' | 'alt_digits'
7093 | era_d_t_fmt | era_t_fmt ;
7094 time_keyword_week = 'week' ;
7095 time_keyword_num = 'first_weekday' | 'first_workday'
7096 | 'cal_direction' ;
7097 time_list = time_list semicolon quoted_string
7098 | quoted_string ;
7099 time_tail = 'END' SP 'LC_TIME' EOL ;
7100
7101
7102
7103
7104
7105 % The following is the LC_XLITERATE category grammar
7106 lc_xliterate = translit_head [translit_include]
7107 [default_missing] translit_statement*
7108 translit_tail | translit_head copy_FDCC_set
7109 translit_tail ;
7110 translit_head = 'LC_XLITERATE' EOL ;
7111 translit_include = 'include' SP FDCC_set_name semicolon
7112 quoted_nonempty_string EOL ;
7113 default_missing = 'default_missing' SP quoted_string EOL ;
7114 translit_ignore = 'translit_ignore' SP charclass_list EOL ;
7115 translit_statement = char_or_string SP char_or_string [ semicolon
7116 char_or_string ]* EOL ;
7117 translit_tail = 'END' SP 'LC_XLITERATE' EOL ;
7118
7119 % The following is the LC_NAME category grammar
7120 lc_name = name_head name_keyword* name_tail
7121 | name_head copy_FDCC_set name_tail ;
7122 name_head = 'LC_NAME' EOL ;
7123 name_keyword = name_keyword_string SP quoted_string EOL ;
7124 name_keyword_string = 'name_fmt' | 'name_gen' | 'name_mr'
7125 | 'name_mrs' | 'name_ms' | 'name_miss'
7126 | 'name_ms' ;
7127 name_tail = 'END' SP 'LC_NAME' EOL ;
7128
7129 % The following is the LC_ADDRESS category grammar
7130 lc_address = address_head address_keyword* address_tail
7131 | address_head copy_FDCC_set address_tail ;
7132 address_head = 'LC_ADDRESS' EOL ;
7133 address_keyword = address_keyword_string SP quoted_string EOL ;
7134 address_keyword_string = 'postal_fmt' | 'country_name' |
7135 'country_post' | 'lang_name' | 'lang_ab2' |
7136 'lang_ab3_term' | 'lang_ab3_lib' ;
7137 address_tail = 'END' SP 'LC_ADDRESS' EOL ;
7138
7139 % The following is the LC_TELEPHONE category grammar
7140 lc_tel = tel_head tel_keyword* tel_tail
7141 | tel_head copy_FDCC_set tel_tail ;
7142 tel_head = 'LC_TELEPHONE' EOL ;
7143 tel_keyword = tel_keyword_string SP quoted_string EOL ;
7144 tel_keyword_string = 'tel_int_fmt' | 'tel_dom_fmt' | 'int_select'
7145 | 'int_prefix' ;
7146 tel_tail = 'END' SP 'LC_TELEPHONE' EOL ;
7147
7148 % The following grammar rules are common to all categories
7149 char = <any character except those that makes an End
7150 Of Line>
7151 graphic_char = <any char except control_chars and space> ;
7152 space = ' ' | <TAB> ;
7153 SP = space+ ;
7154 EOL = end_of_line | comment end_of_line ;
7155 end_of_line = <anything that makes an End Of Line (EOL) in
7156 the operating system employed> ;
7157 comment_char = <defined by the 'comment_char' keyword> ;
7158 escape_char = <defined by the 'escape_char' keyword> ;
7159 charsymbol = simple_symbol | ucs_symbol ;
7160 collsymbol = simple_symbol ;

```

```

7161 collelement = simple_symbol ;
7162 sectionsymbol = simple_symbol ;
7163 octdigit = '0'|'1'|'2'|'3'|'4'|'5'|'6'|'7' ;
7164 digit = '0'|'1'|'2'|'3'|'4'|'5'|'6'|'7'|'8'|'9' ;
