

Reference number of working document: **ISO/IEC JTC1/SC22/WG20 N822**

Date: 2001-03-22

Reference number of document: **ISO/IEC DTR 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*

1

Document type: International standard  
Document subtype: if applicable  
Document stage: (40) Enquiry  
Document language: E

		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	36
13	4.6 LC_NUMERIC	40
14	4.7 LC_TIME	41
15	4.8 LC_MESSAGES	48
16	4.9 LC_XLITERATE	48
17	4.10 LC_NAME	50
18	4.11 LC_ADDRESS	51
19	4.12 LC_TELEPHONE	53
20	5 CHARMAP	53
21	6 REPERTOIREMAP	58
22	7 CONFORMANCE	85
23		
24	Annex A (informative) DIFFERENCES FROM POSIX	86
25	Annex B (informative) RATIONALE	88
26	Annex C (informative) BNF GRAMMAR	102
27	Annex D (informative) ISSUES LIST	107
28	Annex E (informative) INDEX	109
29	BIBLIOGRAPHY	112

## 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

# Information technology — Specification method for cultural conventions

## 1 SCOPE

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

The specification is upward compatible with POSIX locale specifications - a locale conformant to POSIX specifications will also be conformant to the specifications in this Technical Report, while the reverse condition will not hold. The descriptions are intended to be coded in text files to be used via Application Programming Interfaces, that are expected to be developed for a number of programming languages.

## 2 NORMATIVE REFERENCES

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

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

ISO/IEC 2022, *Information technology - Character code structure and extension techniques*.

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

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

ISO 8601, *Data elements and interchange formats - Information interchange - Representation of dates and times*.

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

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

ISO/IEC 14651:2000, *Information technology - International string ordering - Method for comparing character strings and description of a default tailorable ordering*.

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

### 3 TERMS, DEFINITIONS AND NOTATIONS

#### 3.1 Terms and definitions

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

##### 3.1.1 Bytes and characters

###### 3.1.1.1

**byte:**

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

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

###### 3.1.1.2

**character:**

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

###### 3.1.1.3

**coded character:**

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

###### 3.1.1.4

**text file:**

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

##### 3.1.2 cultural and other major concepts

###### 3.1.2.1

**cultural convention:**

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

###### 3.1.2.2

**FDCC**

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

###### 3.1.2.3

**FDCC-set:**

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

###### 3.1.2.4

**charmap:**

A definition of a mapping between symbolic character names and character codes, plus related information.

239  
240  
241  
242  
243  
244  
245  
246  
247  
248  
249  
250  
251  
252  
253  
254  
255  
256  
257  
258  
259  
260  
261  
262  
263  
264  
265  
266  
267  
268  
269  
270  
271  
272  
273  
274  
275  
276  
277  
278  
279  
280  
281  
282  
283  
284  
285

### **3.1.2.5**

#### **repertoiremap:**

A definition of a mapping between symbolic character names and characters for the repertoire of characters used in a FDCC-set, further described in clause 6.

### **3.1.3 FDCC categories related**

#### **3.1.3.1**

##### **character class:**

A named set of characters sharing an attribute associated with the name of the class.

#### **3.1.3.2**

##### **collation:**

The logical ordering of strings according to defined precedence rules.

#### **3.1.3.3**

##### **collating element:**

The smallest entity used to determine logical ordering.

See collating sequence. A collating element consists of either a single character, or two or more characters collating as a single entity. The LC\_COLLATE category in the associated FDCC-set determines the set of collating elements.

#### **3.1.3.4**

##### **multicharacter collating element:**

A sequence of two or more characters that collate as an entity.

For example, in some languages two characters are sorted as one letter, as in the case for Danish and Norwegian "aa".

#### **3.1.3.5**

##### **collating sequence:**

The relative order of collating elements as determined by the setting of the LC\_COLLATE category in the applied FDCC-set.

#### **3.1.3.6**

##### **equivalence class:**

A set of collating elements with the same primary collation weight.

Elements in an equivalence class are typically elements that naturally group together, such as all accented letters based on the same letter.

The collation order of elements within an equivalence class is determined by the weights assigned on any subsequent levels after the primary weight.



## 3.2 Notations

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

### 3.2.1 Notation for defining syntax

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

The syntax notation looks as follows:

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

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

```
%s    specifies a string
%d    specifies a decimal integer
%c    specifies a character
%o    specifies an octal integer
%x    specifies a hexadecimal integer
```

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

All other characters in the format string except

```
%%    specifies a single %
\n    specifies an end-of-line
```

represent themselves.

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

### 3.2.3 Portable character set

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

**Table 1: Portable character set**

Symbolic name	Glyph	UCS	Description
<NUL>		<U0000>	NULL (NUL)
<alert>		<U0007>	BELL (BEL)
<backspace>		<U0008>	BACKSPACE (BS)
<tab>		<U0009>	CHARACTER TABULATION (HT)

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

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

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

#### 4 FDCC-set

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

460	LC_IDENTIFICATION	Versions and status of categories
461	LC_CTYPE	Character classification, case conversion and code transformation.
462		
463	LC_COLLATE	Collation order.
464	LC_TIME	Date and time formats.
465	LC_NUMERIC	Numeric, non-monetary formatting.
466	LC_MONETARY	Monetary formatting.
467	LC_MESSAGES	Formats of informative and diagnostic messages and interactive responses.
468		
469	LC_XLITERATE	Character transliteration.
470	LC_NAME	Format of writing personal names.
471	LC_ADDRESS	Format of postal addresses.

472 LC\_TELEPHONE Format for telephone numbers, and other telephone  
473 information.  
474

475 Note: In future editions of this Technical Report further categories may be added.  
476

477 Other category names beginning with the 3 characters "LC\_" are reserved for future  
478 standardization, except for category names beginning with the five characters "LC\_X\_"  
479 which is not used for future addition of categories specified in this Technical Report. An  
480 application may thus use category names beginning with the five characters "LC\_X\_" for  
481 application defined categories to avoid clashes with future standardized categories.  
482

483 This Technical Report also defines an FDCC-set named "i18n" with values for some of  
484 the above categories in order to simplify FDCC-set descriptions for a number of cultures.  
485 The contents of "i18n" categories should not necessarily be considered as the most  
486 commonly accepted values, while in many cases it could be the recommended values.  
487

#### 488 4.1 FDCC-set description 489

490 FDCC-sets are described with the syntax presented in this subclause. For the purposes of  
491 this Technical Report, the text is referred to as the FDCC-set definition text or FDCC-set  
492 source text.  
493

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

503 There are no **naming conventions** for FDCC-sets specified in this Technical Report, but  
504 clause 6.8 in ISO/IEC 15897:1999 specifies naming rules for POSIX locales, charmaps  
505 and repertoire maps, that may also be applied to FDCC-sets, charmaps and repertoire maps  
506 specified according to this Technical Report.  
507

508 A **category source definition** consists of a category header, a category body, and a  
509 category trailer. A category header consists of the character string naming of the category,  
510 beginning with the characters "LC\_". The category trailer consists of the string "END",  
511 followed by one or more "blank"s and the string used in the corresponding category  
512 header.  
513

514 The **category body** consists of one or more lines of text. Each line is one of the  
515 following:  
516

- 517 - a line containing an identifier, optionally followed by one or more operands. Identifiers  
518 are either keywords, identifying a particular FDCC, or collating elements, or section  
519 symbols,
- 520 - one of transliteration statements defined in 4.3.  
521

522 In addition to the keywords defined in this Technical Report, the source can contain  
523 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

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

Example: c ä "May"

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

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

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

Example: \x63;\xe7;

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

Example: \d99; \d231;

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

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

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

Note: The <character> symbolic character notation is recommended for use of specifying all characters in a FDCC-set, to facilitate portability of the FDCC-sets, as the coded character set of the application of the FDCC-set may be different from the coded character set of the FDCC-set source. This is also recommended for format effectors in strings, such as in LC\_DATE or LC\_ADDRESS, where the format effectors are allowed to be stored together with the rest of the string, in a binary string with a different encoding from that of the source FDCC-set.

#### 4.1.2 Continuation of lines

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

### 629 4.1.3 Names for copy keyword

630

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

633

- 634 - "i18n" which indicate the "i18n" FDCC-set defined in this specification,
- 635 - the name of a FDCC-set or POSIX locale registered by the process defined in ISO/IEC  
636 15897,
- 637 - any other name which may be recognized in some local context - not being  
638 recommended as an international specification.

639

### 640 4.1.4 Pre-category statements

641

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

644

#### 645 4.1.4.1 comment\_char

646

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

649

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

651

652 The comment character defaults to the number-sign (#). All examples in this Technical  
653 Report use "%" as the <comment\_character>, except where otherwise noted. Blank lines  
654 and lines containing the <comment\_character> in the first position are ignored. In collating  
655 statements a <comment\_character> occurring where the delimiter ";" may occur,  
656 terminates the collating statement.

657

#### 658 4.1.4.2 escape\_char

659

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

662

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

664

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

668

#### 669 4.1.4.3 repertoiremap

670

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

674

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

675

676 The name is one of:

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

681 **4.1.4.4 charmap**

682

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

685

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

687

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

692

693 The name is one of:

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

697

698 **4.2 LC\_IDENTIFICATION**

699

700 The LC\_IDENTIFICATION category defines properties of the FDCC-set, and which  
701 specification methods the FDCC-set is conforming to. All keywords are mandatory unless  
702 otherwise noted, and the operands are strings. The following keywords are defined:

703

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

727

728 If information required for any of the mandatory keywords above is not available, then the  
729 corresponding string is an empty string. If required information is not present in ISO 639  
730 or ISO 3166, the relevant Maintenance Authority should be approached to get the needed  
731 item registered.

732



733 Note: Only one language per territory can be addressed with a single FDCC-set; an  
 734 additional FDCC-set is required for each additional language for that territory.  
 735

736 **category** Is used to define that a category is present and what  
 737 specification the category is claiming conformance to. The  
 738 first operand is a string in double-quotes that describes the  
 739 specification that the category is claiming conformance to,  
 740 and the following values are defined:  
 741 "i18n:2001"  
 742 "posix:1993"  
 743 The second operand is a string with the category name,  
 744 where the category names of clause 4 are defined. More than  
 745 one "category" keyword may be given, but only one per  
 746 category name.  
 747

748 The "i18n" LC\_IDENTIFICATION category is:  
 749

```

750 LC_IDENTIFICATION
751 % This is the ISO/IEC TR 14652 "i18n" definition for
752 % the LC_IDENTIFICATION category.
753 %
754 title "ISO/IEC TR 14652 i18n FDCC-set"
755 source "ISO/IEC Copyright Office"
756 address "Case postale 56, CH-1211 Geneve 20, Switzerland"
757 contact ""
758 email ""
759 tel ""
760 fax ""
761 language ""
762 territory ""
763 revision "1.0"
764 date "2001-03-22"
765 %
766 category "i18n:2001";LC_IDENTIFICATION
767 category "i18n:2001";LC_CTYPE
768 category "i18n:2001";LC_COLLATE
769 category "i18n:2001";LC_TIME
770 category "i18n:2001";LC_NUMERIC
771 category "i18n:2001";LC_MONETARY
772 category "i18n:2001";LC_MESSAGES
773 category "i18n:2001";LC_NAME
774 category "i18n:2001";LC_ADDRESS
775 category "i18n:2001";LC_TELEPHONE
776
777 END LC_IDENTIFICATION
  
```

778

779

### 780 4.3 LC\_CTYPE

781

782 The LC\_CTYPE category defines character classification, case conversion, character  
 783 transformation, and other character attribute mappings. Support for the portable character  
 784 set is required.  
 785

786 A series of characters in a specification can be represented by the hexadecimal symbolic  
 787 ellipsis symbol ".." (two dots), the decimal symbolic ellipses symbols "...." (4 dots), the  
 788 double increment hexadecimal symbolic ellipses "..(2)..", or the absolute ellipses "... (3  
 789 dots).  
 790

791 The **hexadecimal symbolic ellipsis** ("..") specification is only valid between symbolic  
 792 character names. The symbolic names consists of zero or more nonnumeric characters

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

804  
 805 The **decimal symbolic ellipsis** ("...") specification is only valid between symbolic  
 806 character names. The symbolic names consist of zero or more nonnumeric characters from  
 807 the set shown with visible glyphs in Table 1, followed by an integer formed by one or  
 808 more decimal digits. The characters preceding the decimal integer are identical in the two  
 809 symbolic names, and the integer formed by the decimal digits in the second symbolic  
 810 name is identical to or greater than the integer formed by the decimal digits in the first  
 811 name. This is interpreted as a series of symbolic names formed from the common part and  
 812 each of the integers in decimal format between the first and the second integer, inclusive,  
 813 and with a length of the symbolic names generated that is equal to the length of the first  
 814 (and also the second) symbolic name. As an example, <j0101>....<j0104> is interpreted as  
 815 the symbolic names <j0101>, <j0102>, <j0103>, and <j0104>, in that order.

816  
 817 The **double increment hexadecimal symbolic ellipses** ("..(2)..") works like the  
 818 hexadecimal symbolic ellipses, but generates only every other of the symbolic character  
 819 names. As an example. <U01AC>..(2)..<U01B2> is interpreted as the symbolic character  
 820 names <U01AC>, <U01AE>, <U01B0>, and <U01B2>, in that order.

821  
 822 The **absolute ellipsis** specification is only valid within a single encoded character set. An  
 823 ellipsis is interpreted as including in the list all characters with an encoded value higher  
 824 than the encoded value of the character preceding the ellipsis and lower than the encoded  
 825 value of the character following the ellipsis. The absolute ellipsis specification is  
 826 deprecated, as this is only relevant to FDCC-sets not using symbolic characters.  
 827 As an example, \x30;...;\x39 includes in the character class all characters with encoded  
 828 values between the endpoints.

### 829 4.3.1 Character classification keywords

830  
 831 The following keywords are recognized. In the descriptions, the term "automatically  
 832 included" means that it is not an error to either include the referenced characters or to  
 833 omit them; the interpreting system provides them if missing and accept them silently if  
 834 present.

835  
 836  
 837 **copy** Specify the name of an existing FDCC-set to be used as the source for the  
 838 definition of this category. If this keyword is specified, no other keyword is  
 839 specified.  
 840 **upper** Define characters to be classified as uppercase letters. No character  
 841 specified for the keywords "cntrl", "digit", "punct", or "space" is specified.  
 842 The uppercase letters A through Z of the portable character set,  
 843 automatically belong to this class, with application-defined character values.  
 844 The keyword may be omitted.

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

897		"xdigit", or as the <space> character is specified. The keyword is specified.
898	<b>xdigit</b>	Define the characters to be classified as hexadecimal digits. Only the
899		characters defined for the class "digit" are specified, in ascending sequence
900		by numerical value, followed by sets of six characters representing the
901		hexadecimal digits 10 through 15 in ascending order (for example <A>,
902		<B>, <C>, <D>, <E>, <F>, <a>, <b>, <c>, <d>, <e>, <f>). If this keyword
903		is not specified, the digits <0> through <9>, the uppercase letters "A"
904		through <F>, and the lowercase letters <a> through <f>, automatically
905		belong to this class, with application-defined character values.
906	<b>graph</b>	Define characters to be classified as printable characters, not including the <space>
907		character. If this keyword is not specified, characters specified for the keywords
908		"upper", "lower", "alpha", "digit", "xdigit", and "punct" belong to this character
909		class. No character specified for the keyword "cntrl" is specified.
910	<b>print</b>	Define characters to be classified as printable characters, including the
911		<space> character. If this keyword is not provided, characters specified for
912		the keywords upper, lower, alpha, digit, xdigit, punct, graph, and the
913		<space> character belong to this character class. No character specified for
914		the keyword "cntrl" is specified.
915	<b>toupper</b>	Define the mapping of lowercase letters to uppercase letters. The operand
916		consists of character pairs, separated by semicolons. The characters in each
917		character pair are separated by a comma and the pair enclosed by paren-
918		theses. The first character in each pair is the lowercase letter, the second the
919		corresponding uppercase letter. Only characters specified for the keywords
920		"lower" and "upper" are specified. If this keyword is not specified, the
921		lowercase letters <a> through <z>, and their corresponding uppercase letters
922		<A> through <Z>, are automatically included, with application-defined
923		character values.
924	<b>tolower</b>	Define the mapping of uppercase letters to lowercase letters. The operand
925		consists of character pairs, separated by semicolons. The characters in each
926		character pair are separated by a comma and the pair enclosed by
927		parentheses. The first character in each pair is the uppercase letter, the
928		second the corresponding lowercase letter. Only characters specified for the
929		keywords "lower" and "upper" are specified. If this keyword is specified,
930		the uppercase letters <A> through <Z>, and their corresponding lowercase
931		letter, are specified. If this keyword is not specified, the mapping is the
932		reverse mapping of the one specified for toupper.
933	<b>class</b>	Define characters to be classified in the class with the name given in the
934		first operand, which is a string. This string only contains characters of the
935		portable character set that either has the string "LETTER" in its description,
936		or is a digit or <hyphen-minus> or <low-line>. The following operands are
937		characters. This keyword is optional. The keyword can only be specified
938		once per named class. The following two names are recognized:
939	<b>combining</b>	Characters to form composite graphic symbols, such
940		as characters listed in ISO/IEC 10646:1993 annex B.1.
941	<b>combining_level3</b>	Characters to form composite graphic symbols, that
942		may also be represented by other characters, such as
943		characters listed in ISO/IEC 10646-1:1993 annex B.2.
944		The class names "upper", "lower", "alpha", "digit", "space", "cntrl", "punct",
945		"graph", "print", "xdigit", and "blank" are taken to mean the classes defined
946		by the respective keywords.
947	<b>width</b>	Define the column width of characters, for example for use of the C
948		function wwidth(). The operands are first a list for characters, possibly

949 using various ellipses, and semicolon separated, then a <colon>, and then  
 950 the width of these characters given as an unsigned positive integer. Such  
 951 width-lists separated by <semicolon> may be given for the various widths.  
 952 The default value of width of characters in class "cntrl" and class  
 953 "combining" is 0, else the default value of width is 1. A width for a  
 954 character may be overridden by a WIDTH specification in a charmap. This  
 955 keyword is optional.

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

964  
 965 The mapping names "toupper", and "tolower" are taken to mean the  
 966 mapping defined by the respective keywords.

967  
 968 Example of use of the "map" keyword:

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

970  
 971 This example introduces a new mapping "kana" that maps three Katakana characters to corresponding Hiragana  
 972 characters.  
 973

974  
 975 Table 2 shows the allowed character class combinations.

976  
 977 **Table 2: Valid Character Class Combinations**

978	Class	upper	lower	alpha	digit	space	cntrl	punct	graph	print	xdigit	blank
979	upper		+	A	x	x	x	x	A	A	+	x
980	lower	+		A	x	x	x	x	A	A	+	x
981	alpha	+	+		x	x	x	x	A	A	+	x
982	digit	x	x	x		x	x	x	A	A	A	x
983	space	x	x	x	x		+	*	*	*	x	+
984	cntrl	x	x	x	x	+		x	x	x	x	+
985	punct	x	x	x	x	+	x		A	A	x	+
986	graph	+	+	+	+	+	x	+		A	+	+
987	print	+	+	+	+	+	x	+	+		+	+
988	xdigit	+	+	+	+	x	x	x	A	A		x
989	blank	x	x	x	x	A	+	*	*	*	x	

990  
 991 Note 1: Explanation of codes:

992 A Automatically included; see text

993 + Permitted

994 x Mutually exclusive

995 \* See note 2

996  
 997 Note 2: The <space> character, which is part of the "space" and "blank" class, cannot belong to "punct" or  
 998 "graph", but automatically belong to the "print" class. Other "space" or "blank" characters can be classified  
 999 as "punct", "graph", and/or "print".  
 1000  
 1001  
 1002  
 1003

1004 **4.3.2 "i18n" LC\_CTYPE category**

1005

1006 The "i18n" FDCC-set for the LC\_CTYPE is defined as follows:

1007

1008

LC\_CTYPE

1009

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

1010

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

1011

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

1012

%

1013

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

1014

upper /

1015

% COLLECTION 1 BASIC LATIN/

1016

&lt;U0041&gt;..&lt;U005A&gt;;/

1017

% COLLECTION 2 LATIN-1 SUPPLEMENT/

1018

&lt;U00C0&gt;..&lt;U00D6&gt;;&lt;U00D8&gt;..&lt;U00DE&gt;;/

1019

% COLLECTION 3 LATIN EXTENDED-A/

1020

&lt;U0100&gt;..(2)..&lt;U0136&gt;;/

1021

&lt;U0139&gt;..(2)..&lt;U0147&gt;;/

1022

&lt;U014A&gt;..(2)..&lt;U0178&gt;;/

1023

&lt;U0179&gt;..(2)..&lt;U017D&gt;;/

1024

% COLLECTION 4 LATIN EXTENDED-B/

1025

&lt;U0181&gt;;&lt;U0182&gt;..(2)..&lt;U0186&gt;;&lt;U0187&gt;;/

1026

&lt;U0189&gt;..&lt;U018B&gt;;&lt;U018E&gt;..&lt;U0191&gt;;&lt;U0193&gt;;&lt;U0194&gt;;/

1027

&lt;U0196&gt;..&lt;U0198&gt;;&lt;U019C&gt;;&lt;U019D&gt;;&lt;U019F&gt;;/

1028

&lt;U01A0&gt;..(2)..&lt;U01A4&gt;;/

1029

&lt;U01A7&gt;;&lt;U01A9&gt;;&lt;U01AC&gt;;&lt;U01AE&gt;;&lt;U01AF&gt;;&lt;U01B1&gt;..&lt;U01B3&gt;;/

1030

&lt;U01B5&gt;;&lt;U01B7&gt;;&lt;U01B8&gt;;&lt;U01BC&gt;;&lt;U01C4&gt;;&lt;U01C5&gt;;&lt;U01C7&gt;;&lt;U01C8&gt;;/

1031

&lt;U01CA&gt;;&lt;U01CB&gt;;/

1032

&lt;U01CD&gt;..(2)..&lt;U01DB&gt;;/

1033

&lt;U01DE&gt;..(2)..&lt;U01EE&gt;;/

1034

&lt;U01F1&gt;;&lt;U01F2&gt;;&lt;U01F4&gt;;&lt;U01FA&gt;..(2)..&lt;U01FE&gt;;/

1035

&lt;U0200&gt;..(2)..&lt;U0216&gt;;/

1036

% COLLECTION 8 BASIC GREEK/

1037

&lt;U0386&gt;;&lt;U0388&gt;..&lt;U038A&gt;;&lt;U038C&gt;;&lt;U038E&gt;;&lt;U038F&gt;;&lt;U0391&gt;..&lt;U03A1&gt;;/

1038

&lt;U03A3&gt;..&lt;U03AB&gt;;&lt;U03D2&gt;..&lt;U03D4&gt;/

1039

% COLLECTION 9 GREEK SYMBOLS AND COPTIC/

1040

&lt;U03E2&gt;..(2)..&lt;U03EE&gt;;/

1041

% COLLECTION 10 CYRILLIC/

1042

&lt;U0401&gt;..&lt;U040C&gt;;&lt;U040E&gt;..&lt;U042F&gt;;&lt;U0460&gt;..(2)..&lt;U047E&gt;;/

1043

&lt;U0480&gt;;&lt;U0490&gt;..(2)..&lt;U04BE&gt;;&lt;U04C1&gt;;&lt;U04C3&gt;;&lt;U04C7&gt;;&lt;U04CB&gt;;/

1044

&lt;U04D0&gt;..(2)..&lt;U04EA&gt;;&lt;U04EE&gt;..(2)..&lt;U04F4&gt;;&lt;U04F8&gt;;/

1045

% COLLECTION 11 ARMENIAN/

1046

&lt;U0531&gt;..&lt;U0556&gt;;/

1047

% COLLECTION 28 GEORGIAN EXTENDED/

1048

&lt;U10A0&gt;..&lt;U10C5&gt;;/

1049

% COLLECTION 30 LATIN EXTENDED ADDITIONAL/

1050

&lt;U1E00&gt;..(2)..&lt;U1E7E&gt;;/

1051

&lt;U1E80&gt;..(2)..&lt;U1E94&gt;;/

1052

&lt;U1EA0&gt;..(2)..&lt;U1EF8&gt;;/

1053

% COLLECTION 31 GREEK EXTENDED/

1054

&lt;U1F08&gt;..&lt;U1F0F&gt;;&lt;U1F18&gt;..&lt;U1F1D&gt;;&lt;U1F28&gt;..&lt;U1F2F&gt;;&lt;U1F38&gt;..&lt;U1F3F&gt;;/

1055

&lt;U1F48&gt;..&lt;U1F4D&gt;;&lt;U1F59&gt;..(2)..&lt;U1F5F&gt;;&lt;U1F68&gt;..&lt;U1F6F&gt;;/

1056

&lt;U1F88&gt;..&lt;U1F8F&gt;;&lt;U1F98&gt;..&lt;U1F9F&gt;;&lt;U1FA8&gt;..&lt;U1FAF&gt;;&lt;U1FB8&gt;..&lt;U1FBC&gt;;/

1057

&lt;U1FC8&gt;..&lt;U1FCC&gt;;&lt;U1FD8&gt;..&lt;U1FDB&gt;;&lt;U1FE8&gt;..&lt;U1FEC&gt;;&lt;U1FF8&gt;..&lt;U1FFC&gt;

1058

% COLLECTION 28 GEORGIAN EXTENDED is not addressed as the letters does not

1059

% have a uppercase/lowercase relation

1060

%

1061

% The "lower" class reflects the lowercase characters of class "alpha"

1062

lower /

1063

% COLLECTION 1 BASIC LATIN/

1064

&lt;U0061&gt;..&lt;U007A&gt;;/

1065

% COLLECTION 2 LATIN-1 SUPPLEMENT/

1066

&lt;U00DF&gt;..&lt;U00F6&gt;;&lt;U00F8&gt;..&lt;U00FF&gt;;/

1067

% COLLECTION 3 LATIN EXTENDED-A/

1068

&lt;U0101&gt;..(2)..&lt;U0137&gt;;&lt;U0138&gt;..(2)..&lt;U0148&gt;;/

1069

&lt;U0149&gt;..(2)..&lt;U0177&gt;;&lt;U017A&gt;..(2)..&lt;U017E&gt;;&lt;U017F&gt;;/

1070

% COLLECTION 4 LATIN EXTENDED-B/

1071

&lt;U0180&gt;;&lt;U0183&gt;;&lt;U0185&gt;;&lt;U0188&gt;;&lt;U018C&gt;;&lt;U018D&gt;;&lt;U0192&gt;;&lt;U0195&gt;;/

1072

&lt;U0199&gt;..&lt;U019B&gt;;&lt;U019E&gt;;&lt;U01A1&gt;;&lt;U01A3&gt;;&lt;U01A5&gt;;&lt;U01A8&gt;;&lt;U01AB&gt;;&lt;U01AD&gt;;/

1073

&lt;U01B0&gt;;&lt;U01B4&gt;;&lt;U01B6&gt;;&lt;U01B9&gt;;&lt;U01BA&gt;;&lt;U01BD&gt;;&lt;U01C5&gt;;&lt;U01C6&gt;;/

1074

&lt;U01C8&gt;;&lt;U01C9&gt;;&lt;U01CB&gt;;&lt;U01CC&gt;..(2)..&lt;U01DC&gt;;/

1075

&lt;U01DD&gt;..(2)..&lt;U01F1&gt;;&lt;U01F3&gt;;&lt;U01F5&gt;;&lt;U01FB&gt;;&lt;U01FD&gt;;&lt;U01FF&gt;;/

1076

&lt;U0201&gt;..(2)..&lt;U0217&gt;;/

1077

% COLLECTION 5 IPA EXTENSIONS/

1078

&lt;U0250&gt;..&lt;U0293&gt;;&lt;U0299&gt;..&lt;U02A0&gt;;&lt;U02A3&gt;..&lt;U02A8&gt;;/

1079

% COLLECTION 8 BASIC GREEK/

```

1080     <U0390>;<U03AC>..<U03CE>;/
1081 % COLLECTION 9 GREEK SYMBOLS AND COPTIC/
1082     <U03E2>..(2)..<U03EE>;/
1083 % COLLECTION 10 CYRILLIC/
1084     <U0430>..<U044F>;<U0451>..<U045C>;<U045E>;<U045F>;<U0460>..(2)..<U047F>;/
1085     <U04801>;<U0490>..(2)..<U04BF>;<U04C2>;<U04C4>;<U04C8>;<U04CC>;/
1086     <U04D1>..(2)..<U04EB>;<U04EF>..(2)..<U04F5>;<U04F9>;/
1087 % COLLECTION 11 ARMENIAN/
1088     <U0561>..<U0587>;/
1089 % COLLECTION 28 GEORGIAN EXTENDED/
1090     <U10D0>..<U10F6>;/
1091 % COLLECTION 30 LATIN EXTENDED ADDITIONAL/
1092     <U1E01>..(2)..<U1E95>;<U1EA1>..(2)..<U1EF9>;/
1093 % COLLECTION 31 GREEK EXTENDED/
1094     <U1F08>..<U1F0F>;<U1F18>..<U1F1D>;<U1F28>..<U1F2F>;<U1F38>..<U1F3F>;/
1095     <U1F48>..<U1F4D>;<U1F59>..(2)..<U1F5F>;<U1F68>..<U1F6F>;/
1096     <U1F00>..<U1F07>;<U1F10>..<U1F15>;<U1F20>..<U1F27>;<U1F30>..<U1F37>;/
1097     <U1F40>..<U1F45>;<U1F50>..<U1F57>;<U1F60>..<U1F67>;<U1F70>..<U1F7D>;/
1098     <U1F80>..<U1F87>;<U1F90>..<U1F97>;<U1FA0>..<U1FA7>;<U1FB0>..<U1FB4>;/
1099     <U1FB6>;<U1FB7>;<U1FC2>..<U1FC4>;<U1FC6>;<U1FC7>;<U1FD0>..<U1FD3>;/
1100     <U1FD6>;<U1FD7>;<U1FE0>..<U1FE7>;<U1FF2>..<U1FF4>;<U1FF6>;<U1FF7>;/
1101 % COLLECTION 33 SUPERSCRIPTS AND SUBSCRIPTS/
1102     <U207F>
1103 %
1104 % The "alpha" class of the "i18n" FDCC-set is reflecting
1105 % the recommendations in TR 10176 annex A
1106 alpha /
1107 % COLLECTION 1 BASIC LATIN/
1108     <U0041>..<U005A>;<U0061>..<U007A>;/
1109 % COLLECTION 2 LATIN-1 SUPPLEMENT/
1110     <U00AA>;<U00BA>;<U00C0>..<U00D6>;<U00D8>..<U00F6>;<U00F8>..<U00FF>;/
1111 % COLLECTION 3 LATIN EXTENDED-A/
1112     <U0100>..<U017F>;/
1113 % COLLECTION 4 LATIN EXTENDED-B/
1114     <U0180>..<U01F5>;<U01FA>..<U0217>;/
1115 % COLLECTION 5 IPA EXTENSIONS/
1116     <U0250>..<U02A8>;/
1117 % COLLECTION 30 LATIN EXTENDED ADDITIONAL/
1118     <U1E00>..<U1E9B>;<U1EA0>..<U1EF9>;/
1119 % COLLECTION 33 SUPERSCRIPTS AND SUBSCRIPTS/
1120     <U207F>;/
1121 % COLLECTION 8 BASIC GREEK/
1122     <U0386>;<U0388>..<U038A>;<U038C>;<U038E>..<U03A1>;<U03A3>..<U03CE>;/
1123 % COLLECTION 9 GREEK SYMBOLS AND COPTIC/
1124     <U03D0>..<U03D6>;<U03DA>;<U03DC>;<U03DE>;<U03E0>;<U03E2>..<U03F3>;/
1125 % COLLECTION 31 GREEK EXTENDED/
1126     <U1F00>..<U1F15>;<U1F18>..<U1F1D>;<U1F20>..<U1F45>;<U1F48>..<U1F4D>;/
1127     <U1F50>..<U1F57>;<U1F59>;<U1F5B>;<U1F5D>;<U1F5F>..<U1F7D>;/
1128     <U1F80>..<U1FB4>;<U1FB6>..<U1FBC>;<U1FC2>..<U1FC4>;<U1FC6>..<U1FCC>;/
1129     <U1FD0>..<U1FD3>;<U1FD6>..<U1FDB>;<U1FE0>..<U1FEC>;<U1FF2>..<U1FF4>;/
1130     <U1FF6>..<U1FFC>;/
1131 % COLLECTION 10 CYRILLIC/
1132     <U0401>..<U040C>;<U040E>..<U044F>;<U0451>..<U045C>;<U045E>..<U0481>;/
1133     <U0490>..<U04C4>;<U04C7>..<U04C8>;<U04CB>..<U04CC>;<U04D0>..<U04EB>;/
1134     <U04EE>..<U04F5>;<U04F8>..<U04F9>;/
1135 % COLLECTION 11 ARMENIAN/
1136     <U0531>..<U0556>;<U0561>..<U0587>;/
1137 % COLLECTION 13 HEBREW EXTENDED/
1138     <U05B0>..<U05B9>;<U05BB>..<U05BD>;<U05BF>;<U05C1>..<U05C2>;/
1139     <U05D0>..<U05EA>;<U05F0>..<U05F2>;/
1140 % COLLECTION 15 ARABIC EXTENDED/
1141     <U0621>..<U063A>;<U0641>..<U064A>;<U0670>..<U06B7>;<U06BA>..<U06BE>;/
1142     <U06C0>..<U06CE>;<U06D0>..<U06D3>;<U06D5>..<U06DC>;<U06E5>..<U06E8>;/
1143 % COLLECTION 16 DEVANAGARI/
1144     <U0901>..<U0903>;<U0905>..<U0939>;<U093E>..<U094D>;<U0950>..<U0952>;/
1145     <U0958>..<U0963>;/
1146 % COLLECTION 17 BENGALI/
1147     <U0981>..<U0983>;<U0985>..<U098C>;<U098F>..<U0990>;/
1148     <U0993>..<U09A8>;<U09AA>..<U09B0>;<U09B2>;<U09B6>..<U09B9>;/
1149     <U09BE>..<U09C4>;<U09C7>..<U09C8>;<U09CB>..<U09CD>;<U09DC>..<U09DD>;/
1150     <U09DF>..<U09E3>;<U09F0>..<U09F1>;/
1151 % COLLECTION 18 GURMUKHI/
1152     <U0A02>;<U0A05>..<U0A0A>;<U0A0F>..<U0A10>;<U0A13>..<U0A28>;/
1153     <U0A2A>..<U0A30>;<U0A32>..<U0A33>;<U0A35>..<U0A36>;<U0A38>..<U0A39>;/
1154     <U0A3E>..<U0A42>;<U0A47>..<U0A48>;<U0A4B>..<U0A4D>;<U0A59>..<U0A5C>;/
1155     <U0A5E>;<U0A74>;/
1156 % COLLECTION 19 GUJARATI/
1157     <U0A81>..<U0A83>;<U0A85>..<U0A8B>;<U0A8D>;<U0A8F>..<U0A91>;/

```

```

1158     <U0A93>..<U0AA8>;<U0AAA>..<U0AB0>;<U0AB2>..<U0AB3>;<U0AB5>..<U0AB9>;/
1159     <U0ABD>..<U0AC5>;<U0AC7>..<U0AC9>;<U0ACB>..<U0ACD>;<U0AD0>;<U0AE0>;/
1160 % COLLECTION 20 ORIYA/
1161     <U0B01>..<U0B03>;<U0B05>..<U0B0C>;<U0B0F>..<U0B10>;<U0B13>..<U0B28>;/
1162     <U0B2A>..<U0B30>;<U0B32>..<U0B33>;<U0B36>..<U0B39>;<U0B3E>..<U0B43>;/
1163     <U0B47>..<U0B48>;<U0B4B>..<U0B4D>;<U0B5C>..<U0B5D>;<U0B5F>..<U0B61>;/
1164 % COLLECTION 21 TAMIL/
1165     <U0B82>..<U0B83>;<U0B85>..<U0B8A>;<U0B8E>..<U0B90>;<U0B92>..<U0B95>;/
1166     <U0B99>..<U0B9A>;<U0B9C>;<U0B9E>..<U0B9F>;<U0BA3>..<U0BA4>;/
1167     <U0BA8>..<U0BAA>;<U0BAE>..<U0BB5>;<U0BB7>..<U0BB9>;<U0BBE>..<U0BC2>;/
1168     <U0BC6>..<U0BC8>;<U0BCA>..<U0BCD>;/
1169 % COLLECTION 22 TELUGU/
1170     <U0C01>..<U0C03>;<U0C05>..<U0C0C>;<U0C0E>..<U0C10>;<U0C12>..<U0C28>;/
1171     <U0C2A>..<U0C33>;<U0C35>..<U0C39>;<U0C3E>..<U0C44>;<U0C46>..<U0C48>;/
1172     <U0C4A>..<U0C4D>;<U0C60>..<U0C61>;/
1173 % COLLECTION 23 KANNADA/
1174     <U0C82>..<U0C83>;<U0C85>..<U0C8C>;<U0C8E>..<U0C90>;<U0C92>..<U0CA8>;/
1175     <U0CAA>..<U0CB3>;<U0CB5>..<U0CB9>;<U0CBE>..<U0CC4>;<U0CC6>..<U0CC8>;/
1176     <U0CCA>..<U0CCD>;<U0CDE>;<U0CE0>..<U0CE1>;/
1177 % COLLECTION 24 MALAYALAM/
1178     <U0D02>..<U0D03>;<U0D05>..<U0D0C>;<U0D0E>..<U0D10>;<U0D12>..<U0D28>;/
1179     <U0D2A>..<U0D39>;<U0D3E>..<U0D43>;<U0D46>..<U0D48>;<U0D4A>..<U0D4D>;/
1180     <U0D60>..<U0D61>;/
1181 % COLLECTION 25 THAI/
1182     <U0E01>..<U0E3A>;<U0E40>..<U0E4E>;/
1183 % COLLECTION 26 LAO/
1184     <U0E81>..<U0E82>;<U0E84>;<U0E87>..<U0E88>;<U0E8A>;<U0E8D>;/
1185     <U0E94>..<U0E97>;<U0E99>..<U0E9F>;<U0EA1>..<U0EA3>;<U0EA5>;<U0EA7>;/
1186     <U0EAA>..<U0EAB>;<U0EAD>..<U0EAE>;<U0EB0>..<U0EB9>;<U0EBB>..<U0EBD>;/
1187     <U0EC0>..<U0EC4>;<U0EC6>;<U0EC8>..<U0ECD>;<U0EDC>..<U0EDD>;/
1188 % TIBETAN Amendment 6/
1189     <U0F00>;<U0F18>..<U0F19>;<U0F35>;<U0F37>;<U0F39>;<U0F40>..<U0F47>;/
1190     <U0F49>..<U0F69>;/
1191     <U0F71>..<U0F84>;<U0F86>..<U0F8B>;<U0F90>..<U0F95>;<U0F97>;/
1192     <U0F99>..<U0FAD>;<U0FB1>..<U0FB7>;<U0FB9>;/
1193 % COLLECTION 28 GEORGIAN EXTENDED/
1194     <U10A0>..<U10C5>;<U10D0>..<U10F6>;/
1195 % COLLECTION 50 HIRAGANA/
1196     <U3041>..<U3093>;<U309B>..<U309C>;/
1197 % COLLECTION 51 KATAKANA/
1198     <U30A1>..<U30F6>;<U30FB>..<U30FC>;/
1199 % COLLECTION 52 BOPOMOFO/
1200     <U3105>..<U312C>;/
1201 % CJK unified ideographs/
1202     <U4E00>..<U9FA5>;/
1203 % HANGUL amendment 5/
1204     <UAC00>..<UD7A3>;/
1205 % Miscellaneous/
1206     <U00B5>;<U02B0>..<U02B8>;<U02BB>;<U02BD>..<U02C1>;/
1207     <U02D0>..<U02D1>;<U02E0>..<U02E4>;<U037A>;<U0559>;<U093D>;<U0B3D>;/
1208     <U1FBE>;<U2160>..<U2182>;<U3021>..<U3029>
1209 %
1210 % The "digit" class of the "i18n" FDCC-set is reflecting
1211 % the recommendations in TR 10176 annex A
1212 digit /
1213 % COLLECTION 1 BASIC LATIN/
1214     <U0030>..<U0039>;/
1215 % COLLECTION 15 ARABIC EXTENDED/
1216     <U0660>..<U0669>;<U06F0>..<U06F9>;/
1217 % COLLECTION 16 DEVANAGARI/
1218     <U0966>..<U096F>;/
1219 % COLLECTION 18 BENGALI/
1220     <U09E6>..<U09EF>;/
1221 % COLLECTION 18 GURMUKHI/
1222     <U0A66>..<U0A6F>;/
1223 % COLLECTION 19 GUJARATI/
1224     <U0AE6>..<U0AEF>;/
1225 % COLLECTION 20 ORIYA/
1226     <U0B66>..<U0B6F>;/
1227 % COLLECTION 21 TAMIL/
1228     <0>;<U0BE7>..<U0BEF>;/
1229 % COLLECTION 22 TELUGU/
1230     <U0C66>..<U0C6F>;/
1231 % COLLECTION 23 KANNADA/
1232     <U0CE6>..<U0CEF>;/
1233 % COLLECTION 24 MALAYALAM/
1234     <U0D66>..<U0D6F>;/
1235 % COLLECTION 25 THAI/

```



```

1236     <U0E50>..<U0E59>;/
1237 % COLLECTION 26 LAO/
1238     <U0ED0>..<U0ED9>;/
1239 % TIBETAN Amendment 6/
1240     <U0F20>..<U0F29>
1241 %
1242 outdigit <U0030>..<U0039>
1243 %
1244 space /
1245 % ISO/IEC 6429/
1246     <U0008>;<U000A>..<U000D>;/
1247 % COLLECTION 1 BASIC LATIN/
1248     <U0020>;/
1249 % COLLECTION 35 GENERAL PUNCTUATION/
1250     <U2000>..<U2006>;<U2008>..<U200B>;/
1251 % COLLECTION 50 CJK SYMBOLS AND PUNCTUATION, HIRAGANA/
1252     <U3000>
1253 %
1254 cntrl <U0000>..<U001F>;<U007F>..<U009F>
1255 %
1256 punct /
1257     <U0021>..<U002F>;<U003A>..<U0040>;<U005B>..<U0060>;<U007B>..<U007E>;/
1258     <U00A0>..<U00A9>;<U00AB>..<U00B4>;<U00B6>..<U00B9>;<U00BB>..<U00BF>;/
1259     <U00D7>;<U00F7>;/
1260     <U037E>;<U0482>;<U055A>..<U055F>;<U0589>;<U05BE>;<U05C0>;<U05C3>;/
1261     <U05F3>;<U05F4>;<U060C>;<U061B>;<U061F>;<U0640>;<U064B>..<U0652>;/
1262     <U066A>..<U066D>;<U06D4>;<U06DD>..<U06E1>;<U06E9>..<U06EC>;<U10FB>;/
1263     <U2010>..<U2029>;<U2030>..<U2046>;<U20A0>..<U20AA>;<U2100>..<U210B>;/
1264     <U210D>..<U2110>;<U2112>..<U211B>;<U211D>..<U2127>;<U212A>..<U212C>;/
1265     <U212E>..<U2138>;<U2200>..<U22F1>;<U2300>;<U2302>..<U237A>;<U2400>..<U2424>;/
1266     <U2440>..<U244A>;<U2580>..<U2595>;<U25A0>..<U25EF>;<U2600>..<U2613>;/
1267     <U261A>..<U266F>;<U2701>..<U2704>;<U2706>..<U2709>;<U270C>..<U2727>;/
1268     <U2729>..<U274B>;<U274D>;<U274F>..<U2752>;<U2756>;<U2758>..<U275E>;/
1269     <U2761>..<U2767>;<U3000>..<U3020>;<U3030>;<U3036>;<U3037>;<U303F>;<U3164>;/
1270     <U3190>..<U319F>;<U3200>..<U321C>;<U3220>..<U3243>;<U3260>..<U327B>;/
1271     <U327F>..<U32B0>;<U32C0>..<U32CB>;<U32D0>..<U32FE>;<U3300>..<U3376>;/
1272     <U337B>..<U33DD>;<U33E0>..<U33FE>;<UFD3E>;<UFD3F>;<UFE49>..<UFE52>;/
1273     <UFE54>..<UFE66>;<UFE68>..<UFE6B>;<UFEFF>;<UFF01>..<UFF0F>;<UFF1A>..<UFF20>;/
1274     <UFF3B>..<UFF40>;<UFF5B>..<UFF5E>;<UFF61>..<UFF65>;<UFF70>;<UFF9E>..<UFFA0>;/
1275     <UFFE0>..<UFFE6>;<UFFE8>..<UFFEE>;<UFFFD>
1276 %
1277 graph /
1278     <U0021>..<U007E>;<U00A0>..<U01F5>;<U01FA>..<U0217>;/
1279     <U0250>..<U02A8>;<U02B0>..<U02DE>;<U02E0>..<U02E9>;<U0300>..<U0345>;/
1280     <U0360>;<U0361>;<U0374>;<U0375>;<U037A>;<U037E>;<U0384>..<U038A>;<U038C>;/
1281     <U038E>..<U03A1>;<U03A3>..<U03CE>;<U03D0>..<U03D6>;<U03DA>;<U03DC>;<U03DE>;/
1282     <U03E0>;<U03E2>..<U03F3>;<U0401>..<U040C>;<U040E>..<U044F>;/
1283     <U0451>..<U045C>;<U045E>..<U0486>;<U0490>..<U04C4>;<U04C7>;<U04C8>;/
1284     <U04CB>;<U04CC>;<U04D0>..<U04EB>;<U04EE>..<U04F5>;<U04F8>;<U04F9>;/
1285     <U0531>..<U0556>;<U0559>..<U055F>;<U0561>..<U0587>;<U0589>;/
1286     <U0591>..<U05A1>;<U05A3>..<U05AF>;<U05B0>..<U05B9>;/
1287     <U05BB>..<U05C4>;<U05D0>..<U05EA>;<U05F0>..<U05F4>;<U060C>;<U061B>;<U061F>;/
1288     <U0621>..<U063A>;<U0640>..<U0652>;<U0660>..<U066D>;<U0670>..<U06B7>;/
1289     <U06BA>..<U06BE>;<U06C0>..<U06CE>;<U06D0>..<U06ED>;<U06F0>..<U06F9>;/
1290     <U0901>..<U0903>;<U0905>..<U0939>;<U093C>..<U094D>;<U0950>..<U0954>;/
1291     <U0958>..<U0970>;<U0981>..<U0983>;<U0985>..<U098C>;<U098F>;<U0990>;/
1292     <U0993>..<U09A8>;<U09AA>..<U09B0>;<U09B2>;<U09B6>..<U09B9>;<U09BC>;/
1293     <U09BE>..<U09C4>;<U09C7>;<U09C8>;<U09CB>..<U09CD>;<U09D7>;<U09DC>;<U09DD>;/
1294     <U09DF>..<U09E3>;<U09E6>..<U09FA>;<U0A02>;<U0A05>..<U0A0A>;<U0A0F>;<U0A10>;/
1295     <U0A13>..<U0A28>;<U0A2A>..<U0A30>;<U0A32>;<U0A33>;<U0A35>;<U0A36>;/
1296     <U0A38>;<U0A39>;<U0A3C>;<U0A3E>..<U0A42>;<U0A47>;<U0A48>;<U0A4B>..<U0A4D>;/
1297     <U0A59>..<U0A5C>;<U0A5E>;<U0A66>..<U0A74>;<U0A81>..<U0A83>;<U0A85>..<U0A8B>;/
1298     <U0A8D>;<U0A8F>..<U0A91>;<U0A93>..<U0AA8>;<U0AAA>..<U0AB0>;/
1299     <U0AB2>;<U0AB3>;<U0AB5>..<U0AB9>;<U0ABC>..<U0AC5>;<U0AC7>..<U0AC9>;/
1300     <U0ACB>..<U0ACD>;<U0AD0>;<U0AE0>;<U0AE6>..<U0AEF>;<U0B01>..<U0B03>;/
1301     <U0B05>..<U0B0C>;<U0B0F>;<U0B10>;<U0B13>..<U0B28>;<U0B2A>..<U0B30>;/
1302     <U0B32>;<U0B33>;<U0B36>..<U0B39>;<U0B3C>..<U0B43>;<U0B47>;<U0B48>;/
1303     <U0B4B>..<U0B4D>;<U0B56>;<U0B57>;<U0B5C>;<U0B5D>;<U0B5F>..<U0B61>;/
1304     <U0B66>..<U0B70>;<U0B82>;<U0B83>;<U0B85>..<U0B8A>;<U0B8E>..<U0B90>;/
1305     <U0B92>..<U0B95>;<U0B99>;<U0B9A>;<U0B9C>;<U0B9E>;<U0B9F>;<U0BA3>;<U0BA4>;/
1306     <U0BA8>..<U0BAA>;<U0BAE>..<U0BB5>;<U0BB7>..<U0BB9>;<U0BBE>..<U0BC2>;/
1307     <U0BC6>..<U0BC8>;<U0BCA>..<U0BCD>;<U0BD7>;<U0BE7>..<U0BF2>;<U0C01>..<U0C03>;/
1308     <U0C05>..<U0C0C>;<U0C0E>..<U0C10>;<U0C12>..<U0C28>;<U0C2A>..<U0C33>;/
1309     <U0C35>..<U0C39>;<U0C3E>..<U0C44>;<U0C46>..<U0C48>;<U0C4A>..<U0C4D>;/
1310     <U0C55>;<U0C56>;<U0C60>;<U0C61>;<U0C66>..<U0C6F>;<U0C82>;<U0C83>;/
1311     <U0C85>..<U0C8C>;<U0C8E>..<U0C90>;<U0C92>..<U0CA8>;<U0CAA>..<U0CB3>;/
1312     <U0CB5>..<U0CB9>;<U0CBE>..<U0CC4>;<U0CC6>..<U0CC8>;<U0CCA>..<U0CCD>;/
1313     <U0CD5>;<U0CD6>;<U0CDE>;<U0CE0>;<U0CE1>;<U0CE6>..<U0CEF>;<U0D02>;<U0D03>;/

```

```

1314 <U0D05>..<U0D0C>;<U0D0E>..<U0D10>;<U0D12>..<U0D28>;<U0D2A>..<U0D39>;/
1315 <U0D3E>..<U0D43>;<U0D46>..<U0D48>;<U0D4A>..<U0D4D>;<U0D57>;<U0D60>;<U0D61>;/
1316 <U0D66>..<U0D6F>;<U0E01>..<U0E3A>;<U0E3F>..<U0E5B>;<U0E81>;<U0E82>;<U0E84>;/
1317 <U0E87>;<U0E88>;<U0E8A>;<U0E8D>;<U0E94>..<U0E97>;<U0E99>..<U0E9F>;/
1318 <U0EA1>..<U0EA3>;<U0EA5>;<U0EA7>;<U0EAA>;<U0EAB>;<U0EAD>..<U0EB9>;/
1319 <U0EBB>..<U0EBD>;<U0EC0>..<U0EC4>;<U0EC6>;<U0EC8>..<U0ECD>;<U0ED0>..<U0ED9>;/
1320 <U0EDC>;<U0EDD>;/
1321 <U0F00>..<U0F47>;<U0F49>..<U0F69>;<U0F71>..<U0F7F>;/
1322 <U10A0>..<U10C5>;<U10D0>..<U10F6>;<U10FB>;<U1100>..<U1159>;/
1323 <U115F>..<U11A2>;<U11A8>..<U11F9>;<U1E00>..<U1E9B>;<U1EA0>..<U1EF9>;/
1324 <U1F00>..<U1F15>;<U1F18>..<U1F1D>;<U1F20>..<U1F45>;<U1F48>..<U1F4D>;/
1325 <U1F50>..<U1F57>;<U1F59>;<U1F5B>;<U1F5D>;<U1F5F>..<U1F7D>;<U1F80>..<U1FB4>;/
1326 <U1FB6>..<U1FC4>;<U1FC6>..<U1FD3>;<U1FD6>..<U1FDB>;<U1FDD>..<U1FEF>;/
1327 <U1FF2>..<U1FF4>;<U1FF6>..<U1FFE>;<U2000>..<U202E>;<U2030>..<U2046>;/
1328 <U206A>..<U2070>;<U2074>..<U208E>;<U20A0>..<U20AC>;<U20D0>..<U20E1>;/
1329 <U2100>..<U2138>;<U2153>..<U2182>;<U2190>..<U21EA>;<U2200>..<U22F1>;<U2300>;/
1330 <U2302>..<U237A>;<U2400>..<U2424>;<U2440>..<U244A>;<U2460>..<U24EA>;/
1331 <U2500>..<U2595>;<U25A0>..<U25EF>;<U2600>..<U2613>;<U261A>..<U266F>;/
1332 <U2701>..<U2704>;<U2706>..<U2709>;<U270C>..<U2727>;<U2729>..<U274B>;<U274D>;/
1333 <U274F>..<U2752>;<U2756>;<U2758>..<U275E>;<U2761>..<U2767>;<U2776>..<U2794>;/
1334 <U2798>..<U27AF>;<U27B1>..<U27BE>;<U3000>..<U3037>;<U303F>;<U3041>..<U3094>;/
1335 <U3099>..<U309E>;<U30A1>..<U30FE>;<U3105>..<U312C>;<U3131>..<U318E>;/
1336 <U3190>..<U319F>;<U3200>..<U321C>;<U3220>..<U3243>;<U3260>..<U327B>;/
1337 <U327F>..<U32B0>;<U32C0>..<U32CB>;<U32D0>..<U32FE>;<U3300>..<U3376>;/
1338 <U337B>..<U33DD>;<U33E0>..<U33FE>;<UFB00>..<UFB06>;<UFB13>..<UFB17>;/
1339 <UFB1E>..<UFB36>;<UFB38>..<UFB3C>;<UFB3E>;<UFB40>;<UFB41>;<UFB43>;<UFB44>;/
1340 <UFB46>..<UFBB1>;<UFBDB3>..<UFD3F>;<UFD50>..<UFD8F>;<UFD92>..<UFD7C>;/
1341 <UFD70>..<UFD7B>;<UFE20>..<UFE23>;<UFE30>..<UFE44>;<UFE49>..<UFE52>;/
1342 <UFE54>..<UFE66>;<UFE68>..<UFE6B>;<UFE70>..<UFE72>;<UFE74>;<UFE76>..<UFEFC>;/
1343 <UFEFF>;<UFF01>..<UFF5E>;<UFF61>..<UFFBE>;<UFFC2>..<UFFC7>;/
1344 <UFFCA>..<UFFCF>;<UFFD2>..<UFFD7>;<UFFDA>..<UFFDC>;<UFFE0>..<UFFE6>;/
1345 <UFFE8>..<UFFEE>;<UFFFC>;<UFFFD>
1346 %
1347 % "print" is by default "graph", and the <space> character
1348 xdigit <U0030>..<U0039>;<U0041>..<U0046>;<U0061>..<U0066>
1349 blank <U0008>;<U0020>;<U2000>..<U2006>;<U2008>..<U200B>;<U3000>
1350 %
1351 toupper /
1352 (<U0061>,<U0041>);(<U0062>,<U0042>);(<U0063>,<U0043>);(<U0064>,<U0044>);/
1353 (<U0065>,<U0045>);(<U0066>,<U0046>);(<U0067>,<U0047>);(<U0068>,<U0048>);/
1354 (<U0069>,<U0049>);(<U006A>,<U004A>);(<U006B>,<U004B>);(<U006C>,<U004C>);/
1355 (<U006D>,<U004D>);(<U006E>,<U004E>);(<U006F>,<U004F>);(<U0070>,<U0050>);/
1356 (<U0071>,<U0051>);(<U0072>,<U0052>);(<U0073>,<U0053>);(<U0074>,<U0054>);/
1357 (<U0075>,<U0055>);(<U0076>,<U0056>);(<U0077>,<U0057>);(<U0078>,<U0058>);/
1358 (<U0079>,<U0059>);(<U007A>,<U005A>);(<U00E0>,<U00C0>);(<U00E1>,<U00C1>);/
1359 (<U00E2>,<U00C2>);(<U00E3>,<U00C3>);(<U00E4>,<U00C4>);(<U00E5>,<U00C5>);/
1360 (<U00E6>,<U00C6>);(<U00E7>,<U00C7>);(<U00E8>,<U00C8>);(<U00E9>,<U00C9>);/
1361 (<U00EA>,<U00CA>);(<U00EB>,<U00CB>);(<U00EC>,<U00CC>);(<U00ED>,<U00CD>);/
1362 (<U00EE>,<U00CE>);(<U00EF>,<U00CF>);(<U00F0>,<U00D0>);(<U00F1>,<U00D1>);/
1363 (<U00F2>,<U00D2>);(<U00F3>,<U00D3>);(<U00F4>,<U00D4>);(<U00F5>,<U00D5>);/
1364 (<U00F6>,<U00D6>);(<U00F7>,<U00D7>);(<U00F8>,<U00D8>);(<U00F9>,<U00D9>);/
1365 (<U00FA>,<U00DA>);(<U00FB>,<U00DB>);(<U00FC>,<U00DC>);(<U00FD>,<U00DD>);(<U00FE>,<U00DE>);/
1366 (<U00FF>,<U0178>);(<U0101>,<U0100>);(<U0103>,<U0102>);(<U0105>,<U0104>);/
1367 (<U0107>,<U0106>);(<U0109>,<U0108>);(<U010B>,<U010A>);(<U010D>,<U010C>);/
1368 (<U010F>,<U010E>);(<U0111>,<U0110>);(<U0113>,<U0112>);(<U0115>,<U0114>);/
1369 (<U0117>,<U0116>);(<U0119>,<U0118>);(<U011B>,<U011A>);(<U011D>,<U011C>);/
1370 (<U011F>,<U011E>);(<U0121>,<U0120>);(<U0123>,<U0122>);(<U0125>,<U0124>);/
1371 (<U0127>,<U0126>);(<U0129>,<U0128>);(<U012B>,<U012A>);(<U012D>,<U012C>);/
1372 (<U012F>,<U012E>);(<U0131>,<U0049>);/
1373 (<U0133>,<U0132>);(<U0135>,<U0134>);(<U0137>,<U0136>);/
1374 (<U013A>,<U0139>);(<U013C>,<U013B>);(<U013E>,<U013D>);(<U0140>,<U013F>);/
1375 (<U0142>,<U0141>);(<U0144>,<U0143>);(<U0146>,<U0145>);(<U0148>,<U0147>);/
1376 (<U014B>,<U014A>);(<U014D>,<U014C>);(<U014F>,<U014E>);(<U0151>,<U0150>);/
1377 (<U0153>,<U0152>);(<U0155>,<U0154>);(<U0157>,<U0156>);(<U0159>,<U0158>);/
1378 (<U015B>,<U015A>);(<U015D>,<U015C>);(<U015F>,<U015E>);(<U0161>,<U0160>);/
1379 (<U0163>,<U0162>);(<U0165>,<U0164>);(<U0167>,<U0166>);(<U0169>,<U0168>);/
1380 (<U016B>,<U016A>);(<U016D>,<U016C>);(<U016F>,<U016E>);(<U0171>,<U0170>);/
1381 (<U0173>,<U0172>);(<U0175>,<U0174>);(<U0177>,<U0176>);(<U017A>,<U0179>);/
1382 (<U017C>,<U017B>);(<U017E>,<U017D>);(<U017F>,<U0053>);(<U0183>,<U0182>);/
1383 (<U0185>,<U0184>);(<U0188>,<U0187>);(<U018C>,<U018B>);(<U0192>,<U0191>);/
1384 (<U0199>,<U0198>);(<U01A1>,<U01A0>);(<U01A3>,<U01A2>);(<U01A5>,<U01A4>);/
1385 (<U01A8>,<U01A7>);(<U01AD>,<U01AC>);(<U01B0>,<U01AF>);(<U01B4>,<U01B3>);/
1386 (<U01B6>,<U01B5>);(<U01B9>,<U01B8>);(<U01BD>,<U01BC>);(<U01C5>,<U01C4>);/
1387 (<U01C6>,<U01C4>);(<U01C8>,<U01C7>);/
1388 (<U01C9>,<U01C7>);(<U01CB>,<U01CA>);(<U01CC>,<U01CA>);/
1389 (<U01CE>,<U01CD>);(<U01D0>,<U01CF>);(<U01D2>,<U01D1>);(<U01D4>,<U01D3>);/
1390 (<U01D6>,<U01D5>);(<U01D8>,<U01D7>);(<U01DA>,<U01D9>);(<U01DC>,<U01DB>);/
1391 (<U01DD>,<U018E>);(<U01DF>,<U01DE>);(<U01E1>,<U01E0>);(<U01E3>,<U01E2>);/