7165 hex_upper = 'A'|'B'|'C'|'D'|'E'|'F' digit ;
7166 hexdigit = hex_upper | 'a'|'b'|'c'|'d'|'e'|'f' ;
7167 letter = 'a'|'b'|'c'|'d'|'e'|'f'|'g'|'h'|'i'|'j'|'k'|
7168 |'l'|'m'|'n'|'o'|'p'|'q'|'r'|'s'|
7169 |'t'|'u'|'v'|'w'|'x'|'y'|'z'|'A'|'B'|'C'|'D'|
7170 |'E'|'F'|'G'|'H'|'I'|'J'|'K'|'L'|'M'|'N'|'O'|
7171 |'P'|'Q'|'R'|'S'|'T'|'U'|'V'|'W'|'X'|'Y'|'Z' ;
7172 portable_graph_gtr = letter | digit | '!'|'"'|'#'|'$'|%'|'&' ;
7173 |'"'|'(')|')'|'*'|'+'|',''|'-'|'.'|'/'|':'|';' ;
7174 |'<'|'='|'?'|'@'|'['|'\'|']'|'^'|'_' ;
7175 |'`'|'{'|'|'}'|'~' ;
7176 portable_graph = portable_graph_gtr | '>' ;
7177 portable_char = portable_graph | '<NUL>' | '<ALERT>'
7178 | '<BACKSPACE>' | '<TAB>' | '<CARRIAGE_RETURN>'
7179 | '<NEWLINE>' | '<VERTICAL_TAB>' | '<FORM_FEED>' ;
7180 octal_char = escape_char octdigit octdigit octdigit* ;
7181 hex_char = escape_char 'x' hexdigit hexdigit hexdigit* ;
7182 decimal_char = escape_char 'd' digit digit digit* ;
7183 number = digit+ ;
7184 id_part = letter | digit | '-' | '_' ;
7185 four_digit_hex_string = hex_upper hex_upper hex_upper hex_upper ;
7186 identifier = letter id_part* ;
7187 simple_symbol = space* '<' portable_graph_gtr+ '>' ;
7188 ucs_symbol = space* '<U' four_digit_hex_string
7189 [ four_digit_hex_string ] '>' ;
7190 quoted_string = '"' char_symbol* '"' ;
7191 quoted_nonempty_string = '"' char_symbol+ '"' ;
7192 char_symbol = char | charsymbol
7193 | octal_char | hex_char | decimal_char ;
7194 elem_list = elem+ ;
7195 elem = char_symbol | collsymbol | collelement ;
7196 symb_list = collsymbol+ ;
7197 FDCC_set_name = FDCC-name | '"' FDCC-name '"' ;
7198 copy_FDCC_set = 'copy' FDCC_set_name EOL ;
7199 FDCC-name = portable_graph+ ;
7200 semicolon = space* ';' space* ;
7201 comma = space* ',' space* ;
7202 comment = comment_char char* ;

```

7203
7204
7205
7206
7207
7208
7209
7210
7211
7212
7213
7214
7215
7216
7217
7218
7219
7220
7221
7222
7223
7224
7225
7226
7227
7228
7229
7230
7231
7232
7233
7234
7235
7236
7237
7238
7239
7240
7241
7242
7243
7244
7245
7246
7247
7248
7249
7250
7251
7252
7253
7254

Annex D (informative)

Outstanding issues

This Technical Reports presents a trial for defining a general mechanism to specify cultural conventions. Though its contents are developed in order to form a standard, it has been decided that it will be a technical report in order to give information to public earlier.

The preparer of this report, ISO/IEC JTC1/SC22, expects the rapid progress of internationalization in the field of information technology will solve the issues mentioned below, and that this technical report will be used as a base for a new standard in the near future.

D.1 Comments from the Japanese member body

Japan considered this document should not be published as an international standard for the following reasons:

1) It is not clear whether the features which have their origin in ISO/IEC 9945-2 -- POSIX Part 2 - works well or not, after its separation from ISO/IEC 9945-2. Japan considers some mechanisms, e.g. "copy", will not work outside the POSIX environments.

2) It is not clear whether it makes sense or not to have a default value, which may be considered as a recommendation, for each cultural convention item. Japan is afraid that those default values are considered as Global Uniformity values - see ISO/IEC TR 11017:1998 for details.

3) It is not clear whether each specification form fits for world-wide cultural variations or not.

D.2 Comments from the U.S. member body

The U.S. National Body continues to be extremely disappointed with the contents of this Technical Report. Among the serious technical problems we see in this document are:

1. As an extension of the POSIX locale syntax (cf. ISO/IEC 9945-2), this document maintains the drawbacks of POSIX as a "specification method for cultural conventions" per se. In fact, it exacerbates the weaknesses of POSIX in this regard by conflating more, poorly justified LC_XXX formal definitions into a monolithic FDCC-set construct. This was clearly done with a particular implementation model in mind, but does not follow, nor even seem to be particularly informed by best current practice in the internationalization of software.

2. In an attempt to extend the POSIX LC_CTYPE specification to cover the repertoire of ISO/IEC 10646-1, this document blunders badly in asserting the cultural contextualization of character properties for the UCS. The treatment of LC_CTYPE as part of locales, i.e., as part of cultural adaptability, is an artifact of POSIX architecture and results from the need to have a place to put localized differences for case mapping. But by cloning other character properties having nothing to do with case mapping into LC_CTYPE, the net effect is to create a second source for specification of UCS character properties, with

7255 attendant dangers of divergence and errors, and with inevitable difficulties of maintenance
7256 and versioning. The clear intent is to influence other ISO standards to obtain their
7257 character property definitions from this document, instead of by reference to the widely
7258 implemented UCS property tables published by the Unicode
7259 Consortium. This will lead to confusion and interoperability problems for character
7260 properties. It has demonstrably already been a problem for the maintenance of the COBOL
7261 standard.

7262
7263 3. Each of the categories in the FDCC-set description has unaddressed problems and
7264 limitations. Rather than being resolved during the development of this document, many of
7265 these limitations were simply asserted to be "requirements". It appears to us that those are
7266 limitations of a particular envisioned implementation, engendered by legacy compatibility
7267 issues with POSIX, rather than requirements following from the legitimate needs for
7268 specification of cultural conventions. Because of this, implementers attempting to make
7269 use of the FDCC-set categories are immediately faced with an unexplained host of
7270 problems and mismatches to the actual cultural adaptability which they are trying to
7271 specify and implement to meet customer needs for information technology.

7272
7273 4. The repertoire map and LC_CTYPE sections deal with the repertoire of ISO/IEC 10646
7274 as it was in 1998, but nearly 55,000 more characters have been added to ISO/IEC
7275 10646-1:2000 and ISO/IEC 10646-2:2001. It would be a serious mistake for a technical
7276 report to be published in 2002 that uses an obsolete repertoire of characters.

7277
7278 Even for the characters which are in the repertoire, there are problems in the LC_CTYPE
7279 section. The classes to which characters are assigned - or in which they do not appear --
7280 often differ from comparable property lists in the Unicode Standard without any
7281 reasonable rationale being given. Since many implementations currently base their
7282 character properties on the data files in the Unicode Standard, arbitrary departure from
7283 those values is a recipe for interoperability problems. For example, the punct class
7284 includes many currency symbols, but for no apparent reason omits such currency symbols
7285 as the drachma, dong, and kip signs. The digit class includes a large group of digits
7286 from many cultures, but does not include Myanmar, Ethiopic, FullWidth, and others that
7287 are included in the comparable Unicode class.

7288
7289 Furthermore, the print and graph classes in LC_CTYPE do not include any Han
7290 ideographs, even though thousands of ideographs have been in ISO/IEC 10646-1 since
7291 1993. And the tolower/toupper classes do not include the fullwidth Latin character pairs,
7292 even though Japanese national standards do include such characters, and implementations
7293 must support case mappings of the fullwidth Latin letters.