```

1392 (<U01E5>, <U01E4>); (<U01E7>, <U01E6>); (<U01E9>, <U01E8>); (<U01EB>, <U01EA>); /  
 1393 (<U01ED>, <U01EC>); (<U01EF>, <U01EE>); (<U01F2>, <U01F1>); /  
 1394 (<U01F3>, <U01F1>); (<U01F5>, <U01F4>); (<U01FB>, <U01FA>); (<U01FD>, <U01FC>); /  
 1395 (<U01FF>, <U01FE>); (<U0201>, <U0200>); (<U0203>, <U0202>); (<U0205>, <U0204>); /  
 1396 (<U0207>, <U0206>); (<U0209>, <U0208>); (<U020B>, <U020A>); (<U020D>, <U020C>); /  
 1397 (<U020F>, <U020E>); (<U0211>, <U0210>); (<U0213>, <U0212>); (<U0215>, <U0214>); /  
 1398 (<U0217>, <U0216>); (<U0253>, <U0181>); (<U0254>, <U0186>); (<U0256>, <U0189>); /  
 1399 (<U0257>, <U018A>); (<U0259>, <U018F>); (<U025B>, <U0190>); (<U0260>, <U0193>); /  
 1400 (<U0263>, <U0194>); (<U0268>, <U0197>); (<U0269>, <U0196>); (<U026F>, <U019C>); /  
 1401 (<U0280>, <U01A6>); /  
 1402 (<U0272>, <U019D>); (<U0275>, <U019F>); (<U0283>, <U01A9>); (<U0288>, <U01AE>); /  
 1403 (<U028A>, <U01B1>); (<U028B>, <U01B2>); (<U0292>, <U01B7>); (<U03AC>, <U0386>); /  
 1404 (<U03AD>, <U0388>); (<U03AE>, <U0389>); (<U03AF>, <U038A>); (<U03B1>, <U0391>); /  
 1405 (<U03B2>, <U0392>); (<U03B3>, <U0393>); (<U03B4>, <U0394>); (<U03B5>, <U0395>); /  
 1406 (<U03B6>, <U0396>); (<U03B7>, <U0397>); (<U03B8>, <U0398>); (<U03B9>, <U0399>); /  
 1407 (<U03BA>, <U039A>); (<U03BB>, <U039B>); (<U03BC>, <U039C>); (<U03BD>, <U039D>); /  
 1408 (<U03BE>, <U039E>); (<U03BF>, <U039F>); (<U03C0>, <U03A0>); (<U03C1>, <U03A1>); /  
 1409 (<U03C2>, <U03A3>); (<U03C3>, <U03A3>); (<U03C4>, <U03A4>); (<U03C5>, <U03A5>); /  
 1410 (<U03C6>, <U03A6>); (<U03C7>, <U03A7>); (<U03C8>, <U03A8>); (<U03C9>, <U03A9>); /  
 1411 (<U03CA>, <U03AA>); (<U03CB>, <U03AB>); (<U03CC>, <U038C>); (<U03CD>, <U038E>); /  
 1412 (<U03CE>, <U038F>); /  
 1413 (<U03E3>, <U03E2>); (<U03E5>, <U03E4>); (<U03E7>, <U03E6>); (<U03E9>, <U03E8>); /  
 1414 (<U03EB>, <U03EA>); (<U03ED>, <U03EC>); (<U03EF>, <U03EE>); /  
 1415 (<U0430>, <U0410>); (<U0431>, <U0411>); (<U0432>, <U0412>); /  
 1416 (<U0433>, <U0413>); (<U0434>, <U0414>); (<U0435>, <U0415>); (<U0436>, <U0416>); /  
 1417 (<U0437>, <U0417>); (<U0438>, <U0418>); (<U0439>, <U0419>); (<U043A>, <U041A>); /  
 1418 (<U043B>, <U041B>); (<U043C>, <U041C>); (<U043D>, <U041D>); (<U043E>, <U041E>); /  
 1419 (<U043F>, <U041F>); (<U0440>, <U0420>); (<U0441>, <U0421>); (<U0442>, <U0422>); /  
 1420 (<U0443>, <U0423>); (<U0444>, <U0424>); (<U0445>, <U0425>); (<U0446>, <U0426>); /  
 1421 (<U0447>, <U0427>); (<U0448>, <U0428>); (<U0449>, <U0429>); (<U044A>, <U042A>); /  
 1422 (<U044B>, <U042B>); (<U044C>, <U042C>); (<U044D>, <U042D>); (<U044E>, <U042E>); /  
 1423 (<U044F>, <U042F>); (<U0451>, <U0401>); (<U0452>, <U0402>); (<U0453>, <U0403>); /  
 1424 (<U0454>, <U0404>); (<U0455>, <U0405>); (<U0456>, <U0406>); (<U0457>, <U0407>); /  
 1425 (<U0458>, <U0408>); (<U0459>, <U0409>); (<U045A>, <U040A>); (<U045B>, <U040B>); /  
 1426 (<U045C>, <U040C>); (<U045E>, <U040E>); (<U045F>, <U040F>); (<U0461>, <U0460>); /  
 1427 (<U0463>, <U0462>); (<U0465>, <U0464>); (<U0467>, <U0466>); (<U0469>, <U0468>); /  
 1428 (<U046B>, <U046A>); (<U046D>, <U046C>); (<U046F>, <U046E>); (<U0471>, <U0470>); /  
 1429 (<U0473>, <U0472>); (<U0475>, <U0474>); (<U0477>, <U0476>); (<U0479>, <U0478>); /  
 1430 (<U047B>, <U047A>); (<U047D>, <U047C>); (<U047F>, <U047E>); (<U0481>, <U0480>); /  
 1431 (<U0491>, <U0490>); (<U0493>, <U0492>); (<U0495>, <U0494>); (<U0497>, <U0496>); /  
 1432 (<U0499>, <U0498>); (<U049B>, <U049A>); (<U049D>, <U049C>); (<U049F>, <U049E>); /  
 1433 (<U04A1>, <U04A0>); (<U04A3>, <U04A2>); (<U04A5>, <U04A4>); (<U04A7>, <U04A6>); /  
 1434 (<U04A9>, <U04A8>); (<U04AB>, <U04AA>); (<U04AD>, <U04AC>); (<U04AF>, <U04AE>); /  
 1435 (<U04B1>, <U04B0>); (<U04B3>, <U04B2>); (<U04B5>, <U04B4>); (<U04B7>, <U04B6>); /  
 1436 (<U04B9>, <U04B8>); (<U04BB>, <U04BA>); (<U04BD>, <U04BC>); (<U04BF>, <U04BE>); /  
 1437 (<U04C2>, <U04C1>); (<U04C4>, <U04C3>); (<U04C8>, <U04C7>); (<U04CC>, <U04CB>); /  
 1438 (<U04D1>, <U04D0>); (<U04D3>, <U04D2>); (<U04D5>, <U04D4>); (<U04D7>, <U04D6>); /  
 1439 (<U04D9>, <U04D8>); (<U04DB>, <U04DA>); (<U04DD>, <U04DC>); (<U04DF>, <U04DE>); /  
 1440 (<U04E1>, <U04E0>); (<U04E3>, <U04E2>); (<U04E5>, <U04E4>); (<U04E7>, <U04E6>); /  
 1441 (<U04E9>, <U04E8>); (<U04EB>, <U04EA>); (<U04EF>, <U04EE>); (<U04F1>, <U04F0>); /  
 1442 (<U04F3>, <U04F2>); (<U04F5>, <U04F4>); (<U04F9>, <U04F8>); (<U0561>, <U0531>); /  
 1443 (<U0562>, <U0532>); (<U0563>, <U0533>); (<U0564>, <U0534>); (<U0565>, <U0535>); /  
 1444 (<U0566>, <U0536>); (<U0567>, <U0537>); (<U0568>, <U0538>); (<U0569>, <U0539>); /  
 1445 (<U056A>, <U053A>); (<U056B>, <U053B>); (<U056C>, <U053C>); (<U056D>, <U053D>); /  
 1446 (<U056E>, <U053E>); (<U056F>, <U053F>); (<U0570>, <U0540>); (<U0571>, <U0541>); /  
 1447 (<U0572>, <U0542>); (<U0573>, <U0543>); (<U0574>, <U0544>); (<U0575>, <U0545>); /  
 1448 (<U0576>, <U0546>); (<U0577>, <U0547>); (<U0578>, <U0548>); (<U0579>, <U0549>); /  
 1449 (<U057A>, <U054A>); (<U057B>, <U054B>); (<U057C>, <U054C>); (<U057D>, <U054D>); /  
 1450 (<U057E>, <U054E>); (<U057F>, <U054F>); (<U0580>, <U0550>); (<U0581>, <U0551>); /  
 1451 (<U0582>, <U0552>); (<U0583>, <U0553>); (<U0584>, <U0554>); (<U0585>, <U0555>); /  
 1452 (<U0586>, <U0556>); /  
 1453 (<U1E01>, <U1E00>); (<U1E03>, <U1E02>); (<U1E05>, <U1E04>); /  
 1454 (<U1E07>, <U1E06>); (<U1E09>, <U1E08>); (<U1E0B>, <U1E0A>); (<U1E0D>, <U1E0C>); /  
 1455 (<U1E0F>, <U1E0E>); (<U1E11>, <U1E10>); (<U1E13>, <U1E12>); (<U1E15>, <U1E14>); /  
 1456 (<U1E17>, <U1E16>); (<U1E19>, <U1E18>); (<U1E1B>, <U1E1A>); (<U1E1D>, <U1E1C>); /  
 1457 (<U1E1F>, <U1E1E>); (<U1E21>, <U1E20>); (<U1E23>, <U1E22>); (<U1E25>, <U1E24>); /  
 1458 (<U1E27>, <U1E26>); (<U1E29>, <U1E28>); (<U1E2B>, <U1E2A>); (<U1E2D>, <U1E2C>); /  
 1459 (<U1E2F>, <U1E2E>); (<U1E31>, <U1E30>); (<U1E33>, <U1E32>); (<U1E35>, <U1E34>); /  
 1460 (<U1E37>, <U1E36>); (<U1E39>, <U1E38>); (<U1E3B>, <U1E3A>); (<U1E3D>, <U1E3C>); /  
 1461 (<U1E3F>, <U1E3E>); (<U1E41>, <U1E40>); (<U1E43>, <U1E42>); (<U1E45>, <U1E44>); /  
 1462 (<U1E47>, <U1E46>); (<U1E49>, <U1E48>); (<U1E4B>, <U1E4A>); (<U1E4D>, <U1E4C>); /  
 1463 (<U1E4F>, <U1E4E>); (<U1E51>, <U1E50>); (<U1E53>, <U1E52>); (<U1E55>, <U1E54>); /  
 1464 (<U1E57>, <U1E56>); (<U1E59>, <U1E58>); (<U1E5B>, <U1E5A>); (<U1E5D>, <U1E5C>); /  
 1465 (<U1E5F>, <U1E5E>); (<U1E61>, <U1E60>); (<U1E63>, <U1E62>); (<U1E65>, <U1E64>); /  
 1466 (<U1E67>, <U1E66>); (<U1E69>, <U1E68>); (<U1E6B>, <U1E6A>); (<U1E6D>, <U1E6C>); /  
 1467 (<U1E6F>, <U1E6E>); (<U1E71>, <U1E70>); (<U1E73>, <U1E72>); (<U1E75>, <U1E74>); /  
 1468 (<U1E77>, <U1E76>); (<U1E79>, <U1E78>); (<U1E7B>, <U1E7A>); (<U1E7D>, <U1E7C>); /  
 1469 (<U1E7F>, <U1E7E>); (<U1E81>, <U1E80>); (<U1E83>, <U1E82>); (<U1E85>, <U1E84>); /

1470 (<U1E87>, <U1E86>); (<U1E89>, <U1E88>); (<U1E8B>, <U1E8A>); (<U1E8D>, <U1E8C>); /  
 1471 (<U1E8F>, <U1E8E>); (<U1E91>, <U1E90>); (<U1E93>, <U1E92>); (<U1E95>, <U1E94>); /  
 1472 (<U1E9B>, <U1E9A>); /  
 1473 (<U1EA1>, <U1EA0>); (<U1EA3>, <U1EA2>); (<U1EA5>, <U1EA4>); (<U1EA7>, <U1EA6>); /  
 1474 (<U1EA9>, <U1EA8>); (<U1EAB>, <U1EAA>); (<U1EAD>, <U1EAC>); (<U1EAF>, <U1EAE>); /  
 1475 (<U1EB1>, <U1EB0>); (<U1EB3>, <U1EB2>); (<U1EB5>, <U1EB4>); (<U1EB7>, <U1EB6>); /  
 1476 (<U1EB9>, <U1EB8>); (<U1EBB>, <U1EBA>); (<U1EBD>, <U1EBC>); (<U1EBF>, <U1EBE>); /  
 1477 (<U1EC1>, <U1EC0>); (<U1EC3>, <U1EC2>); (<U1EC5>, <U1EC4>); (<U1EC7>, <U1EC6>); /  
 1478 (<U1EC9>, <U1EC8>); (<U1ECB>, <U1ECA>); (<U1ECD>, <U1ECC>); (<U1ECF>, <U1ECE>); /  
 1479 (<U1ED1>, <U1ED0>); (<U1ED3>, <U1ED2>); (<U1ED5>, <U1ED4>); (<U1ED7>, <U1ED6>); /  
 1480 (<U1ED9>, <U1ED8>); (<U1EDB>, <U1EDA>); (<U1EDD>, <U1EDC>); (<U1EDF>, <U1EDE>); /  
 1481 (<U1EE1>, <U1EE0>); (<U1EE3>, <U1EE2>); (<U1EE5>, <U1EE4>); (<U1EE7>, <U1EE6>); /  
 1482 (<U1EE9>, <U1EE8>); (<U1EEB>, <U1EEA>); (<U1EED>, <U1EEC>); (<U1EEF>, <U1EEE>); /  
 1483 (<U1EF1>, <U1EF0>); (<U1EF3>, <U1EF2>); (<U1EF5>, <U1EF4>); (<U1EF7>, <U1EF6>); /  
 1484 (<U1EF9>, <U1EF8>); (<U1F00>, <U1F08>); (<U1F01>, <U1F09>); (<U1F02>, <U1F0A>); /  
 1485 (<U1F03>, <U1F0B>); (<U1F04>, <U1F0C>); (<U1F05>, <U1F0D>); (<U1F06>, <U1F0E>); /  
 1486 (<U1F07>, <U1F0F>); (<U1F10>, <U1F18>); (<U1F11>, <U1F19>); (<U1F12>, <U1F1A>); /  
 1487 (<U1F13>, <U1F1B>); (<U1F14>, <U1F1C>); (<U1F15>, <U1F1D>); (<U1F20>, <U1F28>); /  
 1488 (<U1F21>, <U1F29>); (<U1F22>, <U1F2A>); (<U1F23>, <U1F2B>); (<U1F24>, <U1F2C>); /  
 1489 (<U1F25>, <U1F2D>); (<U1F26>, <U1F2E>); (<U1F27>, <U1F2F>); (<U1F30>, <U1F38>); /  
 1490 (<U1F31>, <U1F39>); (<U1F32>, <U1F3A>); (<U1F33>, <U1F3B>); (<U1F34>, <U1F3C>); /  
 1491 (<U1F35>, <U1F3D>); (<U1F36>, <U1F3E>); (<U1F37>, <U1F3F>); (<U1F40>, <U1F48>); /  
 1492 (<U1F41>, <U1F49>); (<U1F42>, <U1F4A>); (<U1F43>, <U1F4B>); (<U1F44>, <U1F4C>); /  
 1493 (<U1F45>, <U1F4D>); (<U1F51>, <U1F59>); (<U1F53>, <U1F5B>); (<U1F55>, <U1F5D>); /  
 1494 (<U1F57>, <U1F5F>); (<U1F60>, <U1F68>); (<U1F61>, <U1F69>); (<U1F62>, <U1F6A>); /  
 1495 (<U1F63>, <U1F6B>); (<U1F64>, <U1F6C>); (<U1F65>, <U1F6D>); (<U1F66>, <U1F6E>); /  
 1496 (<U1F67>, <U1F6F>); (<U1F70>, <U1F7A>); (<U1F71>, <U1F7B>); (<U1F72>, <U1F7C>); /  
 1497 (<U1F73>, <U1F7D>); (<U1F74>, <U1F7E>); (<U1F75>, <U1F7F>); (<U1F76>, <U1F78>); /  
 1498 (<U1F77>, <U1F79>); (<U1F78>, <U1F7A>); (<U1F79>, <U1F7B>); (<U1F7A>, <U1F7C>); /  
 1499 (<U1F7B>, <U1F7D>); (<U1F7C>, <U1F7E>); (<U1F7D>, <U1F7F>); (<U1F7E>, <U1F78>); /  
 1500 (<U1F81>, <U1F89>); (<U1F82>, <U1F8A>); (<U1F83>, <U1F8B>); (<U1F84>, <U1F8C>); /  
 1501 (<U1F85>, <U1F8D>); (<U1F86>, <U1F8E>); (<U1F87>, <U1F8F>); (<U1F90>, <U1F98>); /  
 1502 (<U1F91>, <U1F99>); (<U1F92>, <U1F9A>); (<U1F93>, <U1F9B>); (<U1F94>, <U1F9C>); /  
 1503 (<U1F95>, <U1F9D>); (<U1F96>, <U1F9E>); (<U1F97>, <U1F9F>); (<U1FA0>, <U1FA8>); /  
 1504 (<U1FA1>, <U1FA9>); (<U1FA2>, <U1FAA>); (<U1FA3>, <U1FAB>); (<U1FA4>, <U1FAC>); /  
 1505 (<U1FA5>, <U1FAD>); (<U1FA6>, <U1FAE>); (<U1FA7>, <U1FAF>); (<U1FB0>, <U1FB8>); /  
 1506 (<U1FB1>, <U1FB9>); (<U1FB3>, <U1FBC>); (<U1FC3>, <U1FCC>); (<U1FD0>, <U1FD8>); /  
 1507 (<U1FD1>, <U1FD9>); (<U1FE0>, <U1FE8>); (<U1FE1>, <U1FE9>); (<U1FE5>, <U1FEC>); /  
 1508 (<U1FF3>, <U1FFC>); /  
 1509 tolower /  
 1510 (<U0041>, <U0061>); (<U0042>, <U0062>); (<U0043>, <U0063>); (<U0044>, <U0064>); /  
 1511 (<U0045>, <U0065>); (<U0046>, <U0066>); (<U0047>, <U0067>); (<U0048>, <U0068>); /  
 1512 (<U0049>, <U0069>); (<U004A>, <U006A>); (<U004B>, <U006B>); (<U004C>, <U006C>); /  
 1513 (<U004D>, <U006D>); (<U004E>, <U006E>); (<U004F>, <U006F>); (<U0050>, <U0070>); /  
 1514 (<U0051>, <U0071>); (<U0052>, <U0072>); (<U0053>, <U0073>); (<U0054>, <U0074>); /  
 1515 (<U0055>, <U0075>); (<U0056>, <U0076>); (<U0057>, <U0077>); (<U0058>, <U0078>); /  
 1516 (<U0059>, <U0079>); (<U005A>, <U007A>); (<U00C0>, <U00E0>); (<U00C1>, <U00E1>); /  
 1517 (<U00C2>, <U00E2>); (<U00C3>, <U00E3>); (<U00C4>, <U00E4>); (<U00C5>, <U00E5>); /  
 1518 (<U00C6>, <U00E6>); (<U00C7>, <U00E7>); (<U00C8>, <U00E8>); (<U00C9>, <U00E9>); /  
 1519 (<U00CA>, <U00EA>); (<U00CB>, <U00EB>); (<U00CC>, <U00EC>); (<U00CD>, <U00ED>); /  
 1520 (<U00CE>, <U00EE>); (<U00CF>, <U00EF>); (<U00D0>, <U00F0>); (<U00D1>, <U00F1>); /  
 1521 (<U00D2>, <U00F2>); (<U00D3>, <U00F3>); (<U00D4>, <U00F4>); (<U00D5>, <U00F5>); /  
 1522 (<U00D6>, <U00F6>); (<U00D8>, <U00F8>); (<U00D9>, <U00F9>); (<U00DA>, <U00FA>); /  
 1523 (<U00DB>, <U00FB>); (<U00DC>, <U00FC>); (<U00DD>, <U00FD>); (<U00DE>, <U00FE>); /  
 1524 (<U0178>, <U00FF>); (<U0100>, <U0101>); (<U0102>, <U0103>); (<U0104>, <U0105>); /  
 1525 (<U0106>, <U0107>); (<U0108>, <U0109>); (<U010A>, <U010B>); (<U010C>, <U010D>); /  
 1526 (<U010E>, <U010F>); (<U0110>, <U0111>); (<U0112>, <U0113>); (<U0114>, <U0115>); /  
 1527 (<U0116>, <U0117>); (<U0118>, <U0119>); (<U011A>, <U011B>); (<U011C>, <U011D>); /  
 1528 (<U011E>, <U011F>); (<U0120>, <U0121>); (<U0122>, <U0123>); (<U0124>, <U0125>); /  
 1529 (<U0126>, <U0127>); (<U0128>, <U0129>); (<U012A>, <U012B>); (<U012C>, <U012D>); /  
 1530 (<U012E>, <U012F>); (<U0130>, <U0069>); /  
 1531 (<U0132>, <U0133>); (<U0134>, <U0135>); (<U0136>, <U0137>); /  
 1532 (<U0139>, <U013A>); (<U013B>, <U013C>); (<U013D>, <U013E>); (<U013F>, <U0140>); /  
 1533 (<U0141>, <U0142>); (<U0143>, <U0144>); (<U0145>, <U0146>); (<U0147>, <U0148>); /  
 1534 (<U014A>, <U014B>); (<U014C>, <U014D>); (<U014E>, <U014F>); (<U0150>, <U0151>); /  
 1535 (<U0152>, <U0153>); (<U0154>, <U0155>); (<U0156>, <U0157>); (<U0158>, <U0159>); /  
 1536 (<U015A>, <U015B>); (<U015C>, <U015D>); (<U015E>, <U015F>); (<U0160>, <U0161>); /  
 1537 (<U0162>, <U0163>); (<U0164>, <U0165>); (<U0166>, <U0167>); (<U0168>, <U0169>); /  
 1538 (<U016A>, <U016B>); (<U016C>, <U016D>); (<U016E>, <U016F>); (<U0170>, <U0171>); /  
 1539 (<U0172>, <U0173>); (<U0174>, <U0175>); (<U0176>, <U0177>); (<U0179>, <U017A>); /  
 1540 (<U017B>, <U017C>); (<U017D>, <U017E>); (<U0182>, <U0183>); (<U0184>, <U0185>); /  
 1541 (<U0187>, <U0188>); (<U0256>, <U0189>); (<U018B>, <U018C>); (<U018E>, <U01DD>); /  
 1542 (<U0191>, <U0192>); (<U0198>, <U0199>); (<U019F>, <U0275>); /  
 1543 (<U01A0>, <U01A1>); (<U01A2>, <U01A3>); /  
 1544 (<U01A4>, <U01A5>); (<U01A6>, <U0280>); /  
 1545 (<U01A7>, <U01A8>); (<U01AC>, <U01AD>); (<U01AF>, <U01B0>); /  
 1546 (<U01B3>, <U01B4>); (<U01B5>, <U01B6>); (<U01B8>, <U01B9>); (<U01BC>, <U01BD>); /  
 1547 (<U01C4>, <U01C6>); (<U01C5>, <U01C6>); (<U01C7>, <U01C9>); /