7294
7295 5. The repertoire map itself is a completely unnecessary addition to this document. It is
7296 intended to document and promulgate a particularly bad collection of character mnemonic
7297 short strings. The U.S. views these "mnemonics" as confusing and irrelevant to the
7298 supposed scope of the TR. The need for short identifiers for characters can be met much
7299 better by the standard short UCS identifiers spelled out in ISO/IEC 10646, which *are* in
7300 widespread use.

7301
7302 6. The LC_MONETARY section attempts to add support for multiple currencies, but does
7303 so incorrectly. The idea was to cover the time period when many European countries
7304 would be using individual national currencies and also the euro. However, the definition
7305 allows users to create multiple names for currencies, implying that the names are
7306 synonyms of each other. This is incorrect. Deutschmarks and euros are not synonyms; they

7307 are two different currencies that could be used within one country at the same time.
7308 Similarly, French francs and euros also are not synonyms, but parts of LC_MONETARY
7309 are written as if two currencies like these are the same thing.
7310

7311 Besides the fact that the LC_MONETARY support for dual currencies is incorrect, it also
7312 is moot. By February 28, 2002, all 12 members of the European Union will have retired
7313 their national currencies and adopted the euro for all transactions. The functionality
7314 described in this technical report will be moot before the TR is even finalized.
7315

7316 7. The LC_TIME section includes some changes that are incompatible with POSIX.2.
7317 Some week definitions that have depended on Sunday being considered the first day of the
7318 week are changed in this TR to use Monday as the first day of the week. This would
7319 break existing implementations.
7320

7321 Also in the LC_TIME section, timezone information has been added. The U.S. National
7322 Body objects strongly to this because such information already is separately defined via
7323 the TIMEZONE environment variable and does not belong in a locale or FDCC-set. Many
7324 countries span multiple time zones, and including timezone information makes it
7325 impossible to write a locale or FDCC-set to support such countries.
7326

7327 8. The new LC_XLITERATE section for character transliteration is significantly
7328 incomplete. It also doesn't belong in a locale or FDCC-set anyway. Such functionality,
7329 where defined, should be similar to code set conversion - users should be able to pick any
7330 source and target, rather than having some limited set of transliterations hard-coded in an
7331 FDCC-set.
7332

7333 Even if one believes transliteration should be in an FDCC-set, the support in this TR is
7334 inadequate for international needs. The syntax provided here will not work for many Asian
7335 languages (and some others), and cannot be expanded in a compatible way in the
7336 future to support such languages. The limited string conversion functionality defined here
7337 is inadequate to the general problem of transliteration and is inappropriate for inclusion in
7338 this TR.