1548 (<U01C8>, <U01C9>); (<U01CA>, <U01CC>); (<U01CB>, <U01CC>); /  
 1549 (<U01CD>, <U01CE>); (<U01CF>, <U01D0>); (<U01D1>, <U01D2>); /  
 1550 (<U01D3>, <U01D4>); (<U01D5>, <U01D6>); (<U01D7>, <U01D8>); (<U01D9>, <U01DA>); /  
 1551 (<U01DB>, <U01DC>); (<U01DE>, <U01DF>); (<U01E0>, <U01E1>); (<U01E2>, <U01E3>); /  
 1552 (<U01E4>, <U01E5>); (<U01E6>, <U01E7>); (<U01E8>, <U01E9>); (<U01EA>, <U01EB>); /  
 1553 (<U01EC>, <U01ED>); (<U01EE>, <U01EF>); (<U01F1>, <U01F3>); /  
 1554 (<U01F2>, <U01F3>); (<U01F4>, <U01F5>); (<U01FA>, <U01FB>); (<U01FC>, <U01FD>); /  
 1555 (<U01FE>, <U01FF>); (<U0200>, <U0201>); (<U0202>, <U0203>); (<U0204>, <U0205>); /  
 1556 (<U0206>, <U0207>); (<U0208>, <U0209>); (<U020A>, <U020B>); (<U020C>, <U020D>); /  
 1557 (<U020E>, <U020F>); (<U0210>, <U0211>); (<U0212>, <U0213>); (<U0214>, <U0215>); /  
 1558 (<U0216>, <U0217>); (<U0181>, <U0253>); (<U0186>, <U0254>); (<U018A>, <U0257>); /  
 1559 (<U018E>, <U0258>); (<U018F>, <U0259>); (<U0190>, <U025B>); (<U0193>, <U0260>); /  
 1560 (<U0194>, <U0263>); (<U0197>, <U0268>); (<U0196>, <U0269>); (<U019C>, <U026F>); /  
 1561 (<U019D>, <U0272>); (<U01A9>, <U0283>); (<U01AE>, <U0288>); (<U01B1>, <U028A>); /  
 1562 (<U01B2>, <U028B>); (<U01B7>, <U0292>); (<U0386>, <U03AC>); (<U0388>, <U03AD>); /  
 1563 (<U0389>, <U03AE>); (<U038A>, <U03AF>); (<U0391>, <U03B1>); (<U0392>, <U03B2>); /  
 1564 (<U0393>, <U03B3>); (<U0394>, <U03B4>); (<U0395>, <U03B5>); (<U0396>, <U03B6>); /  
 1565 (<U0397>, <U03B7>); (<U0398>, <U03B8>); (<U0399>, <U03B9>); (<U039A>, <U03BA>); /  
 1566 (<U039B>, <U03BB>); (<U039C>, <U03BC>); (<U039D>, <U03BD>); (<U039E>, <U03BE>); /  
 1567 (<U039F>, <U03BF>); (<U03A0>, <U03C0>); (<U03A1>, <U03C1>); (<U03A3>, <U03C3>); /  
 1568 (<U03A4>, <U03C4>); (<U03A5>, <U03C5>); (<U03A6>, <U03C6>); (<U03A7>, <U03C7>); /  
 1569 (<U03A8>, <U03C8>); (<U03A9>, <U03C9>); (<U03AA>, <U03CA>); (<U03AB>, <U03CB>); /  
 1570 (<U038C>, <U03CC>); (<U038E>, <U03CD>); (<U038F>, <U03CE>); (<U03E2>, <U03E3>); /  
 1571 (<U03E4>, <U03E5>); (<U03E6>, <U03E7>); (<U03E8>, <U03E9>); (<U03EA>, <U03EB>); /  
 1572 (<U03EC>, <U03ED>); (<U03EE>, <U03EF>); (<U0410>, <U0430>); /  
 1573 (<U0411>, <U0431>); (<U0412>, <U0432>); (<U0413>, <U0433>); (<U0414>, <U0434>); /  
 1574 (<U0415>, <U0435>); (<U0416>, <U0436>); (<U0417>, <U0437>); (<U0418>, <U0438>); /  
 1575 (<U0419>, <U0439>); (<U041A>, <U043A>); (<U041B>, <U043B>); (<U041C>, <U043C>); /  
 1576 (<U041D>, <U043D>); (<U041E>, <U043E>); (<U041F>, <U043F>); (<U0420>, <U0440>); /  
 1577 (<U0421>, <U0441>); (<U0422>, <U0442>); (<U0423>, <U0443>); (<U0424>, <U0444>); /  
 1578 (<U0425>, <U0445>); (<U0426>, <U0446>); (<U0427>, <U0447>); (<U0428>, <U0448>); /  
 1579 (<U0429>, <U0449>); (<U042A>, <U044A>); (<U042B>, <U044B>); (<U042C>, <U044C>); /  
 1580 (<U042D>, <U044D>); (<U042E>, <U044E>); (<U042F>, <U044F>); (<U0401>, <U0451>); /  
 1581 (<U0402>, <U0452>); (<U0403>, <U0453>); (<U0404>, <U0454>); (<U0405>, <U0455>); /  
 1582 (<U0406>, <U0456>); (<U0407>, <U0457>); (<U0408>, <U0458>); (<U0409>, <U0459>); /  
 1583 (<U040A>, <U045A>); (<U040B>, <U045B>); (<U040C>, <U045C>); (<U040E>, <U045E>); /  
 1584 (<U040F>, <U045F>); (<U0460>, <U0461>); (<U0462>, <U0463>); (<U0464>, <U0465>); /  
 1585 (<U0466>, <U0467>); (<U0468>, <U0469>); (<U046A>, <U046B>); (<U046C>, <U046D>); /  
 1586 (<U046E>, <U046F>); (<U0470>, <U0471>); (<U0472>, <U0473>); (<U0474>, <U0475>); /  
 1587 (<U0476>, <U0477>); (<U0478>, <U0479>); (<U047A>, <U047B>); (<U047C>, <U047D>); /  
 1588 (<U047E>, <U047F>); (<U0480>, <U0481>); (<U0490>, <U0491>); (<U0492>, <U0493>); /  
 1589 (<U0494>, <U0495>); (<U0496>, <U0497>); (<U0498>, <U0499>); (<U049A>, <U049B>); /  
 1590 (<U049C>, <U049D>); (<U049E>, <U049F>); (<U04A0>, <U04A1>); (<U04A2>, <U04A3>); /  
 1591 (<U04A4>, <U04A5>); (<U04A6>, <U04A7>); (<U04A8>, <U04A9>); (<U04AA>, <U04AB>); /  
 1592 (<U04AC>, <U04AD>); (<U04AE>, <U04AF>); (<U04B0>, <U04B1>); (<U04B2>, <U04B3>); /  
 1593 (<U04B4>, <U04B5>); (<U04B6>, <U04B7>); (<U04B8>, <U04B9>); (<U04BA>, <U04BB>); /  
 1594 (<U04BC>, <U04BD>); (<U04BE>, <U04BF>); (<U04C1>, <U04C2>); (<U04C3>, <U04C4>); /  
 1595 (<U04C7>, <U04C8>); (<U04CB>, <U04CC>); (<U04D0>, <U04D1>); (<U04D2>, <U04D3>); /  
 1596 (<U04D4>, <U04D5>); (<U04D6>, <U04D7>); (<U04D8>, <U04D9>); (<U04DA>, <U04DB>); /  
 1597 (<U04DC>, <U04DD>); (<U04DE>, <U04DF>); (<U04E0>, <U04E1>); (<U04E2>, <U04E3>); /  
 1598 (<U04E4>, <U04E5>); (<U04E6>, <U04E7>); (<U04E8>, <U04E9>); (<U04EA>, <U04EB>); /  
 1599 (<U04EE>, <U04EF>); (<U04F0>, <U04F1>); (<U04F2>, <U04F3>); (<U04F4>, <U04F5>); /  
 1600 (<U04F8>, <U04F9>); (<U0531>, <U0561>); (<U0532>, <U0562>); (<U0533>, <U0563>); /  
 1601 (<U0534>, <U0564>); (<U0535>, <U0565>); (<U0536>, <U0566>); (<U0537>, <U0567>); /  
 1602 (<U0538>, <U0568>); (<U0539>, <U0569>); (<U053A>, <U056A>); (<U053B>, <U056B>); /  
 1603 (<U053C>, <U056C>); (<U053D>, <U056D>); (<U053E>, <U056E>); (<U053F>, <U056F>); /  
 1604 (<U0540>, <U0570>); (<U0541>, <U0571>); (<U0542>, <U0572>); (<U0543>, <U0573>); /  
 1605 (<U0544>, <U0574>); (<U0545>, <U0575>); (<U0546>, <U0576>); (<U0547>, <U0577>); /  
 1606 (<U0548>, <U0578>); (<U0549>, <U0579>); (<U054A>, <U057A>); (<U054B>, <U057B>); /  
 1607 (<U054C>, <U057C>); (<U054D>, <U057D>); (<U054E>, <U057E>); (<U054F>, <U057F>); /  
 1608 (<U0550>, <U0580>); (<U0551>, <U0581>); (<U0552>, <U0582>); (<U0553>, <U0583>); /  
 1609 (<U1E02>, <U1E03>); (<U1E04>, <U1E05>); (<U1E06>, <U1E07>); (<U1E08>, <U1E09>); /  
 1610 (<U1E0A>, <U1E0B>); (<U1E0C>, <U1E0D>); (<U1E0E>, <U1E0F>); (<U1E10>, <U1E11>); /  
 1611 (<U1E12>, <U1E13>); (<U1E14>, <U1E15>); (<U1E16>, <U1E17>); (<U1E18>, <U1E19>); /  
 1612 (<U1E1A>, <U1E1B>); (<U1E1C>, <U1E1D>); (<U1E1E>, <U1E1F>); (<U1E20>, <U1E21>); /  
 1613 (<U1E22>, <U1E23>); (<U1E24>, <U1E25>); (<U1E26>, <U1E27>); (<U1E28>, <U1E29>); /  
 1614 (<U1E2A>, <U1E2B>); (<U1E2C>, <U1E2D>); (<U1E2E>, <U1E2F>); (<U1E30>, <U1E31>); /  
 1615 (<U1E32>, <U1E33>); (<U1E34>, <U1E35>); (<U1E36>, <U1E37>); (<U1E38>, <U1E39>); /  
 1616 (<U1E3A>, <U1E3B>); (<U1E3C>, <U1E3D>); (<U1E3E>, <U1E3F>); (<U1E40>, <U1E41>); /  
 1617 (<U1E42>, <U1E43>); (<U1E44>, <U1E45>); (<U1E46>, <U1E47>); (<U1E48>, <U1E49>); /  
 1618 (<U1E4A>, <U1E4B>); (<U1E4C>, <U1E4D>); (<U1E4E>, <U1E4F>); (<U1E50>, <U1E51>); /  
 1619 (<U1E52>, <U1E53>); (<U1E54>, <U1E55>); (<U1E56>, <U1E57>); (<U1E58>, <U1E59>); /  
 1620 (<U1E5A>, <U1E5B>); (<U1E5C>, <U1E5D>); (<U1E5E>, <U1E5F>); (<U1E60>, <U1E61>); /  
 1621 (<U1E62>, <U1E63>); (<U1E64>, <U1E65>); (<U1E66>, <U1E67>); (<U1E68>, <U1E69>); /  
 1622 (<U1E6A>, <U1E6B>); (<U1E6C>, <U1E6D>); (<U1E6E>, <U1E6F>); (<U1E70>, <U1E71>); /  
 1623 (<U1E72>, <U1E73>); (<U1E74>, <U1E75>); (<U1E76>, <U1E77>); (<U1E78>, <U1E79>); /  
 1624 (<U1E7A>, <U1E7B>); (<U1E7C>, <U1E7D>); (<U1E7E>, <U1E7F>); (<U1E80>, <U1E81>); /  
 1625 (<U1E82>, <U1E83>); (<U1E84>, <U1E85>); (<U1E86>, <U1E87>); (<U1E88>, <U1E89>); /

```

1626 (<U1E8A>, <U1E8B>); (<U1E8C>, <U1E8D>); (<U1E8E>, <U1E8F>); (<U1E90>, <U1E91>); /
1627 (<U1E92>, <U1E93>); (<U1E94>, <U1E95>); (<U1EA0>, <U1EA1>); (<U1EA2>, <U1EA3>); /
1628 (<U1EA4>, <U1EA5>); (<U1EA6>, <U1EA7>); (<U1EA8>, <U1EA9>); (<U1EAA>, <U1EAB>); /
1629 (<U1EAC>, <U1EAD>); (<U1EAE>, <U1EAF>); (<U1EB0>, <U1EB1>); (<U1EB2>, <U1EB3>); /
1630 (<U1EB4>, <U1EB5>); (<U1EB6>, <U1EB7>); (<U1EB8>, <U1EB9>); (<U1EBA>, <U1EBB>); /
1631 (<U1EBC>, <U1EBD>); (<U1EBE>, <U1EBF>); (<U1EC0>, <U1EC1>); (<U1EC2>, <U1EC3>); /
1632 (<U1EC4>, <U1EC5>); (<U1EC6>, <U1EC7>); (<U1EC8>, <U1EC9>); (<U1ECA>, <U1ECB>); /
1633 (<U1ECC>, <U1ECD>); (<U1ECE>, <U1ECF>); (<U1ED0>, <U1ED1>); (<U1ED2>, <U1ED3>); /
1634 (<U1ED4>, <U1ED5>); (<U1ED6>, <U1ED7>); (<U1ED8>, <U1ED9>); (<U1EDA>, <U1EDB>); /
1635 (<U1EDC>, <U1EDD>); (<U1EDE>, <U1EDF>); (<U1EE0>, <U1EE1>); (<U1EE2>, <U1EE3>); /
1636 (<U1EE4>, <U1EE5>); (<U1EE6>, <U1EE7>); (<U1EE8>, <U1EE9>); (<U1EEA>, <U1EEB>); /
1637 (<U1EEC>, <U1EEF>); (<U1EEF>, <U1EEF>); (<U1EF0>, <U1EF1>); (<U1EF2>, <U1EF3>); /
1638 (<U1EF4>, <U1EF5>); (<U1EF6>, <U1EF7>); (<U1EF8>, <U1EF9>); (<U1F08>, <U1F09>); /
1639 (<U1F0A>, <U1F0B>); (<U1F0C>, <U1F0D>); (<U1F0E>, <U1F0F>); (<U1F10>, <U1F11>); /
1640 (<U1F12>, <U1F13>); (<U1F14>, <U1F15>); (<U1F16>, <U1F17>); (<U1F18>, <U1F19>); /
1641 (<U1F1A>, <U1F1B>); (<U1F1C>, <U1F1D>); (<U1F1E>, <U1F1F>); (<U1F20>, <U1F21>); /
1642 (<U1F22>, <U1F23>); (<U1F24>, <U1F25>); (<U1F26>, <U1F27>); (<U1F28>, <U1F29>); /
1643 (<U1F2A>, <U1F2B>); (<U1F2C>, <U1F2D>); (<U1F2E>, <U1F2F>); (<U1F30>, <U1F31>); /
1644 (<U1F32>, <U1F33>); (<U1F34>, <U1F35>); (<U1F36>, <U1F37>); (<U1F38>, <U1F39>); /
1645 (<U1F3A>, <U1F3B>); (<U1F3C>, <U1F3D>); (<U1F3E>, <U1F3F>); (<U1F40>, <U1F41>); /
1646 (<U1F42>, <U1F43>); (<U1F44>, <U1F45>); (<U1F46>, <U1F47>); (<U1F48>, <U1F49>); /
1647 (<U1F4A>, <U1F4B>); (<U1F4C>, <U1F4D>); (<U1F4E>, <U1F4F>); (<U1F50>, <U1F51>); /
1648 (<U1F52>, <U1F53>); (<U1F54>, <U1F55>); (<U1F56>, <U1F57>); (<U1F58>, <U1F59>); /
1649 (<U1F5A>, <U1F5B>); (<U1F5C>, <U1F5D>); (<U1F5E>, <U1F5F>); (<U1F60>, <U1F61>); /
1650 (<U1F62>, <U1F63>); (<U1F64>, <U1F65>); (<U1F66>, <U1F67>); (<U1F68>, <U1F69>); /
1651 (<U1F6A>, <U1F6B>); (<U1F6C>, <U1F6D>); (<U1F6E>, <U1F6F>); (<U1F70>, <U1F71>); /
1652 (<U1F72>, <U1F73>); (<U1F74>, <U1F75>); (<U1F76>, <U1F77>); (<U1F78>, <U1F79>); /
1653 (<U1F7A>, <U1F7B>); (<U1F7C>, <U1F7D>); (<U1F7E>, <U1F7F>); (<U1F80>, <U1F81>); /
1654 (<U1F82>, <U1F83>); (<U1F84>, <U1F85>); (<U1F86>, <U1F87>); (<U1F88>, <U1F89>); /
1655 (<U1F8A>, <U1F8B>); (<U1F8C>, <U1F8D>); (<U1F8E>, <U1F8F>); (<U1F90>, <U1F91>); /
1656 (<U1F92>, <U1F93>); (<U1F94>, <U1F95>); (<U1F96>, <U1F97>); (<U1F98>, <U1F99>); /
1657 (<U1F9A>, <U1F9B>); (<U1F9C>, <U1F9D>); (<U1F9E>, <U1F9F>); (<U1FA0>, <U1FA1>); /
1658 (<U1FA2>, <U1FA3>); (<U1FA4>, <U1FA5>); (<U1FA6>, <U1FA7>); (<U1FA8>, <U1FA9>); /
1659 (<U1FAA>, <U1FAB>); (<U1FAC>, <U1FAD>); (<U1FAE>, <U1FAF>); (<U1FB0>, <U1FB1>); /
1660 (<U1FB2>, <U1FB3>); (<U1FB4>, <U1FB5>); (<U1FB6>, <U1FB7>); (<U1FB8>, <U1FB9>); /
1661 (<U1FBA>, <U1FBB>); (<U1FBC>, <U1FBD>); (<U1FBE>, <U1FBF>); (<U1FC0>, <U1FC1>); /
1662 (<U1FC2>, <U1FC3>); (<U1FC4>, <U1FC5>); (<U1FC6>, <U1FC7>); (<U1FC8>, <U1FC9>); /
1663 (<U1FCA>, <U1FCB>); (<U1FCC>, <U1FCD>); (<U1FCE>, <U1FCE>); (<U1FCE>, <U1FCE>); /
1664 %
1665 % The "combining" class reflects ISO/IEC 10646-1 annex B.1
1666 % That is, all combining characters (level 2+3).
1667 class "combining" /
1668 <U0300>..<U036F>; <U20D0>..<U20FF>; <UFE20>..<UFE2F>; /
1669 <U0483>..<U0486>; <U0591>..<U05A1>; <U05A3>..<U05B9>; /
1670 <U05BB>..<U05BD>; <U05BF>; <U05C1>; <U05C2>; <U05C4>; <U064B>..<U0652>; <U0670>; /
1671 <U06D7>..<U06E4>; <U06E7>; <U06E8>; <U06EA>..<U06ED>; <U0901>..<U0903>; <U093C>; /
1672 <U093E>..<U094D>; <U0951>..<U0954>; <U0962>; <U0963>; <U0981>..<U0983>; <U09BC>; /
1673 <U09BE>..<U09C4>; <U09C7>; <U09C8>; <U09CB>..<U09CD>; <U09D7>; <U09E2>; <U09E3>; /
1674 <U0A02>; <U0A3C>; <U0A3E>..<U0A42>; <U0A47>; <U0A48>; <U0A4B>..<U0A4D>; /
1675 <U0A70>; <U0A71>; <U0A81>..<U0A83>; <U0ABC>; <U0ABE>..<U0AC5>; <U0AC7>..<U0AC9>; /
1676 <U0ACB>..<U0ACD>; <U0B01>..<U0B03>; <U0B3C>; <U0B3E>..<U0B43>; <U0B47>; <U0B48>; /
1677 <U0B4B>..<U0B4D>; <U0B56>; <U0B57>; <U0B82>; <U0B83>; <U0BBE>..<U0BC2>; /
1678 <U0BC6>..<U0BC8>; <U0BCA>..<U0BCD>; <U0BD7>; <U0C01>..<U0C03>; <U0C3E>..<U0C44>; /
1679 <U0C46>..<U0C48>; <U0C4A>..<U0C4D>; <U0C55>; <U0C56>; <U0C82>; <U0C83>; /
1680 <U0CBE>..<U0CC4>; <U0CC6>..<U0CC8>; <U0CCA>..<U0CCD>; <U0CD5>; <U0CD6>; /
1681 <U0D02>; <U0D03>; <U0D3E>..<U0D43>; <U0D46>..<U0D48>; <U0D4A>..<U0D4D>; <U0D57>; /
1682 <U0E31>; <U0E34>..<U0E3A>; <U0E47>..<U0E4E>; <U0EB1>; <U0EB4>..<U0EB9>; /
1683 <U0EBB>; <U0EBC>; <U0EC8>..<U0ECD>; <U0F18>; <U0F19>; <U0F35>; <U0F37>; <U0F39>; /
1684 <U0F3E>; <U0F3F>; <U0F71>..<U0F84>; <U0F86>..<U0F89>; <U0F8B>; <U0F90>..<U0F95>; /
1685 <U0F97>; <U0F99>..<U0FAD>; <U0FB1>..<U0FB7>; <U0FB9>; <U302A>..<U302F>; /
1686 <U3099>; <U309A>; <UFB1E>
1687 %
1688 % The "combining_level3" class reflects ISO/IEC 10646-1 annex B.2
1689 % That is, combining characters of level 3.
1690 class "combining_level3"; /
1691 <U0300>..<U036F>; <U20D0>..<U20FF>; <U1100>..<U11FF>; <UFE20>..<UFE2F>; /
1692 <U0483>..<U0486>; <U0591>..<U05A1>; <U05A3>..<U05AE>; <U05C4>; /
1693 <U05AF>; <U093C>; <U0953>; <U0954>; <U09BC>; <U09D7>; <U0A3C>; /
1694 <U0A70>; <U0A71>; <U0ABC>; <U0B3C>; <U0B56>; <U0B57>; <U0BD7>; <U0C55>; <U0C56>; /
1695 <U0CD5>; <U0CD6>; <U0D57>; <U0F39>; <U302A>..<U302F>; <U3099>; <U309A>
1696 %
1697 width /
1698 <U200B>; <U200C>; <U200D>; <U200E>; <U200F>; <U202A>; <U202B>; /
1699 <U202C>; <U202D>; <U202E>; <UFEFF> : 0; /
1700 <U1100>..<U115F>; <U2E80>..<U3009>; <U300C>..<U3019>; /
1701 <U301C>..<U303E>; <U3040>..<UA4CF>; <UAC00>..<UD7A3>; /
1702 <UF900>..<UFAFF>; <UFE30>..<UFE6F>; <UFF00>..<UFF5F>; /
1703 <UFFE0>..<UFFE6> : 2
1704 END LC_CTYPE

```

1704  
 1705  
 1706  
 1707  
 1708  
 1709  
 1710  
 1711  
 1712  
 1713  
 1714  
 1715  
 1716  
 1717  
 1718  
 1719  
 1720  
 1721  
 1722  
 1723  
 1724  
 1725  
 1726  
 1727  
 1728  
 1729  
 1730  
 1731  
 1732  
 1733  
 1734  
 1735  
 1736  
 1737  
 1738  
 1739  
 1740  
 1741  
 1742  
 1743  
 1744  
 1745  
 1746  
 1747  
 1748  
 1749  
 1750  
 1751  
 1752  
 1753  
 1754  
 1755

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

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

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

**copy** Specify the name of an existing FDCC-set to be used as the source for the definition of this category. If

1756		this keyword is specified, only the "reorder-after",
1757		"reorder-end", "reorder-section-after" and "reorder-
1758		section-end" keywords may also be specified. The
1759		FDCC-set is copied in source form.
1760	<b>coll_weight_max</b>	Define as a decimal number the number of collation
1761		levels that an interpreting system needs to support
1762		for this FDCC-set, this value is elsewhere referred to
1763		as the COLL_WEIGHT_MAX limit (e.g. in the
1764		"order_start" statement). An interpreting system
1765		caters for up to 7 collating levels.
1766	<b>section-symbol</b>	Define a section symbol representing a set of
1767		collation order statements. The section is defined
1768		with the "order_start" keyword until the next
1769		"order_start" or "order_end" keyword. This keyword
1770		is optional.
1771	<b>collating-element</b>	Define a collating-element symbol representing a
1772		multicharacter collating element. This keyword is
1773		optional.
1774	<b>collating-symbol</b>	Define one or more collating symbols for use in
1775		collation order statements. This keyword is optional.
1776	<b>symbol-equivalence</b>	Define a collating-symbol to be equivalent to another
1777		defined collating-symbol.
1778	<b>order_start</b>	Define collation rules. This statement is followed by
1779		one or more collation order statements, assigning
1780		character collation values and collation weights to
1781		collating elements.
1782	<b>order_end</b>	Specify the end of the collation-order statements.
1783	<b>reorder-after</b>	Redefine collating rules. Specify after which
1784		collating element the redefinition of collation order
1785		takes order. This statement is followed by one or
1786		more collation order statements, reassigning character
1787		collation values and collation weights to collating
1788		elements.
1789	<b>reorder-end</b>	Specify the end of the "reorder-after" collating order
1790		statements.
1791	<b>reorder-section-after</b>	Redefine the order of sections. This statement is
1792		followed by one or more section symbols,
1793		reassigning character collation values and collation
1794		weights to collating elements.
1795	<b>reorder-section-end</b>	Specify the end of the "reorder-section" section order
1796		statements.

#### 4.4.1 Collation statements

The "order\_start" and "reorder-after" 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



1808 "UNDEFINED". The weights for each of the collation elements determines the character  
1809 collation sequence - such that each collation statement does not need to be in collation  
1810 order, and weights could be rearranged via for example the "reorder-after" keyword. No  
1811 character has any specific predetermined placement in the collation sequence. The order in  
1812 which collating elements are specified determines the character collation sequence, such  
1813 that each collating element compares less than the elements following it.

1814  
1815 A <collating-element> is used to specify multicharacter collating elements, and indicates  
1816 that the character sequence specified via the <collating-element> is to be collated as a unit  
1817 and in the relative order specified by its place in the list of collating statements.

1818  
1819 A <collating-symbol> is used to define a position in the relative order for use in weights.

1820  
1821 The absolute ellipsis symbol ("...") specifies that a sequence of characters collate according  
1822 to their encoded character values. It is interpreted as indicating that all characters with a  
1823 coded character set value higher than the value of the character in the preceding line, and  
1824 lower than the coded character set value for the character in the following line, in the  
1825 current coded character set, are placed in the character collation order between the  
1826 previous and the following character in ascending order according to their coded character  
1827 set values. An initial ellipsis is interpreted as if the preceding line specified the <NUL>  
1828 character, and a trailing ellipsis as if the following line specified the highest coded  
1829 character set value in the current coded character set. An ellipsis is treated as invalid if the  
1830 preceding or following lines do not specify characters in the current coded character set.  
1831 The use of the ellipsis symbol ties the definition to a specific coded character set and may  
1832 preclude the definition from being portable between applications, and is depreciated.  
1833 Symbolic ellipses may be used as the ellipses symbol, but generating symbolic character  
1834 names, and thus have a better chance of portability between applications.

1835  
1836 The symbolic ellipses (".." or "....") specifies a sequence of collating statements. It is  
1837 interpreted as indicating that all characters with symbolic names higher than the symbolic  
1838 name of the character in the preceding line, and lower in the sequence of symbolic names  
1839 for the character in the following line, is placed in the character collation order between  
1840 the previous and the following character in ascending order.

1841  
1842 The symbol "UNDEFINED" is interpreted as including all coded character set values not  
1843 specified explicitly or via the ellipsis or one of the symbolic ellipses symbols. Such  
1844 characters are inserted in the character collation order at the point indicated by the symbol,  
1845 and in ascending order according to their coded character set values. If no "UNDEFINED"  
1846 symbol is specified, and the current coded character set contains characters not specified  
1847 in this clause, the utility issues a warning message and place such characters at the end of  
1848 the character collation order.

1849  
1850 The optional operands for each collation-element are used to define the primary,  
1851 secondary, or subsequent weights for the collating element. The first operand specifies the  
1852 relative primary weight, the second the relative secondary weight, and so on. Two or more  
1853 collation-elements can be assigned the same weight; they belong to the same equivalence  
1854 class if they have the same primary weight. Collation behaves as if, for each weight level,  
1855 "IGNORE"d elements are removed. Then each successive pair of elements is compared  
1856 according to the relative weights for the elements. If the two strings compare equal, the  
1857 process is repeated for the next weight level, up to the limit "COLL\_WEIGHTS\_MAX" of  
1858 the associated FDCC-set.

1859

1860 Weights are expressed as characters (in any of the forms specified here), <collating-  
 1861 symbol>s, <collating-element>s, an ellipsis, or the special symbol "IGNORE". A single  
 1862 character, a <collating-symbol>, or a <collating-element> represent the relative order in  
 1863 the character collating sequence of the character or symbol, rather than the character or  
 1864 characters themselves.

1865  
 1866 One-to-many mapping is indicated by specifying two or more concatenated characters or  
 1867 symbolic names. Thus, if the character <ss> is given the string <s><s> as a weight,  
 1868 comparisons are performed as if all occurrences of the character <ss> are replaced by  
 1869 <s><s>. If it is desirable to define <ss> and <s><s> as an equivalence class, then a  
 1870 collating-element must be defined for the string "ss", as in the example below.

1871  
 1872 All characters specified via an ellipsis are by default assigned unique weights, equal to the  
 1873 relative order of characters. Characters specified via an explicit or implicit "UNDEFINED"  
 1874 special symbol are by default assigned the same primary weight (i.e., belong to the same  
 1875 equivalence class). An ellipsis symbol as a weight is interpreted to mean that each  
 1876 character in the sequence has unique weights, equal to the relative order of their character  
 1877 in the character collation sequence. Secondary and subsequent weights have unique values.  
 1878 The use of the ellipsis as a weight is treated as an error if the collating element is neither  
 1879 an ellipsis nor the special symbol "UNDEFINED".

1880  
 1881 The special keyword "IGNORE" as a weight indicates that when strings are compared  
 1882 using the weights at the level where "IGNORE" is specified, the collating element is  
 1883 ignored; i.e., as if the string did not contain the collating element. In regular expressions  
 1884 and pattern matching, all characters that are "IGNORE"d in their primary weight form an  
 1885 equivalence class.