		Annex E	
		(informative)	
		Index	
7339			
7340			
7341			
7342			
7343			
7344	abbreviation	4.2	col_weight_max 4.4, 4.4.3
7345	abday	4.7	collating-element 4.4
7346	abmon	4.7	collating statements 4.4.1
7347	absolute ellipses	4.3	collating-symbol 4.4.6
7348	address	4.2	collating element 3.1.13
7349	addresses	4.11	collating sequence 3.1.15
7350	addset	5.1	collating-element 4.4.5
7351	alpha	4.3.1	collating-symbol 4.4
7352	alt_digits	4.7	collation 3.1.12
7353	am_pm	4.7	combining 4.3.1
7354	application	4.2	combining_level3 4.3.1
7355	audience	4.2	comment_char 4.1.4.1, 5.1
7356	blank	4.3.1	conformance 7
7357	block_separator	4.3.1	contact 4.2
7358	byte	3.1.1	continuation line 4.1.2
7359	cal_direction	4.7	control characters 4.3.1
7360	category	4.2	conversion_rate 4.5
7361	category names	4.1	copy 4.1.3, 4.2, 4.3.1, 4.4.2,
7362	category trailer	4.1	4.5, 4.6, 4.7, 4.8, 4.9, 4.10, 4.11, 4.12
7363	category header	4.1	country_ab2 4.11
7364	category body	4.1	country_ab3 4.11
7365	character	3.1.2	country_car 4.11
7366	character, graphic	4.3.1	country_isbn 4.11
7367	character, special	4.3.1	country_name 4.11
7368	character representation	4.1.1	country_num 4.11
7369	character, native digit	4.3.1	country_post 4.11
7370	character, hexadecimal digit	4.3.1	cultural convention 3.1.5
7371	character, multibyte	4.1.1	currency_symbol 4.5
7372	character, decimal constant	4.1.1	d_fmt 4.7
7373	character, hexadecimal constant	4.1.1	d_t_fmt 4.7
7374	character, space	4.3.1	date field descriptors 4.7.1
7375	character, octal constant	4.1.1	date 4.2
7376	character, control	4.3.1	day 4.7
7377	character, blank	4.3.1	decimal_point 4.6
7378	character, digit	4.3.1	default_missing 4.3.2
7379	character, punctuation	4.3.1	definitions 3.1
7380	character, printable	3.1.10	digit 4.3.1
7381	character class	3.1.9	ellipses 4.3, 4.4.1, 5.1
7382	character, coded	3.1.3	ellipses, absolute 4.3, 4.4.1
7383	Character set rationale	B.2	ellipses, symbolic 4.3, 4.4.1, 5.1
7384	charmap text	5.1	email 4.2
7385	charmap 5, 4.1.4.4, 3.1.7		equivalence class 3.1.16
7386	charmap rationale	B.2	era 4.7
7387	class	4.3.1	era_d_fmt 4.7
7388	cntrl	4.3.1	era_year 4.7
7389	code_set_name	5.1	escape_char 4.1.4.2, 5.1.6
7390	coded character	3.1.3	esqseq 5.1

7391	euro	B.1.3	LC_TIME rationale	B.1.6
7392	extended regular expression	4.8	LC_XLITERATE	4.9
7393	fax	4.2	LC_XLITERATE rationale	B.1.8
7394	FDCC-set, definition	4.1	LC_X_	4
7395	FDCC-set	4f	line continuation	4.1.4
7396	FDCC-set	3.1.6	lower	4.3.1
7397	FDCC-set rationale	B.1	map	4.3.1
7398	first_weekday	4.7	mb_cur_max	5.1
7399	first_workday	4.7	mb_cur_min	5.1
7400	frac_digits	4.5	messages	4.8
7401	graph	4.3.1	modified date field descriptors	4.7.2
7402	graphic characters	4.3.1	mon	4.7
7403	grouping	4.6	mon_decimal_point	4.5
7404	height	4.9	mon_grouping	4.5
7405	include	4.3.2	mon_thousands_sep	4.5
7406	include	5.1	monetary	4.5
7407	include	4.3.2.2	multicharacter collating element	3.1.14
7408	int_curr_symbol	4.5	n_cs_precedes	4.5
7409	int_frac_digits	4.5	n_sep_by_space	4.5
7410	int_n_cs_precedes	4.5	n_sign_posn	4.5
7411	int_n_sep_by_space	4.5	name formatting	4.10
7412	int_n_sign_posn	4.5	name_fmt	4.10
7413	int_p_cs_precedes	4.5	name_gen	4.10
7414	int_p_sep_by_space	4.5	name_miss	4.10
7415	int_p_sign_posn	4.5	name_mr	4.10
7416	int_prefix	4.12	name_mrs	4.10
7417	int_select	4.12	name_ms	4.10
7418	keywords	4.1	negative_sign	4.5
7419	lang_ab	4.11	noexpr	4.8
7420	lang_lib	4.11	notations	3.2
7421	lang_name	4.11	numeric	4.6
7422	lang_term	4.11	operands	4.1
7423	language	4.2	order_end	4.4.9, 4.4
7424	LC_ADDRESS	4.11	order_start	4.4, 4.4.8
7425	LC_ADDRESS rationale	B.1.10	outdigit	4.3.1
7426	LC_COLLATE	4.4	p_cs_precedes	4.5
7427	LC_COLLATE rationale	B.1.3	p_sep_by_space	4.5
7428	LC_CTYPE	4.3	p_sign_posn	4.5
7429	LC_CTYPE rationale	B.1.2	portable character set	3.2.4
7430	LC_IDENTIFICATION	4.2	positive_sign	4.5
7431	LC_IDENTIFICATION rationale	B.1.1	POSIX	1
7432	LC_MESSAGES	4.8	POSIX differences	A
7433	LC_MESSAGES rationale	B.1.7	POSIX conformance	4.2
7434	LC_MONETARY	4.5	postal addresses	4.11
7435	LC_MONETARY rationale	B.1.4	postal_fmt	4.11
7436	LC_NAME	4.10	pre-category statements	4.1.4
7437	LC_NAME rationale	B.1.9	print	4.3.1
7438	LC_NUMERIC	4.6	printable character	3.1.10
7439	LC_NUMERIC rationale	B.1.5	punct	4.3.1
7440	LC_TELEPHONE	4.12	punctuation characters	4.3.1
7441	LC_TELEPHONE rationale	B.1.11	redefine	4.3.2
7442	LC_TIME	4.7	references	2

7443	reorder-after	4.4, 4.4.10
7444	reorder-after rationale	B.1.2.1
7445	reorder-end	4.4, 4.4.11
7446	reorder-section-after	4.4, 4.4.13
7447	reorder-section-end	4.4, 4.4.14
7448	repertoire rationale	B.3
7449	repertoire	6
7450	repertoiremap	6, 3.1.8, 5.1, 4.1.4.3
7451	revision	4.2
7452	scope	1
7453	section	4.4, 4.4.4, 4.4.12
7454	source	4.2
7455	space	4.3.1
7456	special characters	4.3.1
7457	symbol-equivalence	4.4, 4.4.7
7458	symbolic ellipses	4.3, 5.1
7459	symbolic name	4.1.1
7460	syntax format	3.2.1
7461	t_fmt	4.7
7462	t_fmt_ampm	4.7
7463	tel	4.2
7464	tel_dom_fmt	4.12
7465	tel_int_fmt	4.12
7466	telephone numbers	4.12
7467	territory	4.2
7468	text file	3.1.4
7469	thousands_sep	4.6
7470	timezone	4.7
7471	title	4.2
7472	tolower	4.3.1
7473	toupper	4.3.1
7474	translit_ignore	4.9
7475	transliteration	4.9
7476	transliteration statements	4.9.1
7477	upper	4.3.1
7478	valid_from	4.5
7479	valid_to	4.5
7480	visible glyph portable characters	3.2.4
7481	week	4.7
7482	white space	3.1.11
7483	width	4.9
7484	xdigit	4.3.1
7485	yesexpr	4.8
7486		
7487		

BIBLIOGRAPHY

- 7488
7489
7490 The following specifications are considered relevant to this Technical Report, in addition
7491 to the normative references.
7492
7493 CEPT, CEPT-MAILCODE, *Country code for mail*.
7494
7495 ISO 646, *Information technology - ISO 7-bit coded character set for information inter-*
7496 *change*.
7497
7498 ISO/IEC 9899, *Information technology - Programming language C*.
7499
7500 ISO/IEC 14977, *Information technology - Syntactic metalanguage - Extended BNF*.
7501
7502 The Unicode Consortium: *The Unicode Standard, Version 2.0*, Addison Wesley
7503 Developers Press, July 1996. ISBN 0-201-48345-9.
7504
7505 IBM: *National Language Design Guide Volume 2 - National Language Support Reference*
7506 *Manual*, IBM SE09-8002-03, August 1994.
7507
7508 STRÍ: *Nordic Cultural Requirements on Information Technology (Summary report)*, STRÍ
7509 TS3, Libris, Reykjavík, Iceland 1992. ISBN 9979-9004-3-1.
7510
7511