1886  
 1887 A <comment\_character> occurring where the delimiter ";" may occur, terminates the  
 1888 collating statement.

1889  
 1890 An empty operand is interpreted as the collating-element itself.

1891  
 1892 For example, the collation statement

```
1893     <a>  <a>;<a>
```

1894  
 1895  
 1896 is equal to

```
1897     <a>
```

1898  
 1899  
 1900 An ellipsis (absolute or symbolic) can be used as an operand if the collating-element was  
 1901 an ellipsis, and is interpreted as the value of each character defined by the ellipsis.

1902 Example:

```
1903 collating-element <ch> from "<c><h>"
1904 collating-element <Ch> from "<C><h>"
1905 order_start      forward;backward
1906 UNDEFINED       IGNORE;IGNORE
1907 <LOW>
1908 <space>         <LOW>;<space>
1909 ...            <LOW>;
1910 <a>             <a>;<a>
1911 <a'>           <a>;<a'>
1912 <A>            <a>;<A>
1913 <A'>          <a>;<A'>
1914 <ch>           <ch>;<ch>
1915 <Ch>           <ch>;<Ch>
```

```

1918 <s>           <s> ; <s>
1919 <ss>         " <s><s> " ; " <ss><ss> "
1920 order_end

```

This example is interpreted as follows:

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

#### 4.4.2 "copy" keyword

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

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

The <FDCC-set-name> consists of one or more characters (in any of the forms defined in 4.1.1). If this keyword is specified, only the "reorder-after", "reorder-end", "reorder-section-after" and "reorder-section-end" keywords may also be specified. The FDCC-set is copied in source form.

#### 4.4.3 "coll\_weight\_max" keyword

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

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

#### 4.4.4 "section-symbol" keyword

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

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

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

```

Example:
section-symbol <LATIN>
section-symbol <ARABIC>

```

#### 1975 4.4.5 "collating-element" keyword

1976  
1977 In addition to the collating elements in the character set, the collating-element keyword is  
1978 used to define multicharacter collating elements. The syntax is

1979  
1980 "collating-element %s from %s\n", <collating-symbol>, <string>

1981  
1982 The <collating-symbol> operand is a symbolic name, enclosed between angle brackets (<  
1983 and >), and does not duplicate any symbolic name in the current charmap or repertoire  
1984 file (if any), or any other symbolic name defined in this collation definition. The string  
1985 operand is a string of two or more characters that collates as an entity. A <collating-  
1986 element> defined via this keyword is only defined within the LC\_COLLATE category.

1987  
1988 Example with ISO/IEC 10646-1:  
1989 collating-element <ch> from "<c><h>"  
1990 collating-element <e-acute> from "<e><combining-acute>"  
1991 collating-element <aa> from "<a><a>"

1992  
1993 Note: The problem of comparing a fully composed character of ISO/IEC 10646 with a decomposed  
1994 representation of the same text is sometimes handled by the two strings comparing equal up to level 3 (the  
1995 case level) of ISO/IEC 14651, but distinguishing the two at the 4th level.

#### 1996 4.4.6 "collating-symbol" keyword

1997  
1998 This keyword is used to define symbols for use in collation sequence statements; e.g.,  
1999 between the order\_start and the order\_end keywords. The syntax is

2000  
2001 "collating-symbol %s;%s;...%s\n", <collating-symbol>, <collating-symbol> ...

2002  
2003 The <collating-symbol> is a symbolic name, enclosed between angle brackets (< and >),  
2004 and does not duplicate any symbolic name in the current charmap (if any), or any other  
2005 symbolic name defined in this collation definition. A <collating-symbol> defined via this  
2006 keyword is only defined within the LC\_COLLATE category. More than one <collating-  
2007 symbol> may be defined with one "collating-symbol" keyword, and symbolic ellipses may  
2008 be used.

2009  
2010 Example:  
2011 collating-symbol <CAPITAL>  
2012 collating-symbol <HIGH>

#### 2013 4.4.7 "symbol-equivalence" keyword

2014  
2015 This keyword is used to define symbols for use in collation sequence statements; and  
2016 assign the same weight as another defined symbol. The syntax is

2017  
2018 "symbol-equivalence %s %s\n", <collating-symbol-1>, <collating-symbol-2>

2019  
2020 The <collating-symbol-1> and <collating-symbol-2> are symbolic names, enclosed  
2021 between angle brackets (< and >). <collating-symbol-1> does not duplicate any symbolic  
2022 name in the current charmap (if any), or any other symbolic name defined in this collation  
2023 definition. <collating-symbol-2> is defined elsewhere in the LC\_COLLATE category as a  
2024 collating-symbol. The use of <collating-symbol-2> is equivalent to using the <collating-  
2025 symbol-1> in the LC\_COLLATE category. A <collating-symbol-1> defined via this  
2026 keyword is only defined within the LC\_COLLATE category.

2029                   Example  
 2030                   collating-symbol <CAP>  
 2031                   symbol-equivalence <CAPITAL> <CAP>

#### 2032 2033 **4.4.8 "order\_start" keyword**

2034  
2035 The "order\_start" keyword precedes collation order entries and also defines the number of  
 2036 weights for this collation sequence definition, the collation section name and other  
 2037 collation rules.

2038  
2039 The syntax of the "order\_start" keyword has two forms:

2040                   "order\_start %s;%s;...;%s\n", <sort-rule>, <sort-rule> ...

2041 and

2042                   "order\_start %s;%s;...;%s\n", <section-symbol>, <sort-rules>, <sort-rules> ...

2043  
2044  
2045 The operands to the order\_start keyword are optional. If present, the operands define rules  
 2046 to be applied when strings are compared. The first operand may be a <section-symbol>  
 2047 surrounded by "<" and ">" and the set of collating statements following the "order\_start"  
 2048 keyword until the "order\_end" keyword are identified with this <section-symbol> or  
 2049 another "order\_start" keyword is encountered. The remaining number of operands define  
 2050 how many weights each element is assigned; if no operands are present, one forward  
 2051 operand is assumed. If present, the first operand defines rules to be applied when  
 2052 comparing strings using the first (primary) weight; the second when comparing strings  
 2053 using the second weight, and so on. Operands are separated by semicolons (;). Each  
 2054 operand consists of one or more collation directives, separated by commas (.). If the  
 2055 number of operands exceeds the (COLL\_WEIGHTS\_MAX) limit, a utility parsing the  
 2056 FDCC-set description issues a warning message. The following directives are supported:

- 2057  
2058 **forward**   Specifies that the direction of scanning a part of a string at a given point in a  
 2059 string is done towards the logical end of the whole string for this weight level.  
 2060 **backward** Specifies that the direction of scanning a part of a string at a given point in a  
 2061 string is done towards the logical beginning of the whole string for this weight  
 2062 level.  
 2063 **position**   Specifies that comparison operations for the weight level will consider the  
 2064 relative position of non-"IGNORE"d elements in the strings. The string  
 2065 containing a non-"IGNORE"d element after the fewest IGNOREd collating  
 2066 elements from the start of the compare collates first. If both strings contain a  
 2067 non-"IGNORE"d character in the same relative position, the collating values  
 2068 assigned to the elements determine the ordering. In case of equality,  
 2069 subsequent non-IGNOREd characters are considered in the same manner.

2070  
2071 The directives "forward" and "backward" are mutually exclusive at a given level. The  
 2072 directives "backward" and "position" are mutually exclusive at a given level.

2073                   Examples:  
 2074                   order\_start forward;backward  
 2075                   order\_start <CYRILLIC>;forward;forward

2076  
2077  
2078 If no operands are specified, a single forward operand is assumed.

#### 2082 4.4.9 "order\_end" keyword

2083  
2084 The collating order entries are terminated with an "order\_end" keyword.

#### 2086 4.4.10 "reorder-after" keyword

2087  
2088 The "reorder-after" keyword is used to specify a modification to a copied collation  
2089 specification of an existing FDCC-set. There can be more than one "reorder-after"  
2090 statement in a collating specification. The syntax is:

```
2091  
2092     "reorder-after %s\n",<collating-symbol>
```

2093  
2094 The <collating-symbol> operand is a symbolic name, enclosed between angle brackets,  
2095 and is present in the source FDCC-set copied via the "copy" keyword.  
2096 The "reorder-after" statement is followed by one or more collation statements as described  
2097 in the "Collating Order" clause (4.4.5), with the exception that the ellipsis symbol (...) is  
2098 not used.

2099  
2100 Each collation statement reassigns character collation values and collation weights to  
2101 collating elements existing in the copied collation specification, by removing the collating  
2102 statement from the copied specification, and inserting the collating element in the collating  
2103 sequence with the new collation weights after the preceding collating element of the  
2104 "reorder-after" specification, the first collating element in the collation sequence being the  
2105 <collating-symbol> specified in the "reorder-after" statement.

2106  
2107 A "reorder-after" specification is terminated by another "reorder-after" specification or the  
2108 "reorder-end" statement.

##### 2109 2110 2111 2112 4.4.10.1 Example of "reorder-after"

```
2113  
2114     reorder-after <y8>  
2115     <U:>         <Y>;<U:>;<CAPITAL>  
2116     <u:>         <Y>;<U:>;<SMALL>  
2117     reorder-after <z8>  
2118     <AE>         <AE>;<NONE>;<CAPITAL>  
2119     <ae>         <AE>;<NONE>;<SMALL>  
2120     <A:>         <AE>;<DIAERESIS>;<CAPITAL>  
2121     <a:>         <AE>;<DIAERESIS>;<SMALL>  
2122     <O/>         <O/>;<NONE>;<CAPITAL>  
2123     <o/>         <O/>;<NONE>;<SMALL>  
2124     <AA>         <AA>;<NONE>;<CAPITAL>  
2125     <aa>         <AA>;<NONE>;<SMALL>  
2126     reorder-end
```

2127  
2128 The example is interpreted as follows (using the "i18nrep" repertoire map):

- 2129  
2130 1. The collating element <U:> is removed from the copied collating sequence and inserted after <y8> in the  
2131 collating sequence with the new weights. The collating element <u:> is removed from the copied collating  
2132 sequence and inserted in the resulting collation sequence after <U:> with the new weights. <y8> is used to  
2133 indicate the last entry of the <y> letters.
- 2134  
2135 2. The second "reorder-after" statement terminates the first list of reordering collation identifier entries, and  
2136 initiates a second list, rearranging the order and weights for the <AE>, <ae>, <A:>, <a:>, <O/>, and <o/>  
2137 collating elements after the <z8> collating symbol in the copied specification. <z8> is used to indicate the  
2138 last entry of the <z> letters.
- 2139  
2140 3. The "reorder-end" statement terminates the second list of reordering entries.

- 2141 4. Thus for the original sequence  
 2142 ... ( U u Ü ü ) V v W w X x Y y Z z

2144 this example reordering gives

2146 ... U u V v W w X x ( Y y Ü ü ) Z z ( Æ æ Ä ä ) Ø ø Å å

2148 where the parenthesis indicate ordering with the same weight on the first level for multiple upper/lowercase  
 2149 pairs.  
 2150

#### 2151 4.4.11 "reorder-end" keyword

2152 The "reorder-end" keyword specifies the end of a list of collating statements, initiated by  
 2153 the "reorder-after" keyword.

#### 2154 4.4.12 "reorder-section-after" keyword

2155 The "reorder-section-after" keyword is used to specify a modification to a copied collation  
 2156 specification of an existing FDCC-set. The "reorder-section-after" statement is followed by  
 2157 one or more statements consisting of section reordering statements.

##### 2158 4.4.12.1 section reordering statements

2159 The section reordering statements rearranges the set of collating entries and changes  
 2160 sorting rules for the set of collating entries identified by a section symbol in a preceding  
 2161 "order\_start" statement. Each section reorder statement has the syntax:

```
2162 "%s %s;...%s\n", <section-symbol>, <sort-rule>, <sort-rule> ...
```

2163 The <section-symbol> identifies the set of collating entries, and is defined via a "section-  
 2164 symbol" keyword.

2165 The <sort-rule>s are as described for the "order\_start" keyword. Specified <sort-rule>s  
 2166 replace the specification for the ordering of the section given on the "order\_start"  
 2167 statement identified by the <section-symbol>. The <sort-rule>s are optional, and <sort-  
 2168 rule>s not to be changed may be given by empty specifications.

2169 Note: The <sort-rule> capability is an extension over ISO/IEC 14651 functionality.

2170 The order of the section reordering statements rearranges the assignment of collation  
 2171 entries for the sets of collation entries identified by the <section-symbols> to the order  
 2172 that the <section-symbols> occur after the "reorder-section-after" statement.

2173 The section reordering statements are terminated by a "reorder-section-end" statement.

##### 2174 4.4.12.2 Example of section reordering

```
2175 copy "i18n"  

  2176 reorder-section-after <DIGITS>  

  2177 <ARABIC>  

  2178 <LATIN> forward;backward;forward;forward,position  

  2179 reorder-section-end
```

2180 This example is interpreted as follows: The LC\_COLLATE category of the "i18n" FDCC-set is copied. Then a  
 2181 reordering of all collating statements for the sections <ARABIC> and <LATIN> is done, leaving the rest of the  
 2182 sections as they were in the "i18n" FDCC-set. The <ARABIC> section is placed immediately after the <DIGITS>

2198 section, and the <LATIN> section immediately following the <ARABIC> section. The ordering rules are kept as  
 2199 they were in the "i18n" FDCC-set, while the <LATIN> section gets new ordering rules as indicated. The  
 2200 "reorder-section-end" keyword terminates the section reordering statements.

2201

#### 2202 **4.4.13 "reorder-section-end" keyword**

2203

2204 The "reorder-section-end" keyword specifies the end of a list of section symbols, initiated  
 2205 by the "reorder-section-after" keyword.

2206

#### 2207 **4.4.14 "i18n" LC\_COLLATE category**

2208

2209 The "i18n" LC\_COLLATE category is defined as the following, which includes the  
 2210 tailorable template in ISO/IEC 14651.

2211

2212

2213

2214

2215

2216

2217

2218

2219

2220

2221

2222

2223

2224

2225

2226

2227

2228

2229

2230

2231

2232

2233

2234

2235

2236

2237

2238

2239

2240

2241

2242

2243

2244

2245

2246

2247

2248

2249

2250

2251

2252

2253

2254

2255

2256

2257

2258

```
LC_COLLATE
```

```
% This is the ISO/IEC TR 14652 i18n fdcc-set definition for  
% the LC_COLLATE category.
```

```
%
```

```
% equivalences
```

```
symbol-equivalence <NONE> <BLANK>
```

```
symbol-equivalence <CAPITAL> <CAP>
```

```
symbol-equivalence <SMALL> <MIN>
```

```
symbol-equivalence <CAPITAL-SMALL> <COMPATCAP>
```

```
symbol-equivalence <SMALL-CAPITAL> <COMPAT>
```

```
symbol-equivalence <MACRON> <MACRO>
```

```
symbol-equivalence <STROKE> <OBLIK>
```

```
symbol-equivalence <ACUTE> <AIGUT>
```

```
symbol-equivalence <CIRCUMFLEX> <CIRCF>
```

```
symbol-equivalence <RING> <CRCL>
```

```
symbol-equivalence <DIAERESIS> <TREMA>
```

```
symbol-equivalence <DOT> <POINT>
```

```
symbol-equivalence <CEDILLA> <CEDIL>
```

```
symbol-equivalence <OGONEK> <OGONK>
```

```
symbol-equivalence <HOOK> <CROOK>
```

```
symbol-equivalence <HORN> <HORNU>
```

```
symbol-equivalence <DOT-BELOW> <POINS>
```

```
order_start forward;forward;forward;forward,position
```

```
% Copy the template from ISO/IEC 14651
```

```
copy "ISO14651_2000_TABLE1.txt"
```

```
order_end
```

```
END LC_COLLATE
```

## 2245 **4.5 LC\_MONETARY**

2246

2247 The LC\_MONETARY category defines the rules and symbols that are used to format  
 2248 monetary numeric information. The operands are strings. For some keywords, the strings  
 2249 can contain only integers. More than one set of monetary values may be provided, and for  
 2250 each set a period of validity and conversion rate may be given. Keywords that are not  
 2251 provided, string values set to the empty string "", or integer keywords set to -1, are used  
 2252 to indicate that the value is unspecified, and then no default is implied. The following  
 2253 keywords are defined:

2254

2255

2256

2257

2258

**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.

**valid\_from** One or more strings separated by semicolons, representing a



2259		Gregorian date in the form "YYYYMMDD" according to
2260		ISO 8601, specifying the beginning date (inclusive from the
2261		beginning of day local time) of the validity of a currency.
2262		The position of the string in the list corresponds to the
2263		position of operands in other keywords in the
2264		LC_MONETARY category. The currencies should be
2265		ordered in terms of validity dates, and for each validity
2266		period with the currency that the amounts are stored in first.
2267		If not specified, it is taken to be an implementation-defined
2268		beginning of time. This keyword is optional.
2269	<b>valid_to</b>	One or more strings separated by semicolons, representing a
2270		Gregorian date in the form "YYYYMMDD" according to
2271		ISO 8601, specifying the end date (inclusive to the end of
2272		day local time) of the validity of a currency. If not specified,
2273		it is taken to be an implementation-defined end of time. This
2274		keyword is optional.
2275	<b>conversion_rate</b>	one or more pairs of integers separated by a <semicolon>
2276		specifying the fixed conversion rate between the current
2277		currency (determined by the parameter number) and the first
2278		currency that is valid, determined by a date provided by the
2279		application. If the currency is not the first valid currency for
2280		the period in question, the first integer is for multiplying the
2281		first valid currency, and the second for dividing this result to
2282		get the amount in the current currency. The currency to be
2283		the current currency is selected by the application from the
2284		date applicable and the currency number (first, second, third
2285		etc valid currency at that date); and whether domestic or
2286		international formatting is used is also determined by the
2287		application. Each pair of integers are separated by a <slash>.
2288		The default value is "1/100". This keyword is optional.
2289		Note: The two integers are used instead of a floating point
2290		value, to be able to cater for legal requirements on Euro
2291		conversion where a multiplication and division is prescribed,
2292		instead of just one floating point multiplication.
2293	<b>currency_symbol</b>	One or more strings separated by semicolons that are used as
2294		the local currency symbol.
2295	<b>mon_decimal_point</b>	The operand is a string containing the symbol that is used as
2296		the decimal delimiter in monetary formatted quantities. In
2297		contexts where other standards limit the "mon_decim-
2298		al_point" to a single byte, the result of specifying a
2299		multibyte operand is unspecified. The keyword is specified,
2300		unless the "copy" keyword is used.
2301	<b>mon_thousands_sep</b>	The operand is a string containing the symbol that is used as
2302		a separator for groups of digits to the left of the decimal
2303		delimiter in formatted monetary quantities. In contexts where
2304		other standards limit the "mon_thousands_sep" to a single
2305		byte, the result of specifying a multibyte operand is
2306		unspecified. The keyword is specified, unless the "copy"
2307		keyword is used.
2308	<b>mon_grouping</b>	Define the size of each group of digits in formatted
2309		monetary quantities. The operand is a sequence of integers
2310		separated by semicolons. Each integer specifies the number

2311		of digits in each group, with the initial integer defining the
2312		size of the group immediately preceding the decimal
2313		delimiter, and the following integers defining the preceding
2314		groups. If the last integer is not -1, then the size of the
2315		previous group (if any) is repeatedly used for the remainder
2316		of the digits. If the last integer is -1, then no further
2317		grouping is performed. The keyword is specified, unless the
2318		"copy" keyword is used.
2319	<b>positive_sign</b>	A string that is used to indicate a nonnegative-valued
2320		formatted monetary quantity. The keyword is specified,
2321		unless the "copy" keyword is used.
2322	<b>negative_sign</b>	A string that is used to indicate a negative-valued formatted
2323		monetary quantity. The keyword is specified, unless the
2324		"copy" keyword is used.
2325	<b>frac_digits</b>	One or more integers separated by semicolons, representing
2326		the number of fractional digits (those to the right of the
2327		decimal delimiter) to be written in a formatted monetary
2328		quantity using "currency_symbol". The keyword is specified,
2329		unless the "copy" keyword is used.
2330	<b>p_cs_precedes</b>	One or more integers separated by semicolons, set to 1 if the
2331		"currency_symbol" precedes the value for a nonnegative
2332		formatted monetary quantity, and set to 0 if the symbol
2333		succeeds the value. The keyword is specified, unless the
2334		"copy" keyword is used.
2335	<b>p_sep_by_space</b>	One or more integers separated by semicolons, set to 0 if no
2336		space separates the "currency_symbol" from the value for a
2337		nonnegative formatted monetary quantity, set to 1 if a space
2338		separates the symbol from the value, and set to 2 if a space
2339		separates the symbol and the sign string, if adjacent. The
2340		keyword is specified, unless the "copy" keyword is used.
2341	<b>n_cs_precedes</b>	One or more integers separated by semicolons, set to 1 if the
2342		"currency_symbol" precedes the value for a negative
2343		formatted monetary quantity, and set to 0 if the symbol
2344		succeeds the value. The keyword is specified, unless the
2345		"copy" keyword is used.
2346	<b>n_sep_by_space</b>	One or more integers separated by semicolons, set to 0 if no
2347		space separates the "currency_symbol" from the value for a
2348		negative formatted monetary quantity, set to 1 if a space
2349		separates the symbol from the value, and set to 2 if a space
2350		separates the symbol and the sign string, if adjacent. The
2351		keyword is specified, unless the "copy" keyword is used.
2352	<b>p_sign_posn</b>	One or more integers separated by semicolons, set to a value
2353		indicating the positioning of the "positive_sign" for a
2354		nonnegative formatted monetary quantity using the
2355		"currency_symbol". The following integer values are defined:
2356		
2357		0 Parentheses enclose the quantity and the
2358		"currency_symbol".
2359		1 The sign string precedes the quantity and the
2360		"currency_symbol".
2361		2 The sign string succeeds the quantity and the
2362		"currency_symbol".

2363		3	The sign string immediately precedes the
2364			"currency_symbol".
2365		4	The sign string immediately succeeds the
2366			"currency_symbol".
2367			The keyword is specified, unless the "copy" keyword is used.
2368			
2369	<b>n_sign_posn</b>		One or more integers separated by semicolons, set to a value
2370			indicating the positioning of the "negative_sign" for a
2371			negative formatted monetary quantity using the
2372			"currency_symbol". The following integer values are defined:
2373			
2374		0	Parentheses enclose the quantity and the
2375			"currency_symbol".
2376		1	The sign string precedes the quantity and the
2377			"currency_symbol".
2378		2	The sign string succeeds the quantity and the
2379			"currency_symbol".
2380		3	The sign string immediately precedes the
2381			"currency_symbol".
2382		4	The sign string immediately succeeds the
2383			"currency_symbol".
2384			The keyword is specified, unless the "copy" keyword is used.
2385	<b>int_curr_symbol</b>		One or more strings separated by semicolons that are used as
2386			the international currency symbols. Each operand is a four
2387			character string, with the first three characters containing the
2388			alphabetic international currency symbol in accordance with
2389			those specified in ISO 4217, <i>Codes for the representation of</i>
2390			<i>currencies and funds</i> . The fourth character is the character
2391			used to separate the international currency symbol from the
2392			monetary quantity. The keyword is specified, unless the
2393			"copy" keyword is used.
2394	<b>int_frac_digits</b>		One or more integers separated by semicolons, representing
2395			the number of fractional digits (those to the right of the
2396			decimal delimiter) to be written in a formatted monetary
2397			quantity using "int_curr_symbol". The keyword is specified,
2398			unless the "copy" keyword is used.
2399	<b>int_p_cs_precedes</b>		One or more integers separated by semicolons; set to 1 if the
2400			"int_curr_symbol" precedes the value for a nonnegative
2401			formatted monetary quantity, and set to 0 if the symbol
2402			succeeds the value. If not specified, the value of
2403			"p_cs_precedes" is taken.
2404	<b>int_p_sep_by_space</b>		One or more integers separated by semicolons; set to 0 if no
2405			space separates the "int_curr_symbol" from the value for a
2406			nonnegative formatted monetary quantity, set to 1 if a space
2407			separates the symbol from the value, and set to 2 if a space
2408			separates the symbol and the sign string, if adjacent. If not
2409			specified, the value of "p_sep_by_space" is taken.
2410	<b>int_n_cs_precedes</b>		One or more integers separated by semicolons; set to 1 if the
2411			"int_curr_symbol" precedes the value for a negative
2412			formatted monetary quantity, and set to 0 if the symbol
2413			succeeds the value. If not specified, the value of
2414			"n_cs_precedes" is taken.

2415	<b>int_n_sep_by_space</b>	One or more integers separated by semicolons; set to 0 if no
2416		space separates the "int_curr_symbol" from the value for a
2417		negative formatted monetary quantity, set to 1 if a space
2418		separates the symbol from the value, and set to 2 if a space
2419		separates the symbol and the sign string, if adjacent. If not
2420		specified, the value of "n_sep_by_space" is taken.
2421	<b>int_p_sign_posn</b>	One or more integers separated by semicolons, set to a value
2422		indicating the positioning of the "positive_sign" for a
2423		nonnegative formatted monetary quantity using the
2424		"int_curr_symbol". The following integer values are defined:
2425		
2426		0 Parentheses enclose the quantity and the
2427		"int_curr_symbol".
2428		1 The sign string precedes the quantity and the
2429		"int_curr_symbol".
2430		2 The sign string succeeds the quantity and the
2431		"int_curr_symbol".
2432		3 The sign string immediately precedes the
2433		"int_curr_symbol".
2434		4 The sign string immediately succeeds the
2435		"int_curr_symbol".
2436		If no "int_p_sign_posn" is present the value of the
2437		"p_sign_posn" is taken.
2438		
2439	<b>int_n_sign_posn</b>	One or more integers separated by semicolons, set to a value
2440		indicating the positioning of the "negative_sign" for a
2441		negative formatted monetary quantity using the
2442		"int_curr_symbol". The following integer values are defined:
2443		
2444		0 Parentheses enclose the quantity and the
2445		"int_curr_symbol".
2446		1 The sign string precedes the quantity and the
2447		"int_curr_symbol".
2448		2 The sign string succeeds the quantity and the
2449		"int_curr_symbol".
2450		3 The sign string immediately precedes the
2451		"int_curr_symbol".
2452		4 The sign string immediately succeeds the
2453		"int_curr_symbol".
2454		If no "int_n_sign_posn" is present the value of the
2455		"n_sign_posn" is taken.
2456		

The "i18n" FDCC-set is defined as follows for the LC\_MONETARY category.

```

2457 LC_MONETARY
2458 % This is the 14652 i18n fdcc-set definition for
2459 % the LC_MONETARY category.
2460 %
2461 int_curr_symbol      ""
2462 currency_symbol     ""
2463 mon_decimal_point   "<U002C>"
2464 mon_thousands_sep  ""
2465 mon_grouping        -1
2466 positive_sign       ""
2467 negative_sign       "<U002E>"
2468
2469

```

```

2470     int_frac_digits      -1
2471     frac_digits          -1
2472     p_cs_precedes        -1
2473     p_sep_by_space       -1
2474     n_cs_precedes        -1
2475     n_sep_by_space       -1
2476     p_sign_posn         -1
2477     n_sign_posn         -1
2478     %
2479     END LC_MONETARY

```

#### 2482 4.6 LC\_NUMERIC

2483  
 2484 The LC\_NUMERIC category defines the rules and symbols that are used to format  
 2485 nonmonetary numeric information. The operands are strings. For some keywords, the  
 2486 strings only can contain integers. Keywords that are not provided, string values set to the  
 2487 empty string (""), or integer keywords set to -1, are used to indicate that the value is  
 2488 unspecified. The following keywords are defined:

- 2489  
 2490 **copy** Specify the name of an existing FDCC-set to be used as the  
 2491 source for the definition of this category. If this keyword is  
 2492 specified, no other keyword is specified.
- 2493 **decimal\_point** The operand is a string containing the symbol that is used as the  
 2494 decimal delimiter in numeric, nonmonetary formatted quantities.  
 2495 This keyword cannot be omitted and cannot be set to the empty  
 2496 string. In contexts where other standards limit the decimal point  
 2497 to a single byte, the result of specifying a multibyte operand is  
 2498 unspecified.
- 2499 **thousands\_sep** The operand is a string containing the symbol that is used as a  
 2500 separator for groups of digits to the left of the decimal delimiter  
 2501 in numeric, nonmonetary formatted monetary quantities. In  
 2502 contexts where other standards limit the "thousands\_sep" to a  
 2503 single byte, the result of specifying a multibyte operand is  
 2504 unspecified.
- 2505 **grouping** Define the size of each group of digits in formatted non-  
 2506 monetary quantities. The operand is a sequence of integers  
 2507 separated by semicolons. Each integer specifies the number of  
 2508 digits in each group, with the initial integer defining the size of  
 2509 the group immediately preceding the decimal delimiter, and the  
 2510 following integers defining the preceding groups. If the last  
 2511 integer is not -1, then the size of the previous group (if any) is  
 2512 repeatedly used for the remainder of the digits. If the last integer  
 2513 is -1, then no further grouping is performed.

2514  
 2515 The "i18n" FDCC-set is for the LC\_NUMERIC category:

```

2516     LC_NUMERIC
2517     % This is the 14652 i18n fdcc-set definition for
2518     % the LC_NUMERIC category.
2519     %
2520     decimal_point      "<U002C>"
2521     thousands_sep      ""
2522     grouping           -1
2523     %
2524     END LC_NUMERIC

```

2528 **4.7 LC\_TIME**

2529

2530 The LC\_TIME category defines the rules and symbols that are used to format date and  
2531 time information. The following keywords are defined:

2532

2533 **copy** Specify the name of an existing FDCC-set to be used as the source  
2534 for the definition of this category. If this keyword is specified, no  
2535 other keyword is specified.

2536 **abday** Define the abbreviated weekday names for calendar systems with  
2537 weeks of constant length, to be referenced by the %a field descriptor.  
2538 The length of the week and a gregorian date for the first weekday is  
2539 defined by the "week" keyword. The operand consists of semicolon-  
2540 separated strings. The first string is the abbreviated name of the day  
2541 corresponding to the first day of the week (default Sunday), the  
2542 second the abbreviated name of the day corresponding to the second  
2543 day of the week (default Monday), and so on.

2544 **day** Define the full weekday names for calendar systems with weeks of  
2545 constant length, to be referenced by the %A field descriptor. The  
2546 length of the week and a gregorian date for the first weekday is  
2547 defined by the "week" keyword. The operand consists of semicolon-  
2548 separated strings. The first string is the full name of the day  
2549 corresponding to the first day of the week (default Sunday), the  
2550 second the full name of the day corresponding to the second day of  
2551 the week (default Monday), and so on.

2552 **week** Is used to define the number of days in a week, and which weekday  
2553 is the first weekday (the first weekday has the value 1), and which  
2554 week is to be considered the first in a year. The first operand is an  
2555 integer specifying the number of days in the week. The second  
2556 operand is an integer specifying the Gregorian date in the format  
2557 YYYYMMDD, and it specifies a day that is a first weekday (all  
2558 other first weekdays may then be calculated by adding or subtracting  
2559 a whole multiplum of the number of days in the week as specified  
2560 with the first operand). The third operand is an integer specifying the  
2561 weekday number to be contained in the first week of the year. The  
2562 third operand may also be understood as the number of days required  
2563 in a week for it to be considered the first week of the year. If the  
2564 keyword is not specified the values are taken as 7, 19971130 (a  
2565 Sunday), and 7 (Saturday), respectively. ISO 8601 conforming  
2566 applications should use the values 7, 19971201 (a Monday), and 4  
2567 (Thursday), respectively. This keyword is optional.

2568 **abmon** Define the abbreviated month names, to be referenced by the %b  
2569 field descriptor. The operand consists of twelve or thirteen  
2570 semicolon-separated strings. The first string is the abbreviated name  
2571 of the first month of the year (January), the second the abbreviated  
2572 name of the second month, and so on.

2573 **mon** Define the full month names, to be referenced by the %B field  
2574 descriptor. The operand consists of twelve or thirteen semicolon-  
2575 separated strings. The first string is the full name of the first month  
2576 of the year (January), the second the full name of the second month,  
2577 and so on.

2578 **d\_t\_fmt** Define the appropriate date and time representation, to be referenced  
2579 by the %c field descriptor. The operand consists of a string, and can

2580		contain any combination of characters and field descriptors. In addition,
2581		the string can contain escape sequences defined in Table 3.
2582	<b>d_fmt</b>	Define the appropriate date representation, to be referenced by the
2583		%x field descriptor. The operand consists of a string, and can contain
2584		any combination of characters and field descriptors. In addition, the
2585		string can contain escape sequences defined in Table 3.
2586	<b>t_fmt</b>	Define the appropriate time representation, to be referenced by the
2587		%X field descriptor. The operand consists of a string, and can
2588		contain any combination of characters and field descriptors. In
2589		addition, the string can contain escape sequences defined in Table 3.
2590	<b>am_pm</b>	Define the appropriate representation of the ante meridiem and post
2591		meridiem strings, to be referenced by the %p field descriptor. The
2592		operand consists of two strings, separated by a semicolon. The first
2593		string represents the antemeridiem designation, the last string the
2594		postmeridiem designation. The keyword is optional. If unspecified,
2595		the %p field descriptor refers to the empty string.
2596	<b>t_fmt_ampm</b>	Define the appropriate time representation in the 12-hour clock
2597		format with "am_pm", to be referenced by the %r field descriptor.
2598		The operand consists of a string and can contain any combination of
2599		characters and field descriptors. If the string is empty, the 12-hour
2600		format is not supported in the FDCC-set.

The following keywords are all optional

2602		
2603		
2604	<b>era</b>	Define how years are counted and displayed for each era in a locale.
2605		The operand shall consist of semicolon-separated strings. Each string
2606		shall be an era description segment with the format:
2607		direction:offset:start_date:end_date:era_name:era_format
2608		according to the definitions below. There can be as many era
2609		description segments as are necessary to describe the different eras.
2610		NOTE: The start of an era might not be the earliest point in the
2611		era - it may be the AD 1, and increases with earlier time.
2612	<b>direction</b>	Either a '+' or a '-' character. The '+' character shall
2613		indicate that years closer to the start_date have lower
2614		numbers than those closer to the end_date. The '-'
2615		character shall indicate that years closer to the start_date
2616		have higher numbers than those closer to the end_date.
2617	<b>offset</b>	The number of the year closest to the start_date in the
2618		era, corresponding to the %Ey conversion specification
2619	<b>start_date</b>	A date in the format YYYYMMDD, where YYYY,
2620		MM, and DD are the year, month, and day numbers
2621		respectively according to ISO 8601 of the start of the
2622		era. Years prior to AD 1 shall be represented as
2623		negative numbers.
2624	<b>end_date</b>	The ending date of the era, in the same format as the
2625		start_date, or one of the two special values "-*" or "+*".
2626		The value "-*" shall indicate that the ending date is the
2627		beginning of time. The value "+*" shall indicate that the
2628		ending date is the end of time.
2629	<b>era_name</b>	A string representing the name of the era, corresponding
2630		to the %EC conversion specification.
2631	<b>era_format</b>	A string for formatting the year in the era,

2632		corresponding to the %EY conversion specification.
2633	<b>era_year</b>	Define the format of the year in alternate Era format, corresponding to the %EY field descriptor.
2634		
2635	<b>era_d_t_fmt</b>	Define the format of the date and time in alternate Era notation, corresponding to the %Ec field descriptor.
2636		
2637	<b>era_d_fmt</b>	Define the format of the date in alternate Era notation, corresponding to the %Ex field descriptor.
2638		
2639	<b>era_t_fmt</b>	Define the format of the time in alternate Era notation, corresponding to the %EX field descriptor.
2640		
2641	<b>alt_digits</b>	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.
2642		
2643		
2644		
2645		
2646		
2647		
2648	<b>first_weekday</b>	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.
2649		
2650		
2651		
2652		
2653		
2654	<b>first_workday</b>	Define the first workday as an integer according to the day numbering specified with the "week" keyword.
2655		
2656	<b>cal_direction</b>	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:
2657		
2658		
2659		1 left-right from top
2660		2 top-down from left
2661		3 right-left from top
2662		The keyword may be omitted, and then the value 1 is taken.
2663	<b>timezone</b>	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:
2664		
2665		
2666		
2667		
2668		
2669		<std><offset><dst>[<offset>][,<rule>[,<rule>...]];
2670		
2671		where
2672		
2673		<std> and <dst> Indicates no less than three, nor more than 10
2674		characters that are the designation for the
2675		standard <std>, or Daylight Savings Time or
2676		summer time <dst> zone. Only <std> is
2677		required; if <dst> is missing, then Daylight
2678		Savings Time or summer time does not apply
2679		in this category. Upper- and lowercase letters
2680		are explicitly allowed. Any characters except a
2681		leading colon <:> or digits, the comma <,>, the
2682		minus <->, the plus <+>, and the null character
2683		are permitted to appear in these fields, but their



2684		meaning is unspecified.
2685		
2686	<offset>	Indicates the value one must add to the local time to arrive at the Coordinated Universal Time. The <offset> has the form:
2687		
2688		
2689		
2690		hh[:mm[:ss]]
2691		
2692		The minutes (mm) and seconds (ss) are optional. The hour (hh) is required and may be a single digit. The <offset> following <std> is required. If no <offset> follows <dst>, summer time is assumed to be one hour ahead of standard time. One or more digits may be used; the value is always interpreted as a decimal number. The hour is between zero and 24, and the minutes (and seconds) - if present - is between zero and 59. If preceded by a "-", the time zone is east of the Prime Meridian; otherwise it is west of (which may be indicated by an optional preceding "+").
2693		
2694		
2695		
2696		
2697		
2698		
2699		
2700		
2701		
2702		
2703		
2704		
2705	<rule>	A specification for Daylight Savings Time changes that indicates when to change to and back from summer time. The <rule> has the form:
2706		
2707		
2708		
2709		<date>[/<time>/<year>],<date>[/<time>/<year>]
2710		
2711		where the first <date> describes when the change from standard time to summer time occurs, and the second <date> describes when the change back happens. Each <time> field describes when, in current local time, the change to the other time is made. The first <year> field defines the beginning of the validity of this rule, and the second <year> field defines the end of the validity of the rule. A number of rules may be given.
2712		
2713		
2714		
2715		
2716		
2717		
2718		
2719		
2720		
2721		
2722		The format of <date> is one of the following:
2723		
2724	J<n>	The Julian day <n> (1 <= n <= 365) Leap years are not counted. That is, in all years - including leap years - February 28 is day 59 and March 1 is day 60. It is impossible to explicitly refer to the occasional February 29.
2725		
2726		
2727		
2728		
2729		
2730		
2731		
2732	<n>	The zero-based Julian day (0 <= n <= 365). Leap years are counted and it is possible to refer to February 29.
2733		
2734		
2735		

2736 M<m>.<n>.<d>  
 2737 the <d>th day (0 <= d <= 7)  
 2738 of week <n> of month <m> (1  
 2739 <= n <= 5, 1 <= m <= 12,  
 2740 where week 5 means "the last  
 2741 <d> day in month <m>"  
 2742 which may occur in either the  
 2743 fourth or fifth week). Week 1  
 2744 is the first week in which the  
 2745 <d>th day occurs. Day zero  
 2746 and day seven is Sunday.

2747  
 2748 The <time> has the same format as <offset>  
 2749 except that no leading sign ("- or "+) is  
 2750 allowed. The default, if <time> is not given, is  
 2751 "02:00:00".  
 2752

2753 The <year> has the format YYYY.  
 2754

2755 NOTE: This way of specifying the timezone is compatible with the  
 2756 format for the environment variable TZ described in Section 8.1.1 of  
 2757 POSIX.1.  
 2758

#### 2759 4.7.1 Date Field Descriptors

2760  
 2761 The LC\_TIME category defines the interpretation of a number of field descriptors. The  
 2762 field descriptors are also available in the definitions with the following LC\_TIME  
 2763 keywords: "d\_t\_fmt", "d\_fmt", "t\_fmt", "t\_fmt\_ampm", "era", "era\_d\_t\_fmt", "era\_d\_fmt",  
 2764 and "era\_t\_fmt". A field descriptor may not be used with the LC\_TIME keywords defining  
 2765 it.  
 2766

#### 2767 Table 3: Escape sequences for the date field

2768		
2769	%a	FDCC-set's abbreviated weekday name.
2770	%A	FDCC-set's full weekday name.
2771	%b	FDCC-set's abbreviated month name.
2772	%B	FDCC-set's full month name.
2773	%c	FDCC-set's appropriate date and time representation.
2774	%C	Century (a year divided by 100 and truncated to integer) as decimal 2775 number (00-99).
2776	%d	Day of the month as a decimal number (01-31).
2777	%D	Date in the format mm/dd/yy.
2778	%e	Day of the month as a decimal number (1-31 in at two-digit field with 2779 leading <space> fill).
2780	%F	The date in the format YYYY-MM-DD (ISO 8601 format).
2781	%g	Week-based year within century, as a decimal number (00-99).
2782	%G	Week-based year with century, as a decimal number (for example 1997).
2783	%h	A synonym for %b.
2784	%H	Hour (24-hour clock), as a decimal number (00-23).
2785	%I	Hour (12-hour clock), as a decimal number (01-12).
2786	%j	Day of the year, as a decimal number (001-366).
2787	%m	Month, as a decimal number (01-13).

2788	%M	Minute, as a decimal number (00-59).
2789	%n	A <newline> character.
2790	%p	FDCC-set's equivalent of either AM or PM.
2791	%r	12-hour clock time (01-12), using the AM/PM notation.
2792	%R	24-hour clock time, in the format "%H:%M".
2793	%S	Seconds, as a decimal number (00-61).
2794	%t	A <tab> character.
2795	%T	24-hour clock time, in the format HH:MM:SS.
2796	%u	Weekday, as a decimal number (1(Monday)-7).
2797	%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.
2800	%v	Week number of the year, as a decimal number with two digits including a possible leading zero, according to "week" keyword.
2802	%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.
2805	%w	Weekday, as a decimal number (0(Sunday)-6).
2806	%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.
2809	%x	FDCC-set's appropriate date representation.
2810	%X	FDCC-set's appropriate time representation.
2811	%y	Year within century (00-99).
2812	%Y	Year with century, as a decimal number.
2813	%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.
2816	%Z	Time-zone name, or no characters if no time zone is determinable.
2817	%%	A <percent-sign> character.

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

#### 4.7.2 Modified Field Descriptors

Some field descriptors can be modified by the E and O modifier characters to indicate a different format or specification as specified in the LC\_TIME FDCC-set description. If the corresponding keyword (see "era", "era\_year", "era\_d\_t\_fmt", "era\_d\_fmt", "era\_t\_fmt" and "alt\_digits") is not specified for the current FDCC-set, the unmodified field descriptor value is used.

2836	%Ec	FDCC-set's alternate date and time representation.
2837	%EC	The name of the base year (period) in the FDCC-set's alternate representation.
2839	%Ex	FDCC-set's alternate date representation.
2840	%EX	FDCC-set's alternate time representation.

2841	%Ey	Offset from %EC (year only) in the FDCC-set's alternate representation.
2842	%EY	Full alternate year representation.
2843	%Od	Day of month using the FDCC-set's alternate numeric symbols.
2844	%Oe	Day of month using the FDCC-set's alternate numeric symbols.
2845	%Of	Weekday as a decimal number according to alt_day (1 is first day).
2846	%OH	Hour (24-hour clock) using the FDCC-set's alternate numeric symbols.
2847	%OI	Hour (12-hour clock) using the FDCC-set's alternate numeric symbols.
2848	%Om	Month using the FDCC-set's alternate numeric symbols.
2849	%OM	Minutes using the FDCC-set's alternate numeric symbols.
2850	%OS	Seconds using the FDCC-set's alternate numeric symbols.
2851	%Ou	Weekday as a number in the alternate representation of the FDCC-set
2852		(Monday=1).
2853	%OU	Week number of the year (Sunday as the first day of the week) using the
2854		FDCC-set's alternate numeric symbols.
2855	%OV	Week number of the year (Monday as the first day of the
2856		week, ISO 8601 rules) using the alternate numeric symbols
2857		of the FDCC-set.
2858	%Ow	Weekday as number in the FDCC-set's alternate representation
2859		(Sunday=0).
2860	%OW	Week number of the year (Monday as the first day of the week) using the
2861		FDCC-set's alternate numeric symbols.
2862	%Oy	Year (offset from %C) in alternate representation.

### 4.7.3 "i18n" LC\_TIME category

The "i18n" LC\_TIME category is (following ISO 8601):

```

2868 LC_TIME
2869 % This is the ISO/IEC TR 14652 "i18n" definition for
2870 % the LC_TIME category.
2871 %
2872 % Weekday and week numbering according to ISO 8601
2873 abday    "<U0031>";"<U0032>";"<U0033>";"<U0034>";/
2874         "<U0035>";"<U0036>";"<U0037>"
2875 day     "<U0031>";"<U0032>";"<U0033>";"<U0034>";/
2876         "<U0035>";"<U0036>";"<U0037>"
2877 week    7;19971201;4
2878 abmon   "<U0030><U0031>";"<U0030><U0032>";"<U0030><U0033>";/
2879         "<U0030><U0034>";"<U0030><U0035>";"<U0030><U0036>";/
2880         "<U0030><U0037>";"<U0030><U0038>";"<U0030><U0039>";/
2881         "<U0031><U0030>";"<U0031><U0031>";"<U0031><U0032>"
2882 mon     "<U0030><U0031>";"<U0030><U0032>";"<U0030><U0033>";/
2883         "<U0030><U0034>";"<U0030><U0035>";"<U0030><U0036>";/
2884         "<U0030><U0037>";"<U0030><U0038>";"<U0030><U0039>";/
2885         "<U0031><U0030>";"<U0031><U0031>";"<U0031><U0032>"
2886 am_pm   "";"
2887 % Date formats following ISO 8601
2888 % Appropriate date and time representation (%c)
2889 %      "%F %T"
2890 d_t_fmt "<U0025><U0046><U0020><U0025><U0054>"
2891 %
2892 % Appropriate date representation (%x)      "%F"
2893 d_fmt   "<U0025><U0046>"
2894 %
2895 % Appropriate time representation (%X)      "%T"
2896 t_fmt   "<U0025><U0054>"
2897 t_fmt_ampm ""
2898 %
2899 END LC_TIME
2900
2901
2902
2903

```

#### 2904 4.8 LC\_MESSAGES

2905  
2906 The LC\_MESSAGES category defines the format and values for affirmative and negative  
2907 responses. The operands are strings or extended regular expressions to specify which  
2908 response strings that should be considered matches; see ISO/IEC 9945-2:1993 clause 2.8.4  
2909 for a definition of extended regular expressions. The following keywords are defined:

2910  
2911 **copy** Specify the name of an existing FDCC-set to be used as the source for the  
2912 definition of this category. If this keyword is specified, no other keyword  
2913 is specified.  
2914 **yesexpr** The operand consists of an extended regular expression that describes the  
2915 acceptable affirmative response to a question expecting an affirmative or  
2916 negative response.  
2917 **noexpr** The operand consists of an extended regular expression that describes the  
2918 acceptable negative response to a question expecting an affirmative or  
2919 negative response.  
2920

2921 The "i18n" LC\_MESSAGES category is:

```
2922 LC_MESSAGES
2923 % This is the ISO/IEC 14652 "i18n" definition for
2924 % the LC_MESSAGES category.
2925 %
2926 %
2927 yesexpr "<U005B><U002B><U0031><U005D>"
2928 noexpr "<U005B><U002D><U0030><U005D>"
2929 END LC_MESSAGES
```

2930  
2931 Note: This uses regular expression syntax with brackets ([]) to for example  
2932 specify the both <+> and <1> is allowed as an affirmative answer.

#### 2933 4.9 LC\_XLITERATE

2934  
2935 The LC\_XLITERATE category defines formats to transliterate strings, by transforming  
2936 substrings in the source to substrings in the target string. The capabilities are limited to  
2937 simple transliteration based on substring substitution, while more advanced transliteration  
2938 schemes, for example based on pattern matching, is either cumbersome to specify, or not  
2939 addressed. The transliteration may for example be from the Cyrillic script to the Latin  
2940 script.  
2941

2942  
2943 Transliteration of an incoming character string to a character string in a FDCC-set can be  
2944 specified with the following transliteration keywords and transliteration statements.

2945  
2946 **copy** Specify the name of an existing FDCC-set to be used as the  
2947 source for the definition of this category. If this keyword is  
2948 specified, no other keyword is specified.  
2949 **include** The name of the FDCC-set in text form to transliterate from,  
2950 and the repertoiremap for the FDCC-set to be used for the  
2951 definition of the transliteration statements. Other  
2952 transliteration statements may follow to replace specification  
2953 of the copied FDCC-set. This keyword is optional.  
2954 **default\_missing** defines a string of one or more characters to be put in the  
2955 output string if no transliteration statement can be applied to a  
2956 input <transliteration-source>. This keyword is optional.  
2957 **translit\_ignore** defines a set of characters, separated by semicolons, that are  
2958 to be ignored in the incoming character string, that is, each of

2959 the occurrences of such characters is treated as the empty  
 2960 string. The characters may use the notations defined in 4.3 for  
 2961 lists of characters. This keyword is optional.  
 2962 **redefine** This keyword introduces a list of transliteration statements  
 2963 where each of the <transliteration\_source> strings have been  
 2964 defined previously in the specification, and the new  
 2965 transliteration statements then replaces the old transliteration  
 2966 statements for the <transliteration\_source> strings specified.  
 2967 This keyword is optional.

#### 2968 4.9.1 Transliteration statements

2969 The syntax for a transliteration statement is:

```
2970 "%s %s;%s;...;%s\n",<transliteration_source>,<transliteration_string>,...
```

2971 Each <transliteration\_source> consists of one or more characters (in any of the forms  
 2972 defined in 4.1.1). The <transliteration\_source> that is the longest in terms of number of  
 2973 characters that match the input string is the one selected for transliteration.

2974 If a transliteration statement contains more than one <transliteration\_string>, the order that  
 2975 each <transliteration\_string> occurs in the transliteration statement defines the precedence  
 2976 order for choosing a particular <transliteration\_string> to substitute for the  
 2977 <transliteration\_source>. When a process makes use of a transliteration statement to  
 2978 transliterate text, and that transliteration statement contains more than one  
 2979 <transliteration\_string>, that process chooses the first <transliteration\_string>, in the  
 2980 defined precedence order, that satisfies the requirements of the transliteration.

2981 Note: the exact definition of the concept of satisfying the requirements of the transliteration is outside the  
 2982 context of this Technical Report. If, for example, a transliteration involves a change in the coded character set  
 2983 of a string, a <transliteration\_string> must be chosen, all of whose elements are members of that coded  
 2984 character set. In order to determine this, it would be expected that a repertoire describing which characters are  
 2985 to be present in the resulting transformed string be available to the transliteration API. Also, a transliteration  
 2986 may involve requirements such as that string length not change under transliteration. Such requirements may  
 2987 also affect the choice among alternative <transliteration\_string> values.

2988 If more than one transliteration statement is given for a given <transliteration\_source> this  
 2989 is an error, and duplicate transliteration statements are ignored. Tailoring of transliteration  
 2990 statements may be done via the "redefine" keyword.

#### 2991 4.9.2 "include" keyword

2992 The "include" keyword specifies a set of transliteration statements in text form to be  
 2993 included in the applied transliteration. The syntax of the "include" statement is:

```
2994 "include %s;%s\n", <FDCC-set>, <repertoiremap>
```

2995 <FDCC-set> is a string identifying the FDCC-set to be included from.

2996 <repertoiremap> is a string identifying the repertoiremap used in the FDCC-set being  
 2997 included, and is used to map character specifications from the specified FDCC-set into the  
 2998 current FDCC-set.

3000  
 3001  
 3002  
 3003  
 3004  
 3005  
 3006  
 3007  
 3008  
 3009  
 3010  
 3011  
 3012

### 4.9.3 Example of use of transliteration

```

LC_XLITERATE
include "de_DE"; "de_repmap"
default_missing <?>
translit_ignore <U3200>..

```

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>..

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:

- |                  |  |
|------------------|--|
| <b>copy</b>      | 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.  |
| <b>name_fmt</b>  | 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 escape sequences defined below. |
| <b>name_gen</b>  | The operand is a string defining a salutation valid for all persons.   |
| <b>name_miss</b> | The operand is a string defining a salutation valid for unmarried females.   |
| <b>name_mr</b>   | The operand is a string defining a salutation valid for males.   |
| <b>name_mrs</b>  | The operand is a string defining a salutation valid for married females.   |
| <b>name_ms</b>   | 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 escape sequences. The escape sequences are also available in the definitions with the following LC\_NAME keywords: "name\_fmt".

3077 Escape sequences for the "name\_fmt" keyword:  
 3078 %f Family names.  
 3079 %F Family names in uppercase.  
 3080 %g First given name.  
 3081 %G First given initial.  
 3082 %l First given name with latin letters.  
 3083 %o Other shorter name, eg. "Bill".  
 3084 %m Middle names.  
 3085 %M Middle initials.  
 3086 %p Profession.  
 3087 %s Salutation, such as "Doctor"  
 3088 %S Abbreviated salutation, such as "Mr." or "Dr."  
 3089 %d Salutation, using the FDCC-sets conventions, with 1 for the name\_gen, 2 for  
 3090 name\_mr, 3 for name\_mrs, 4 for name\_miss, 5 for name\_ms.  
 3091 %t If the preceding escape sequence resulted in an empty string, then the empty string,  
 3092 else a <space>.

3093  
 3094 Each escape sequence may have an <R> after the <%> to specify that the information is  
 3095 taken from a Romanized version string of the entity.

3096  
 3097 The "i18n" LC\_NAME category is:  
 3098

```
3099 LC_NAME
3100 % This is the ISO/IEC TR 14652 "i18n" definition for
3101 % the LC_NAME category.
3102 name_fmt "<U0025><U0070><U0025><U0074><U0025><U0067><U0025><U0074>/
3103 <U0025><U006D><U0025><U0074><U0025><U0066>"
3104 END LC_NAME
```

#### 3105 3106 4.11 LC\_ADDRESS

3107  
 3108 The LC\_ADDRESS category defines formats to be used in specifying a location like a  
 3109 person's living or office, for use in a postal address or in a letter, and other items related  
 3110 to geography. All keywords are optional. The following keywords are recognized:

3111  
 3112 **copy** Specify the name of an existing FDCC-set to be used as the source  
 3113 for the definition of this category. If this keyword is specified, no  
 3114 other keyword is specified.  
 3115 **postal\_fmt** Define the appropriate representation of a postal address such as  
 3116 street and city. The proper formatting of a person's name and title is  
 3117 done with the "name\_fmt" keyword of the LC\_NAME category. The  
 3118 operand consists of a string, and can contain any combination of  
 3119 characters and field descriptors. In addition, the string can contain  
 3120 escape sequences defined below.  
 3121 **country\_name** The operand is a string with the name of the country in the language  
 3122 of the FDCC-set.  
 3123 **country\_post** The operand is a string with the abbreviation of the country, used for  
 3124 postal addresses, for example by CEPT-MAILCODE.  
 3125 **lang\_name** The operand is a string with the name of the language in the  
 3126 language of the FDCC-set.  
 3127 **lang\_ab2** The operand is a string with the two-letter abbreviation of the  
 3128 language, according to ISO 639.  
 3129 **lang\_ab3\_term** The operand is a string with the three-letter abbreviation of the  
 3130 language for terminology use, according to ISO 639-2.



3131 **lang\_ab3\_lib** The operand is a string with the three-letter abbreviation of the  
 3132 language for library use, according to ISO 639-2. If not specified, the  
 3133 value of the "lang\_ab3\_term" keyword is taken.  
 3134

3135 Note: The "lang\_ab3\_term" and "lang\_ab3\_lib" keywords will in most cases contain the  
 3136 same value, but they may differ, e.g the values for the German language is "deu" and  
 3137 "ger" respectively.  
 3138

3139 The LC\_ADDRESS category defines the interpretation of a number of escape sequences.  
 3140 The escape sequences are also available in the definitions with the following  
 3141 LC\_ADDRESS keywords: "postal\_fmt".  
 3142

3143 Escape sequences for the "postal\_fmt" keyword:  
 3144

3145	%a	C/O address.
3146	%f	Firm name.
3147	%d	Department name.
3148	%b	Building name.
3149	%s	Street or block (eg. Japanese) name.
3150	%h	House number or designation.
3151	%N	If any graphical characters have been specified then an end of line is 3152 made.
3153	%t	If the preceding escape sequence resulted in an empty string, then the 3154 empty string, else a <space>.
3155	%r	Room number, door designation.
3156	%e	Floor number.
3157	%C	Country designation, from the <country_post> keyword.
3158	%l	Local township
3159	%z	Zip number, postal code.
3160	%T	Town, city.
3161	%S	State, province, or prefecture.
3162	%c	Country.

3163  
 3164 Each escape sequence may have an <R> after the <%> to specify that the information is  
 3165 taken from a Romanized version string of the entity.  
 3166

3167 NOTE: There are a number of variations for specifying a location among the cultures.  
 3168 Some of the information, like the middle names, or even the family name, is not used  
 3169 in some cultures. The specification here should be regarded as a starting point for this  
 3170 problem.  
 3171

3172 The "i18n" LC\_ADDRESS category is:  
 3173

```

3174 LC_ADDRESS
3175 % This is the ISO/IEC TR 14652 "i18n" definition for
3176 % the LC_ADDRESS category.
3177 %
3178 %
3179 postal_fmt      "<U0025><U0061><U0025><U004E><U0025><U0066><U0025><U004E>/
3180 <U0025><U0064><U0025><U004E><U0025><U0062><U0025><U004E><U0025><U0073>/
3181 <U0020><U0025><U0068><U0020><U0025><U0065><U0020><U0025><U0072><U0025>/
3182 <U004E><U0025><U0043><U002D><U0025><U007A><U0020><U0025><U0054><U0025>/
3183 <U004E><U0025><U0063><U0025><U004E>"
3184 END LC_ADDRESS
  
```

3185  
 3186

3187 **4.12 LC\_TELEPHONE**

3188

3189 The LC\_TELEPHONE category defines formats to be used with telephone services. All  
3190 keywords are optional. The following keywords are defined:

3191

3192 **copy** Specify the name of an existing FDCC-set to be used as the source  
3193 for the definition of this category. If this keyword is specified, no  
3194 other keyword is specified.

3195 **tel\_int\_fmt** Define the appropriate representation of a telephone number for  
3196 international use. The operand consists of a string, and can contain  
3197 any combination of characters and field descriptors. In addition, the  
3198 string can contain escape sequences defined below.

3199 **tel\_dom\_fmt** Define the appropriate representation of a telephone number for  
3200 domestic use. The operand consists of a string, and can contain any  
3201 combination of characters and field descriptors. In addition, the string  
3202 can contain escape sequences defined below.

3203 **int\_select** The operand is a string with the digits used to call international  
3204 telephone numbers.

3205 **int\_prefix** The operand is a string with the prefix used from other countries to  
3206 call the area.

3207

3208 The LC\_TELEPHONE category defines the interpretation of a number of escape  
3209 sequences. The escape sequences are also available in the definitions with the following  
3210 LC\_TELEPHONE keywords: "tel\_int\_fmt" and "tel\_dom\_fmt".

3211

3212 %a area code without prefix (prefix is often <0>).

3213 %A area code including prefix (prefix is often <0>).

3214 %l local number.

3215 %c country code

3216 %C alternative carrier service code used for dialling abroad

3217

3218 The "i18n" LC\_TELEPHONE category is:

3219

3220

```
3221 LC_TELEPHONE
3222 % This is the ISO/IEC TR 14652 "i18n" definition for
3223 % the LC_TELEPHONE category.
```

3224

```
3225 %
3225 tel_int_fmt      "<U002B><U0025><U0063><U0020><U002B><U0061><U0020><U002B>/
```

3226

3227

3228

3229

3230

3231

**5. CHARMAP**

3232 A character set description may exist for each coded character set supported by an  
3233 application. This text is referred elsewhere in this Technical Report as a charmap.

3234

3235 A conforming charmap to be used with a FDCC-set supports the portable character set  
3236 specified in Table 1.

3237

3238 Conforming charmaps specify certain character and character set attributes, as defined in  
3239 5.1.

3240

3241

## 3242 5.1 Character Set Description Text

3243  
3244 The character set description text (charmap) describes the mapping between symbolic  
3245 character names and actual encoding of a coded character set. It is used to bind the  
3246 symbolic character names in a FDCC-set to an actual encoding, so an application can  
3247 process data in this encoding.

3248  
3249 The following declarations can precede the character definitions. Each consist of the  
3250 symbol shown in the following list, starting in column 1, including the surrounding  
3251 brackets, followed by one of more "blank"s, followed by the value to be assigned to the  
3252 symbol. If any of the declarations are included, they are specified in the order shown in  
3253 the following list:

3254  
3255 **<code\_set\_name>** The name of the coded character set for which the character set  
3256 description text is defined. The characters of the name are taken  
3257 from the set of characters with visible glyphs defined in Table 1.

3258  
3259 **<mb\_cur\_max>** The maximum number of bytes in a multibyte character. This  
3260 defaults to 1.

3261  
3262 **<mb\_cur\_min>** An unsigned positive integer value that defines the minimum  
3263 number of bytes in a character for the encoded character set. The  
3264 value is less or equal to "mb\_cur\_max". If not specified, the  
3265 minimum number is equal to "mb\_cur\_max".

3266  
3267 **<escape\_char>** The escape character used to indicate that the characters  
3268 following is interpreted in a special way, as defined later in this  
3269 subclause. This defaults to backslash (\). The character slash (/)  
3270 is used in all the following text and examples, unless otherwise  
3271 noted.

3272  
3273 **<comment\_char>** The character that when placed in column 1 of a charmap line,  
3274 is used to indicate that the line is ignored. The default character  
3275 is the number sign (#). The character percent-sign (%) is used in  
3276 all the following text and examples, unless otherwise noted.

3277  
3278 **<repertoiremap>** The name of the repertoiremap used to define the symbolic  
3279 character names in the charmap. The characters of the name are  
3280 taken from the set of characters with visible glyphs defined in  
3281 Table 1.

3282  
3283 **<escseq2022>** defines the escape sequences for ISO 2022 shifting for the coded  
3284 character set defined by the charmap. The semicolon-separated  
3285 operands are all strings with characters taken from the set of  
3286 characters with visible glyphs defined in table 1. The first  
3287 operand defines the g-set or c-set to be defined, and the  
3288 following values are defined: c0, c1, g0, g1, g2, g3. The second  
3289 operand defines what range of characters in the charmap is  
3290 affected, and the values defined are: c0, c1, g0, g1. The third  
3291 operand is the escape sequence that is defined.

3292  
3293

3294	<addset>	the name of the charmap to be added to the current coded
3295		character set, and to be selected by the escape sequences defined
3296		by <escseq> of the added charmap.
3297		
3298	<include>	include the encoding of another charmap in the current charmap.
3299		The semicolon-separated operands are all strings with characters
3300		taken from the set of characters with visible glyphs defined in
3301		table 1. The first operand defines the g-set or c-set to be defined
3302		in the current charmap, and the following values are defined: c0,
3303		c1, g0, g1, g2, g3. The second operand defines a range of
3304		characters in the referenced charmap, and the values defined are:
3305		c0, c1, g0, g1. The third operand is the name of the charmap to
3306		be included. The coded character sets are defined initially for the
3307		encoding, and therefore do not need escape sequences for
3308		identification. If two g0 sets are defined, the second is switched
3309		to using the SHIFT OUT control character, while the first is
3310		shifted to using the SHIFT IN control character.
3311		

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

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

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

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

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

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

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

In the second and third syntax (symbolic decimal ellipsis), the line in the character set mapping defines a range of one or more symbolic names. The difference between the second and the third syntax is the number of dots in the ellipsis: the second has 3 dots, the third has 4 dots. In these forms the symbolic names consist of zero or more nonnumeric characters from the set shown with visible glyphs in Table 1, followed by an integer formed by one or more decimal digits. The characters preceding the integer are identical in the two symbolic names, and the integer formed by the digits in the second symbolic

3346 name are identical to or greater than the integer formed by the digits in the first name.  
 3347 This is interpreted as a series of symbolic names formed from the common part and each  
 3348 of the integers in decimal format between the first and the second integer, inclusive, and  
 3349 with a length of the symbolic names generated that is equal to the length of the first (and  
 3350 also the second) symbolic name. As an example, <j0101>...<j0104> is interpreted as the  
 3351 symbolic names <j0101>, <j0102>, <j0103>, and <j0104>, in that order.  
 3352

3353 Note: The rationale to allow both a 3-dot and a 4-dot symbol for symbolic decimal  
 3354 ellipses is that in the POSIX standard the decimal symbolic ellipses was defined by a 3-  
 3355 dot symbol for charmaps, while the 3-dot symbol was an absolute ellipses for POSIX  
 3356 locales, and this Technical Report specifies a 4-dot symbol for the decimal symbolic  
 3357 ellipses. The 3-dot symbolic decimal ellipses in charmaps is deprecated.  
 3358

3359 In the fourth syntax (symbolic hexadecimal ellipsis, with two dots), the line in the  
 3360 character set mapping defines a range of one or more symbolic names. In this form the  
 3361 symbolic names consist of zero or more nonnumeric characters from the set shown with  
 3362 visible glyphs in Table 1, followed by an integer formed by one or more hexadecimal  
 3363 digits, using uppercase letters only for the range "A" to "F". The characters preceding the  
 3364 hexadecimal integer are identical in the two symbolic names, and the integer formed by  
 3365 the hexadecimal digits in the second symbolic name is identical to or greater than the  
 3366 integer formed by the hexadecimal digits in the first name. This is interpreted as a series  
 3367 of symbolic names formed from the common part and each of the integers in hexadecimal  
 3368 format using uppercase letters only between the first and the second integer, inclusive, and  
 3369 with a length of the symbolic names generated that is equal to the length of the first (and  
 3370 also the second) symbolic name. As an example, <U010E>..<U0111> is interpreted as the  
 3371 symbolic names <U010E>, <U010F>, <U0110>, and <U0111>, in that order.  
 3372

3373 The encoding part is expressed as one (for single-byte values) or more concatenated  
 3374 decimal, octal or hexadecimal constants (hexadecimal constants is recommended). Decimal  
 3375 constants are represented by two or three decimal digits, preceded by the escape character  
 3376 and the lowercase letter "d"; for example /d05, /d97, or /d143. Hexadecimal constants are  
 3377 represented by two hexadecimal digits, preceded by the escape character and the lowercase  
 3378 letter "x"; for example /x05, /x61, or /x8f. Octal constants are represented by two or three  
 3379 octal digits, preceded by the escape character; for example /05, /141, or /217. In a  
 3380 charmap, each constant should represent an 8 bit byte for portability reasons. Applications  
 3381 supporting other byte sizes may allow constants to represent values larger than those that  
 3382 can be represented in 8 bit bytes, and to allow additional digits in constants. When  
 3383 constants are concatenated for multibyte character values, they may be of different types,  
 3384 and interpreted in byte order from the first to the last with the least significant byte of the  
 3385 multibyte character specified by the last byte. The manner in which these constants are  
 3386 represented in the character stored in the system is application defined. Omitting bytes  
 3387 from a multibyte character produces undefined results.  
 3388

3389 In lines defining ranges of symbolic names, the encoded value is the value for the first  
 3390 symbolic name in the range (the symbolic name preceding the ellipsis). Subsequent  
 3391 symbolic names defined by the range have encoding values in increasing order. For  
 3392 example the line  
 3393

3394 <j0101>...<j0104>            /d129/d254

3395  
 3396 is interpreted as  
 3397

3398 <j0101> /d129/d254  
 3399 <j0102> /d129/d255  
 3400 <j0103> /d130/d000  
 3401 <j0104> /d130/d001

3402  
 3403 The comments parameter is optional.

3404  
 3405  
 3406 Example of using ISO 2022 techniques:

3407  
 3408 The following example defines two coded character sets, a 7-bit and a 14-bit. They are then merged into one  
 3409 encoding. It is an example on how encodings used in Eastern Asia could be specified.

3410  
 3411 The 7-bit charmap

```
3412
3413 <escape_char> /
3414 <comment_char> %
3415 % The 7-bit charmap defines both control and graphic characters
3416 <code_set_name> "eastern7bit"
3417 <escseq2022> "c0";"c0","/x21/x40"
3418 <escseq2022> "g0";"g0","/x28/x48"
3419 <escseq2022> "g1";"g0","/x29/x48"
3420 <escseq2022> "g2";"g0","/x2A/x48"
3421 <escseq2022> "g3";"g0","/x2B/x48"
3422
3423 CHARMAP
3424 <tab> /x08
3425 <newline> /x0D
3426 <a> /x61
3427 % more character encodings to be defined here
3428 END CHARMAP
```

3429  
 3430  
 3431 The 14-bit charmap

```
3432
3433 <escape_char> /
3434 <comment_char> %
3435 <code_set_name> "eastern14bit"
3436 <mb_cur_max> 2
3437 <esqseq> "g0";"g0";"/x24/x40"
3438 <esqseq> "g1";"g0";"/x24/x29/x40"
3439 <esqseq> "g2";"g0";"/x24/x2A/x40"
3440 <esqseq> "g3";"g0";"/x24/x2B/x40"
3441 CHARMAP
3442 <U0165> /d036/d055 % the character codes are only examples
3443 <U0244> /d036/d056
3444 % more character encodings to be defined here
3445 END CHARMAP
```

3446  
 3447  
 3448 The merged encoding

```
3449
3450 <escape_char> /
3451 <comment_char> %
3452 <code_set_name> "shift-eastern"
3453 <mb_cur_max> 2
3454 <mb_cur_min> 1
3455 <include> "c0";"c0";"eastern7bit"
3456 <include> "g0";"g0";"eastern7bit"
3457 <include> "g1";"g0";"eastern14bit"
3458 % This defines the g0 values of "eastern14bit" (without the 8th
3459 % bit set) to be the g1 in this encoding (with the 8th bit set).
3460 %
3461 % So the bytes without the 8th bit set is from the "shift7bit"
3462 % coded character set, while bytes with the 8th bit set are from
3463 % the 14-bit set.
```

3464

3465 Another merged encoding using the same charmaps:

```
3466 <escape_char> /
3467 <comment_char> %
3468 <code_set_name> "EUC-eastern"
3469 <mb_cur_max> 2
3470 <mb_cur_min> 1
3471 <include> "c0";"c0";"eastern7bit"
3472 <include> "g0";"g0";"eastern7bit"
3473 <include> "g0";"g0";"eastern14bit"
3474 % As there are two "g0" sets defined, the first referenced is the
3475 % initial g0 set, while the second can be shifted to via the SHIFT OUT
3476 % control character. The first can then be shifted to by the SHIFT IN
3477 % control character.
```

3479

3480

## 3481 WIDTH section

3482

3483 After the "END CHARMAP" statement the following declarations may follow. Each  
3484 consists of the keyword shown in the following list, starting in column 1, followed by the  
3485 value(s) to be associated to the keyword, as defined below.

3486

3487 WIDTH An unassigned positive integer value defining the column width for the characters  
3488 in the coded character set. Coded character values are defined using symbolic character  
3489 names followed by a column width value. Defining a character with more than one  
3490 WIDTH produces undefined results. The END WIDTH keyword is used to terminate the  
3491 WIDTH definitions.

3492

3493 WIDTH\_DEFAULT An unsigned positive integer value defining the column width for any  
3494 character not listed by one of the WIDTH keywords. If no WIDTH\_DEFAULT keyword  
3495 is included in the charmap, the default character width is 1.

3496

3497 Example:

3498

3499 After the "END CHARMAP" statement, a syntax for width definition would be:

3500

```
3501 WIDTH
3502 <A> 1
3503 <B> 1
3504 <j0101>...<j0195> 2
3505 <U3200>..<UFAFF> 2
3506 END WIDTH
3507 WIDTH_DEFAULT 1
```

3508

3509 In this example, the code point values represented by <A> and <B> are assigned a width of 1. The code  
3510 point values <j0101>...<j0195> (decimal ellipses) and <U3200>..<UFAFF> are assigned a width of 2. The  
3511 last line defines the DEFAULT\_WIDTH to 1.

3512

3513

## 3514 6 REPERTOIREMAP

3515

3516 FDCC-set and Charmap sources may be specified in a coded character set independent  
3517 way, using symbolic character names. The relation between the symbolic character names  
3518 and characters may be specified via a Repertoiremap, which defines the repertoire of  
3519 characters defined for a FDCC-set, and the symbolic character names and corresponding  
3520 abstract character (by a reference to ISO/IEC 10646).

3521

3522 The repertoire mapping is defined by specifying the symbolic character name and the  
3523 ISO/IEC 10646 code position in hexadecimal form (with a preceding 'U') and optionally

3524 the long ISO/IEC 10646 character name in the following syntax:

3525  
3526     "%s %s %s\n",<symbolic-name>,<10646-short-identifier>,<comments>  
3527

3528 The symbolic character name and the ISO/IEC 10646 short identifier are each surrounded  
3529 by angle brackets <>, and the fields are separated by one or more spaces or tabs on a line.  
3530 If a right angle bracket or an escape character is used within a symbolic name, it is  
3531 preceded by the escape character. Characters not in ISO/IEC 10646 may be referenced by  
3532 the symbolic character names <P00000000>..<PF8FFFFFFF>.

3533  
3534 The escape character can be redefined from the default reverse solidus (\) with the first  
3535 line of the Repertoiremap containing the string "escape\_char" followed by one or more  
3536 spaces or tabs and then the escape character.

3537  
3538 Several symbolic character names can refer to the same abstract character, and are then  
3539 used as synonyms in FDCC-sets and charmaps. The set of <U0000>..<UFFFF> and  
3540 <U00000000>..<U7FFFFFFF> symbolic names (no lowercase letters) are predefined and  
3541 refers to the corresponding code points of ISO/IEC 10646 with the same short identifier.  
3542

3543 The "i18nrep" repertoiremap is defined to accommodate prior art, such as defined in  
3544 Annex G of the ISO/IEC 9945-2:1993 standard, and used by ISO and IEC member bodies  
3545 in their national POSIX locale specifications, and as used in POSIX locales distributed by  
3546 the ISO/IEC POSIX working group and The Open Group. Many POSIX charmaps  
3547 registered with ISO/IEC 15897 use these symbolic names. It also reflects use on the  
3548 Internet, and many of the Internet registered charsets are specified using these symbolic  
3549 names. The "i18nrep" repertoiremap thus facilitates reuse of both POSIX locale data and  
3550 POSIX charmaps with data from this Technical Report. The sequence <a8>..<z8> are used  
3551 as hooks for tailoring to denote the last accented Latin letter of each of the ISO/IEC 646  
3552 letters <a>..<z>, so that tailorings that need to have specifications after the last letter of  
3553 such a family, for example to introduce a new letter of an alphabet, can do so with a  
3554 reference that is stable over different versions of the "i18n" FDCC-set. The contents of the  
3555 "i18nrep" repertoiremap is as follows:

```

3556 escape_char /
3557 <NUL>          <U0000>  NULL (NUL)
3558 <SOH>          <U0001>  START OF HEADING (SOH)
3559 <STX>          <U0002>  START OF TEXT (STX)
3560 <ETX>          <U0003>  END OF TEXT (ETX)
3561 <EOT>          <U0004>  END OF TRANSMISSION (EOT)
3562 <ENQ>          <U0005>  ENQUIRY (ENQ)
3563 <ACK>          <U0006>  ACKNOWLEDGE (ACK)
3564 <alert>        <U0007>  BELL (BEL)
3565 <BEL>          <U0007>  BELL (BEL)
3566 <backspace>    <U0008>  BACKSPACE (BS)
3567 <tab>          <U0009>  CHARACTER TABULATION (HT)
3568 <newline>      <U000A>  LINE FEED (LF)
3569 <vertical-tab> <U000B>  LINE TABULATION (VT)
3570 <form-feed>    <U000C>  FORM FEED (FF)
3571 <carriage-return> <U000D>  CARRIAGE RETURN (CR)
3572 <DLE>          <U0010>  DATALINK ESCAPE (DLE)
3573 <DC1>          <U0011>  DEVICE CONTROL ONE (DC1)
3574 <DC2>          <U0012>  DEVICE CONTROL TWO (DC2)
3575 <DC3>          <U0013>  DEVICE CONTROL THREE (DC3)
3576 <DC4>          <U0014>  DEVICE CONTROL FOUR (DC4)
3577 <NAK>          <U0015>  NEGATIVE ACKNOWLEDGE (NAK)
3578 <SYN>          <U0016>  SYNCHRONOUS IDLE (SYN)
3579 <ETB>          <U0017>  END OF TRANSMISSION BLOCK (ETB)
3580 <CAN>          <U0018>  CANCEL (CAN)
3581 <SUB>          <U001A>  SUBSTITUTE (SUB)
3582 <ESC>          <U001B>  ESCAPE (ESC)
3583 <IS4>          <U001C>  FILE SEPARATOR (IS4)
3584 <IS3>          <U001D>  GROUP SEPARATOR (IS3)
3585 <intro>        <U001D>  GROUP SEPARATOR (IS3)
3586 <IS2>          <U001E>  RECORD SEPARATOR (IS2)
3587 <IS1>          <U001F>  UNIT SEPARATOR (IS1)
3588 <DEL>          <U007F>  DELETE (DEL)

```



<space>	<U0020>	SPACE
<exclamation-mark>	<U0021>	EXCLAMATION MARK
<quotation-mark>	<U0022>	QUOTATION MARK
<number-sign>	<U0023>	NUMBER SIGN
<dollar-sign>	<U0024>	DOLLAR SIGN
<percent-sign>	<U0025>	PERCENT SIGN
<ampersand>	<U0026>	AMPERSAND
<apostrophe>	<U0027>	APOSTROPHE
<left-parenthesis>	<U0028>	LEFT PARENTHESIS
<right-parenthesis>	<U0029>	RIGHT PARENTHESIS
<asterisk>	<U002A>	ASTERISK
<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>	<U0252>	Weight indicating the position of the last a
<b8>	<U0182>	Weight indicating the position of the last b
<c8>	<U0255>	Weight indicating the position of the last c
<d8>	<U018D>	Weight indicating the position of the last d
<e8>	<U0264>	Weight indicating the position of the last e
<f8>	<U0191>	Weight indicating the position of the last f
<g8>	<U01A2>	Weight indicating the position of the last g
<h8>	<U02BD>	Weight indicating the position of the last h
<i8>	<U0196>	Weight indicating the position of the last i
<j8>	<U0284>	Weight indicating the position of the last j
<k8>	<U029E>	Weight indicating the position of the last k
<l8>	<U028E>	Weight indicating the position of the last l
<m8>	<U0271>	Weight indicating the position of the last m
<n8>	<U014A>	Weight indicating the position of the last n
<o8>	<U0277>	Weight indicating the position of the last o
<p8>	<U0278>	Weight indicating the position of the last p
<q8>	<U0138>	Weight indicating the position of the last q
<r8>	<U02B6>	Weight indicating the position of the last r
<s8>	<U0286>	Weight indicating the position of the last s
<t8>	<U0287>	Weight indicating the position of the last t
<u8>	<U01B1>	Weight indicating the position of the last u
<v8>	<U028C>	Weight indicating the position of the last v
<w8>	<U028D>	Weight indicating the position of the last w
<x8>	<U216B>	Weight indicating the position of the last x
<y8>	<U01B3>	Weight indicating the position of the last y
<z8>	<U0293>	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)

67	<HT>	<U0009>	CHARACTER TABULATION (HT)
68	<LF>	<U000A>	LINE FEED (LF)
69	<VT>	<U000B>	LINE TABULATION (VT)
70	<FF>	<U000C>	FORM FEED (FF)
71	<CR>	<U000D>	CARRIAGE RETURN (CR)
72	<SO>	<U000E>	SHIFT OUT (SO)
73	<SI>	<U000F>	SHIFT IN (SI)
74	<DL>	<U0010>	DATALINK ESCAPE (DLE)
75	<D1>	<U0011>	DEVICE CONTROL ONE (DC1)
76	<D2>	<U0012>	DEVICE CONTROL TWO (DC2)
77	<D3>	<U0013>	DEVICE CONTROL THREE (DC3)
78	<D4>	<U0014>	DEVICE CONTROL FOUR (DC4)
79	<NK>	<U0015>	NEGATIVE ACKNOWLEDGE (NAK)
80	<SY>	<U0016>	SYNCHRONOUS IDLE (SYN)
81	<EB>	<U0017>	END OF TRANSMISSION BLOCK (ETB)
82	<CN>	<U0018>	CANCEL (CAN)
83	<EM>	<U0019>	END OF MEDIUM (EM)
84	<SB>	<U001A>	SUBSTITUTE (SUB)
85	<EC>	<U001B>	ESCAPE (ESC)
86	<FS>	<U001C>	FILE SEPARATOR (IS4)
87	<GS>	<U001D>	GROUP SEPARATOR (IS3)
88	<RS>	<U001E>	RECORD SEPARATOR (IS2)
89	<US>	<U001F>	UNIT SEPARATOR (IS1)
90	<DT>	<U007F>	DELETE (DEL)
91	<PA>	<U0080>	PADDING CHARACTER (PAD)
92	<HO>	<U0081>	HIGH OCTET PRESET (HOP)
93	<BH>	<U0082>	BREAK PERMITTED HERE (BPH)
94	<NH>	<U0083>	NO BREAK HERE (NBH)
95	<IN>	<U0084>	INDEX (IND)
96	<NL>	<U0085>	NEXT LINE (NEL)
97	<SA>	<U0086>	START OF SELECTED AREA (SSA)
98	<ES>	<U0087>	END OF SELECTED AREA (ESA)
99	<HS>	<U0088>	CHARACTER TABULATION SET (HTS)
100	<HJ>	<U0089>	CHARACTER TABULATION WITH JUSTIFICATION (HTJ)
101	<VS>	<U008A>	LINE TABULATION SET (VTS)
102	<PD>	<U008B>	PARTIAL LINE FORWARD (PLD)
103	<PU>	<U008C>	PARTIAL LINE BACKWARD (PLU)
104	<RI>	<U008D>	REVERSE LINE FEED (RI)
105	<S2>	<U008E>	SINGLE-SHIFT TWO (SS2)
106	<S3>	<U008F>	SINGLE-SHIFT THREE (SS3)
107	<DC>	<U0090>	DEVICE CONTROL STRING (DCS)
108	<P1>	<U0091>	PRIVATE USE ONE (PU1)
109	<P2>	<U0092>	PRIVATE USE TWO (PU2)
110	<TS>	<U0093>	SET TRANSMIT STATE (STS)
111	<CC>	<U0094>	CANCEL CHARACTER (CCH)
112	<MW>	<U0095>	MESSAGE WAITING (MW)
113	<SG>	<U0096>	START OF GUARDED AREA (SPA)
114	<EG>	<U0097>	END OF GUARDED AREA (EPA)
115	<SS>	<U0098>	START OF STRING (SOS)
116	<GC>	<U0099>	SINGLE GRAPHIC CHARACTER INTRODUCER (SGCI)
117	<SC>	<U009A>	SINGLE CHARACTER INTRODUCER (SCI)
118	<CI>	<U009B>	CONTROL SEQUENCE INTRODUCER (CSI)
119	<ST>	<U009C>	STRING TERMINATOR (ST)
120	<OC>	<U009D>	OPERATING SYSTEM COMMAND (OSC)
121	<PM>	<U009E>	PRIVACY MESSAGE (PM)
122	<AC>	<U009F>	APPLICATION PROGRAM COMMAND (APC)
123	<SP>	<U0020>	SPACE
124	<!>	<U0021>	EXCLAMATION MARK
125	<">	<U0022>	QUOTATION MARK
126	<#>	<U0023>	NUMBER SIGN
127	<\$>	<U0024>	DOLLAR SIGN
128	<%>	<U0025>	PERCENT SIGN
129	<&>	<U0026>	AMPERSAND
130	<'>	<U0027>	APOSTROPHE
131	<(>	<U0028>	LEFT PARENTHESIS
132	<)>	<U0029>	RIGHT PARENTHESIS
133	<*>	<U002A>	ASTERISK
134	<+>	<U002B>	PLUS SIGN
135	<, >	<U002C>	COMMA
136	<->	<U002D>	HYPHEN-MINUS
137	<.>	<U002E>	FULL STOP
138	< / >	<U002F>	SOLIDUS
139	<0>	<U0030>	DIGIT ZERO
140	<1>	<U0031>	DIGIT ONE
141	<2>	<U0032>	DIGIT TWO
142	<3>	<U0033>	DIGIT THREE
143	<4>	<U0034>	DIGIT FOUR
144	<5>	<U0035>	DIGIT FIVE
145	<6>	<U0036>	DIGIT SIX
146	<7>	<U0037>	DIGIT SEVEN
147	<8>	<U0038>	DIGIT EIGHT
148	<9>	<U0039>	DIGIT NINE
149	<:>	<U003A>	COLON
150	< ; >	<U003B>	SEMICOLON
151	<<>	<U003C>	LESS-THAN SIGN
152	<=>	<U003D>	EQUALS SIGN
153	< / >>	<U003E>	GREATER-THAN SIGN
154	<?>	<U003F>	QUESTION MARK

766	<At>	<U0040>	COMMERCIAL AT
767	<A>	<U0041>	LATIN CAPITAL LETTER A
768	<B>	<U0042>	LATIN CAPITAL LETTER B
769	<C>	<U0043>	LATIN CAPITAL LETTER C
770	<D>	<U0044>	LATIN CAPITAL LETTER D
771	<E>	<U0045>	LATIN CAPITAL LETTER E
772	<F>	<U0046>	LATIN CAPITAL LETTER F
773	<G>	<U0047>	LATIN CAPITAL LETTER G
774	<H>	<U0048>	LATIN CAPITAL LETTER H
775	<I>	<U0049>	LATIN CAPITAL LETTER I
776	<J>	<U004A>	LATIN CAPITAL LETTER J
777	<K>	<U004B>	LATIN CAPITAL LETTER K
778	<L>	<U004C>	LATIN CAPITAL LETTER L
779	<M>	<U004D>	LATIN CAPITAL LETTER M
780	<N>	<U004E>	LATIN CAPITAL LETTER N
781	<O>	<U004F>	LATIN CAPITAL LETTER O
782	<P>	<U0050>	LATIN CAPITAL LETTER P
783	<Q>	<U0051>	LATIN CAPITAL LETTER Q
784	<R>	<U0052>	LATIN CAPITAL LETTER R
785	<S>	<U0053>	LATIN CAPITAL LETTER S
786	<T>	<U0054>	LATIN CAPITAL LETTER T
787	<U>	<U0055>	LATIN CAPITAL LETTER U
788	<V>	<U0056>	LATIN CAPITAL LETTER V
789	<W>	<U0057>	LATIN CAPITAL LETTER W
790	<X>	<U0058>	LATIN CAPITAL LETTER X
791	<Y>	<U0059>	LATIN CAPITAL LETTER Y
792	<Z>	<U005A>	LATIN CAPITAL LETTER Z
793	<< >	<U005B>	LEFT SQUARE BRACKET
794	</// >	<U005C>	REVERSE SOLIDUS
795	< / >	<U005D>	RIGHT SQUARE BRACKET
796	<' >	<U005E>	CIRCUMFLEX ACCENT
797	<_ >	<U005F>	LOW LINE
798	<' ! >	<U0060>	GRAVE ACCENT
799	<a >	<U0061>	LATIN SMALL LETTER A
800	<b >	<U0062>	LATIN SMALL LETTER B
801	<c >	<U0063>	LATIN SMALL LETTER C
802	<d >	<U0064>	LATIN SMALL LETTER D
803	<e >	<U0065>	LATIN SMALL LETTER E
804	<f >	<U0066>	LATIN SMALL LETTER F
805	<g >	<U0067>	LATIN SMALL LETTER G
806	<h >	<U0068>	LATIN SMALL LETTER H
807	<i >	<U0069>	LATIN SMALL LETTER I
808	<j >	<U006A>	LATIN SMALL LETTER J
809	<k >	<U006B>	LATIN SMALL LETTER K
810	<l >	<U006C>	LATIN SMALL LETTER L
811	<m >	<U006D>	LATIN SMALL LETTER M
812	<n >	<U006E>	LATIN SMALL LETTER N
813	<o >	<U006F>	LATIN SMALL LETTER O
814	<p >	<U0070>	LATIN SMALL LETTER P
815	<q >	<U0071>	LATIN SMALL LETTER Q
816	<r >	<U0072>	LATIN SMALL LETTER R
817	<s >	<U0073>	LATIN SMALL LETTER S
818	<t >	<U0074>	LATIN SMALL LETTER T
819	<u >	<U0075>	LATIN SMALL LETTER U
820	<v >	<U0076>	LATIN SMALL LETTER V
821	<w >	<U0077>	LATIN SMALL LETTER W
822	<x >	<U0078>	LATIN SMALL LETTER X
823	<y >	<U0079>	LATIN SMALL LETTER Y
824	<z >	<U007A>	LATIN SMALL LETTER Z
825	<( ! >	<U007B>	LEFT CURLY BRACKET
826	< ! ! >	<U007C>	VERTICAL LINE
827	< ! ) >	<U007D>	RIGHT CURLY BRACKET
828	<' ? >	<U007E>	TILDE
829	<NS >	<U00A0>	NO-BREAK SPACE
830	< ! I >	<U00A1>	INVERTED EXCLAMATION MARK
831	<Ct >	<U00A2>	CENT SIGN
832	<Pd >	<U00A3>	POUND SIGN
833	<Cu >	<U00A4>	CURRENCY SIGN
834	<Ye >	<U00A5>	YEN SIGN
835	<BB >	<U00A6>	BROKEN BAR
836	<SE >	<U00A7>	SECTION SIGN
837	<' : >	<U00A8>	DIAERESIS
838	<Co >	<U00A9>	COPYRIGHT SIGN
839	<-a >	<U00AA>	FEMININE ORDINAL INDICATOR
840	<<< >	<U00AB>	LEFT-POINTING DOUBLE ANGLE QUOTATION MARK
841	<NO >	<U00AC>	NOT SIGN
842	<-- >	<U00AD>	SOFT HYPHEN
843	<Rg >	<U00AE>	REGISTERED SIGN
844	<' m >	<U00AF>	MACRON
845	<DG >	<U00B0>	DEGREE SIGN
846	<+ - >	<U00B1>	PLUS-MINUS SIGN
847	<2S >	<U00B2>	SUPERSCRIP TWO
848	<3S >	<U00B3>	SUPERSCRIP THREE
849	<' ' >	<U00B4>	ACUTE ACCENT
850	<My >	<U00B5>	MICRO SIGN
851	<PI >	<U00B6>	PILCROW SIGN
852	<.M >	<U00B7>	MIDDLE DOT
853	<' , >	<U00B8>	CEDILLA

<1S>	<U00B9>	SUPERSCRIPIT 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
<AA>	<U00C5>	LATIN CAPITAL LETTER A WITH RING ABOVE
<AE>	<U00C6>	LATIN CAPITAL LETTER AE (ash)
<C,>	<U00C7>	LATIN CAPITAL LETTER C WITH CEDILLA
<E!>	<U00C8>	LATIN CAPITAL LETTER E WITH GRAVE
<E'>	<U00C9>	LATIN CAPITAL LETTER E WITH ACUTE
<E/>>	<U00CA>	LATIN CAPITAL LETTER E WITH CIRCUMFLEX
<E:>	<U00CB>	LATIN CAPITAL LETTER E WITH DIAERESIS
<I!>	<U00CC>	LATIN CAPITAL LETTER I WITH GRAVE
<I'>	<U00CD>	LATIN CAPITAL LETTER I WITH ACUTE
<I/>>	<U00CE>	LATIN CAPITAL LETTER I WITH CIRCUMFLEX
<I:>	<U00CF>	LATIN CAPITAL LETTER I WITH DIAERESIS
<D->	<U00D0>	LATIN CAPITAL LETTER ETH (Icelandic)
<N?>	<U00D1>	LATIN CAPITAL LETTER N WITH TILDE
<O!>	<U00D2>	LATIN CAPITAL LETTER O WITH GRAVE
<O'>	<U00D3>	LATIN CAPITAL LETTER O WITH ACUTE
<O/>>	<U00D4>	LATIN CAPITAL LETTER O WITH CIRCUMFLEX
<O?>	<U00D5>	LATIN CAPITAL LETTER O WITH TILDE
<O:>	<U00D6>	LATIN CAPITAL LETTER O WITH DIAERESIS
<*X>	<U00D7>	MULTIPLICATION SIGN
<O//>	<U00D8>	LATIN CAPITAL LETTER O WITH STROKE
<U!>	<U00D9>	LATIN CAPITAL LETTER U WITH GRAVE
<U'>	<U00DA>	LATIN CAPITAL LETTER U WITH ACUTE
<U/>>	<U00DB>	LATIN CAPITAL LETTER U WITH CIRCUMFLEX
<U:>	<U00DC>	LATIN CAPITAL LETTER U WITH DIAERESIS
<Y'>	<U00DD>	LATIN CAPITAL LETTER Y WITH ACUTE
<TH>	<U00DE>	LATIN CAPITAL LETTER THORN (Icelandic)
<ss>	<U00DF>	LATIN SMALL LETTER SHARP S (German)
<a!>	<U00E0>	LATIN SMALL LETTER A WITH GRAVE
<a'>	<U00E1>	LATIN SMALL LETTER A WITH ACUTE
<a/>>	<U00E2>	LATIN SMALL LETTER A WITH CIRCUMFLEX
<a?>	<U00E3>	LATIN SMALL LETTER A WITH TILDE
<a:>	<U00E4>	LATIN SMALL LETTER A WITH DIAERESIS
<aa>	<U00E5>	LATIN SMALL LETTER A WITH RING ABOVE
<ae>	<U00E6>	LATIN SMALL LETTER AE (ash)
<c,>	<U00E7>	LATIN SMALL LETTER C WITH CEDILLA
<e!>	<U00E8>	LATIN SMALL LETTER E WITH GRAVE
<e'>	<U00E9>	LATIN SMALL LETTER E WITH ACUTE
<e/>>	<U00EA>	LATIN SMALL LETTER E WITH CIRCUMFLEX
<e:>	<U00EB>	LATIN SMALL LETTER E WITH DIAERESIS
<i!>	<U00EC>	LATIN SMALL LETTER I WITH GRAVE
<i'>	<U00ED>	LATIN SMALL LETTER I WITH ACUTE
<i/>>	<U00EE>	LATIN SMALL LETTER I WITH CIRCUMFLEX
<i:>	<U00EF>	LATIN SMALL LETTER I WITH DIAERESIS
<d->	<U00F0>	LATIN SMALL LETTER ETH (Icelandic)
<n?>	<U00F1>	LATIN SMALL LETTER N WITH TILDE
<o!>	<U00F2>	LATIN SMALL LETTER O WITH GRAVE
<o'>	<U00F3>	LATIN SMALL LETTER O WITH ACUTE
<o/>>	<U00F4>	LATIN SMALL LETTER O WITH CIRCUMFLEX
<o?>	<U00F5>	LATIN SMALL LETTER O WITH TILDE
<o:>	<U00F6>	LATIN SMALL LETTER O WITH DIAERESIS
<-:>	<U00F7>	DIVISION SIGN
<o//>	<U00F8>	LATIN SMALL LETTER O WITH STROKE
<u!>	<U00F9>	LATIN SMALL LETTER U WITH GRAVE
<u'>	<U00FA>	LATIN SMALL LETTER U WITH ACUTE
<u/>>	<U00FB>	LATIN SMALL LETTER U WITH CIRCUMFLEX
<u:>	<U00FC>	LATIN SMALL LETTER U WITH DIAERESIS
<y'>	<U00FD>	LATIN SMALL LETTER Y WITH ACUTE
<th>	<U00FE>	LATIN SMALL LETTER THORN (Icelandic)
<y:>	<U00FF>	LATIN SMALL LETTER Y WITH DIAERESIS
<A->	<U0100>	LATIN CAPITAL LETTER A WITH MACRON
<a->	<U0101>	LATIN SMALL LETTER A WITH MACRON
<A>	<U0102>	LATIN CAPITAL LETTER A WITH BREVE
<a>	<U0103>	LATIN SMALL LETTER A WITH BREVE
<A>	<U0104>	LATIN CAPITAL LETTER A WITH OGONEK
<a; >	<U0105>	LATIN SMALL LETTER A WITH OGONEK
<C'>	<U0106>	LATIN CAPITAL LETTER C WITH ACUTE
<c'>	<U0107>	LATIN SMALL LETTER C WITH ACUTE
<C/>>	<U0108>	LATIN CAPITAL LETTER C WITH CIRCUMFLEX
<c/>>	<U0109>	LATIN SMALL LETTER C WITH CIRCUMFLEX
<C.>	<U010A>	LATIN CAPITAL LETTER C WITH DOT ABOVE
<c.>	<U010B>	LATIN SMALL LETTER C WITH DOT ABOVE
<C<>	<U010C>	LATIN CAPITAL LETTER C WITH CARON
<c<>	<U010D>	LATIN SMALL LETTER C WITH CARON
<D<>	<U010E>	LATIN CAPITAL LETTER D WITH CARON
<d<>	<U010F>	LATIN SMALL LETTER D WITH CARON
<D//>	<U0110>	LATIN CAPITAL LETTER D WITH STROKE

<d//>	<U0111>	LATIN SMALL LETTER D WITH STROKE
<E->	<U0112>	LATIN CAPITAL LETTER E WITH MACRON
<e->	<U0113>	LATIN SMALL LETTER E WITH MACRON
<E(>	<U0114>	LATIN CAPITAL LETTER E WITH BREVE
<e(>	<U0115>	LATIN SMALL LETTER E WITH BREVE
<E.>	<U0116>	LATIN CAPITAL LETTER E WITH DOT ABOVE
<e.>	<U0117>	LATIN SMALL LETTER E WITH DOT ABOVE
<E; >	<U0118>	LATIN CAPITAL LETTER E WITH OGONEK
<e; >	<U0119>	LATIN SMALL LETTER E WITH OGONEK
<E<>	<U011A>	LATIN CAPITAL LETTER E WITH CARON
<e<>	<U011B>	LATIN SMALL LETTER E WITH CARON
<G/>>	<U011C>	LATIN CAPITAL LETTER G WITH CIRCUMFLEX
<g/>>	<U011D>	LATIN SMALL LETTER G WITH CIRCUMFLEX
<G(>	<U011E>	LATIN CAPITAL LETTER G WITH BREVE
<g(>	<U011F>	LATIN SMALL LETTER G WITH BREVE
<G.>	<U0120>	LATIN CAPITAL LETTER G WITH DOT ABOVE
<g.>	<U0121>	LATIN SMALL LETTER G WITH DOT ABOVE
<G.>	<U0122>	LATIN CAPITAL LETTER G WITH CEDILLA
<g.>	<U0123>	LATIN SMALL LETTER G WITH CEDILLA
<H/>>	<U0124>	LATIN CAPITAL LETTER H WITH CIRCUMFLEX
<h/>>	<U0125>	LATIN SMALL LETTER H WITH CIRCUMFLEX
<H//>	<U0126>	LATIN CAPITAL LETTER H WITH STROKE
<h//>	<U0127>	LATIN SMALL LETTER H WITH STROKE
<I?>	<U0128>	LATIN CAPITAL LETTER I WITH TILDE
<i?>	<U0129>	LATIN SMALL LETTER I WITH TILDE
<I->	<U012A>	LATIN CAPITAL LETTER I WITH MACRON
<i->	<U012B>	LATIN SMALL LETTER I WITH MACRON
<I(>	<U012C>	LATIN CAPITAL LETTER I WITH BREVE
<i(>	<U012D>	LATIN SMALL LETTER I WITH BREVE
<I; >	<U012E>	LATIN CAPITAL LETTER I WITH OGONEK
<i; >	<U012F>	LATIN SMALL LETTER I WITH OGONEK
<I.>	<U0130>	LATIN CAPITAL LETTER I WITH DOT ABOVE
<i.>	<U0131>	LATIN SMALL LETTER DOTLESS I
<IJ>	<U0132>	LATIN CAPITAL LIGATURE IJ
<ij>	<U0133>	LATIN SMALL LIGATURE IJ
<J/>>	<U0134>	LATIN CAPITAL LETTER J WITH CIRCUMFLEX
<j/>>	<U0135>	LATIN SMALL LETTER J WITH CIRCUMFLEX
<K.>	<U0136>	LATIN CAPITAL LETTER K WITH CEDILLA
<k.>	<U0137>	LATIN SMALL LETTER K WITH CEDILLA
<kk>	<U0138>	LATIN SMALL LETTER KRA (Greenlandic)
<L'>	<U0139>	LATIN CAPITAL LETTER L WITH ACUTE
<l'>	<U013A>	LATIN SMALL LETTER L WITH ACUTE
<L.>	<U013B>	LATIN CAPITAL LETTER L WITH CEDILLA
<l.>	<U013C>	LATIN SMALL LETTER L WITH CEDILLA
<L<>	<U013D>	LATIN CAPITAL LETTER L WITH CARON
<l<>	<U013E>	LATIN SMALL LETTER L WITH CARON
<L.>	<U013F>	LATIN CAPITAL LETTER L WITH MIDDLE DOT
<l.>	<U0140>	LATIN SMALL LETTER L WITH MIDDLE DOT
<L//>	<U0141>	LATIN CAPITAL LETTER L WITH STROKE
<l//>	<U0142>	LATIN SMALL LETTER L WITH STROKE
<N'>	<U0143>	LATIN CAPITAL LETTER N WITH ACUTE
<n'>	<U0144>	LATIN SMALL LETTER N WITH ACUTE
<N.>	<U0145>	LATIN CAPITAL LETTER N WITH CEDILLA
<n.>	<U0146>	LATIN SMALL LETTER N WITH CEDILLA
<N<>	<U0147>	LATIN CAPITAL LETTER N WITH CARON
<n<>	<U0148>	LATIN SMALL LETTER N WITH CARON
<'n>	<U0149>	LATIN SMALL LETTER N PRECEDED BY APOSTROPHE
<NG>	<U014A>	LATIN CAPITAL LETTER ENG (Sami)
<ng>	<U014B>	LATIN SMALL LETTER ENG (Sami)
<O->	<U014C>	LATIN CAPITAL LETTER O WITH MACRON
<o->	<U014D>	LATIN SMALL LETTER O WITH MACRON
<O(>	<U014E>	LATIN CAPITAL LETTER O WITH BREVE
<o(>	<U014F>	LATIN SMALL LETTER O WITH BREVE
<O">	<U0150>	LATIN CAPITAL LETTER O WITH DOUBLE ACUTE
<o">	<U0151>	LATIN SMALL LETTER O WITH DOUBLE ACUTE
<OE>	<U0152>	LATIN CAPITAL LIGATURE OE
<oe>	<U0153>	LATIN SMALL LIGATURE OE
<R'>	<U0154>	LATIN CAPITAL LETTER R WITH ACUTE
<r'>	<U0155>	LATIN SMALL LETTER R WITH ACUTE
<R.>	<U0156>	LATIN CAPITAL LETTER R WITH CEDILLA
<r.>	<U0157>	LATIN SMALL LETTER R WITH CEDILLA
<R<>	<U0158>	LATIN CAPITAL LETTER R WITH CARON
<r<>	<U0159>	LATIN SMALL LETTER R WITH CARON
<S'>	<U015A>	LATIN CAPITAL LETTER S WITH ACUTE
<s'>	<U015B>	LATIN SMALL LETTER S WITH ACUTE
<S/>>	<U015C>	LATIN CAPITAL LETTER S WITH CIRCUMFLEX
<s/>>	<U015D>	LATIN SMALL LETTER S WITH CIRCUMFLEX
<S.>	<U015E>	LATIN CAPITAL LETTER S WITH CEDILLA
<s.>	<U015F>	LATIN SMALL LETTER S WITH CEDILLA
<S<>	<U0160>	LATIN CAPITAL LETTER S WITH CARON
<s<>	<U0161>	LATIN SMALL LETTER S WITH CARON
<T.>	<U0162>	LATIN CAPITAL LETTER T WITH CEDILLA
<t.>	<U0163>	LATIN SMALL LETTER T WITH CEDILLA
<T<>	<U0164>	LATIN CAPITAL LETTER T WITH CARON
<t<>	<U0165>	LATIN SMALL LETTER T WITH CARON
<T//>	<U0166>	LATIN CAPITAL LETTER T WITH STROKE
<t//>	<U0167>	LATIN SMALL LETTER T WITH STROKE
<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
4	<w/>>	<U0175>	LATIN SMALL LETTER W WITH CIRCUMFLEX
4	<Y/>>	<U0176>	LATIN CAPITAL LETTER Y WITH CIRCUMFLEX
4	<y/>>	<U0177>	LATIN SMALL LETTER Y WITH CIRCUMFLEX
4	<Y: >	<U0178>	LATIN CAPITAL LETTER Y WITH DIAERESIS
4	<Z' >	<U0179>	LATIN CAPITAL LETTER Z WITH ACUTE
4	<z' >	<U017A>	LATIN SMALL LETTER Z WITH ACUTE
4	<Z. >	<U017B>	LATIN CAPITAL LETTER Z WITH DOT ABOVE
4	<z. >	<U017C>	LATIN SMALL LETTER Z WITH DOT ABOVE
4	<Z< >	<U017D>	LATIN CAPITAL LETTER Z WITH CARON
4	<z< >	<U017E>	LATIN SMALL LETTER Z WITH CARON
4	<s1 >	<U017F>	LATIN SMALL LETTER LONG S
4	<b//>	<U0180>	LATIN SMALL LETTER B WITH STROKE
4	<B2 >	<U0181>	LATIN CAPITAL LETTER B WITH HOOK
4	<C2 >	<U0187>	LATIN CAPITAL LETTER C WITH HOOK
4	<c2 >	<U0188>	LATIN SMALL LETTER C WITH HOOK
4	<F2 >	<U0191>	LATIN CAPITAL LETTER F WITH HOOK
4	<f2 >	<U0192>	LATIN SMALL LETTER F WITH HOOK
4	<K2 >	<U0198>	LATIN CAPITAL LETTER K WITH HOOK
4	<k2 >	<U0199>	LATIN SMALL LETTER K WITH HOOK
4	<O9 >	<U01A0>	LATIN CAPITAL LETTER O WITH HORN
4	<o9 >	<U01A1>	LATIN SMALL LETTER O WITH HORN
4	<OI >	<U01A2>	LATIN CAPITAL LETTER OI
4	<oi >	<U01A3>	LATIN SMALL LETTER OI
4	<yr >	<U01A6>	LATIN LETTER YR
4	<U9 >	<U01AF>	LATIN CAPITAL LETTER U WITH HORN
4	<u9 >	<U01B0>	LATIN SMALL LETTER U WITH HORN
4	<Z//>	<U01B5>	LATIN CAPITAL LETTER Z WITH STROKE
4	<z//>	<U01B6>	LATIN SMALL LETTER Z WITH STROKE
4	<ED >	<U01B7>	LATIN CAPITAL LETTER EZH
4	<DZ< >	<U01C4>	LATIN CAPITAL LETTER DZ WITH CARON
4	<Dz< >	<U01C5>	LATIN CAPITAL LETTER D WITH SMALL LETTER Z WITH CARON
4	<dz< >	<U01C6>	LATIN SMALL LETTER DZ WITH CARON
4	<LJ3 >	<U01C7>	LATIN CAPITAL LETTER LJ
4	<Lj3 >	<U01C8>	LATIN CAPITAL LETTER L WITH SMALL LETTER J
4	<lj3 >	<U01C9>	LATIN SMALL LETTER LJ
4	<NJ3 >	<U01CA>	LATIN CAPITAL LETTER NJ
4	<Nj3 >	<U01CB>	LATIN CAPITAL LETTER N WITH SMALL LETTER J
4	<nj3 >	<U01CC>	LATIN SMALL LETTER NJ
4	<A< >	<U01CD>	LATIN CAPITAL LETTER A WITH CARON
4	<a< >	<U01CE>	LATIN SMALL LETTER A WITH CARON
4	<I< >	<U01CF>	LATIN CAPITAL LETTER I WITH CARON
4	<i< >	<U01D0>	LATIN SMALL LETTER I WITH CARON
4	<O< >	<U01D1>	LATIN CAPITAL LETTER O WITH CARON
4	<o< >	<U01D2>	LATIN SMALL LETTER O WITH CARON
4	<U< >	<U01D3>	LATIN CAPITAL LETTER U WITH CARON
4	<u< >	<U01D4>	LATIN SMALL LETTER U WITH CARON
4	<U: ->	<U01D5>	LATIN CAPITAL LETTER U WITH DIAERESIS AND MACRON
4	<u: ->	<U01D6>	LATIN SMALL LETTER U WITH DIAERESIS AND MACRON
4	<U: ' >	<U01D7>	LATIN CAPITAL LETTER U WITH DIAERESIS AND ACUTE
4	<u: ' >	<U01D8>	LATIN SMALL LETTER U WITH DIAERESIS AND ACUTE
4	<U: < >	<U01D9>	LATIN CAPITAL LETTER U WITH DIAERESIS AND CARON
4	<u: < >	<U01DA>	LATIN SMALL LETTER U WITH DIAERESIS AND CARON
4	<U: ! >	<U01DB>	LATIN CAPITAL LETTER U WITH DIAERESIS AND GRAVE
4	<u: ! >	<U01DC>	LATIN SMALL LETTER U WITH DIAERESIS AND GRAVE
4	<e1 >	<U01DD>	LATIN SMALL LETTER TURNED E
4	<A1 >	<U01DE>	LATIN CAPITAL LETTER A WITH DIAERESIS AND MACRON
4	<a1 >	<U01DF>	LATIN SMALL LETTER A WITH DIAERESIS AND MACRON
4	<A7 >	<U01E0>	LATIN CAPITAL LETTER A WITH DOT ABOVE AND MACRON
4	<a7 >	<U01E1>	LATIN SMALL LETTER A WITH DOT ABOVE AND MACRON
4	<A3 >	<U01E2>	LATIN CAPITAL LETTER AE WITH MACRON (ash)
4	<a3 >	<U01E3>	LATIN SMALL LETTER AE WITH MACRON (ash)
4	<G//>	<U01E4>	LATIN CAPITAL LETTER G WITH STROKE
4	<g//>	<U01E5>	LATIN SMALL LETTER G WITH STROKE
4	<G< >	<U01E6>	LATIN CAPITAL LETTER G WITH CARON
4	<g< >	<U01E7>	LATIN SMALL LETTER G WITH CARON
4	<K< >	<U01E8>	LATIN CAPITAL LETTER K WITH CARON
4	<k< >	<U01E9>	LATIN SMALL LETTER K WITH CARON
4	<O; >	<U01EA>	LATIN CAPITAL LETTER O WITH OGONEK
4	<o; >	<U01EB>	LATIN SMALL LETTER O WITH OGONEK
4	<O1 >	<U01EC>	LATIN CAPITAL LETTER O WITH OGONEK AND MACRON
4	<o1 >	<U01ED>	LATIN SMALL LETTER O WITH OGONEK AND MACRON
4	<EZ >	<U01EE>	LATIN CAPITAL LETTER EZH WITH CARON
4	<ez >	<U01EF>	LATIN SMALL LETTER EZH WITH CARON
4	<j< >	<U01F0>	LATIN SMALL LETTER J WITH CARON
4	<DZ3 >	<U01F1>	LATIN CAPITAL LETTER DZ
4	<Dz3 >	<U01F2>	LATIN CAPITAL LETTER D WITH SMALL LETTER Z

<dz3>	<U01F3>	LATIN SMALL LETTER DZ
<G'>	<U01F4>	LATIN CAPITAL LETTER G WITH ACUTE
<g'>	<U01F5>	LATIN SMALL LETTER G WITH ACUTE
<AA'>	<U01FA>	LATIN CAPITAL LETTER A WITH RING ABOVE AND ACUTE
<aa'>	<U01FB>	LATIN SMALL LETTER A WITH RING ABOVE AND ACUTE
<AE'>	<U01FC>	LATIN CAPITAL LETTER AE WITH ACUTE (ash)
<ae'>	<U01FD>	LATIN SMALL LETTER AE WITH ACUTE (ash)
<O'/'>	<U01FE>	LATIN CAPITAL LETTER O WITH STROKE AND ACUTE
<o'/'>	<U01FF>	LATIN SMALL LETTER O WITH STROKE AND ACUTE
<A!!>	<U0200>	LATIN CAPITAL LETTER A WITH DOUBLE GRAVE
<a!!>	<U0201>	LATIN SMALL LETTER A WITH DOUBLE GRAVE
<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)
<1->	<U02C9>	MODIFIER LETTER MACRON (Mandarin Chinese first tone)
<1!>	<U02CB>	MODIFIER LETTER GRAVE ACCENT (Mandarin Chinese fourth tone)
<'(>	<U02D8>	BREVE
<' .>	<U02D9>	DOT ABOVE (Mandarin Chinese light tone)
<' 0>	<U02DA>	RING ABOVE
<' ;>	<U02DB>	OGONEK
<1?>	<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

4206	<e%>	<U03AD>	GREEK SMALL LETTER EPSILON WITH TONOS
4207	<y%>	<U03AE>	GREEK SMALL LETTER ETA WITH TONOS
4208	<i%>	<U03AF>	GREEK SMALL LETTER IOTA WITH TONOS
4209	<u3>	<U03B0>	GREEK SMALL LETTER UPSILON WITH DIALYTIKA AND TONOS
4210	<a*>	<U03B1>	GREEK SMALL LETTER ALPHA
4211	<b*>	<U03B2>	GREEK SMALL LETTER BETA
4212	<g*>	<U03B3>	GREEK SMALL LETTER GAMMA
4213	<d*>	<U03B4>	GREEK SMALL LETTER DELTA
4214	<e*>	<U03B5>	GREEK SMALL LETTER EPSILON
4215	<z*>	<U03B6>	GREEK SMALL LETTER ZETA
4216	<y*>	<U03B7>	GREEK SMALL LETTER ETA
4217	<h*>	<U03B8>	GREEK SMALL LETTER THETA
4218	<i*>	<U03B9>	GREEK SMALL LETTER IOTA
4219	<k*>	<U03BA>	GREEK SMALL LETTER KAPPA
4220	<l*>	<U03BB>	GREEK SMALL LETTER LAMDA
4221	<m*>	<U03BC>	GREEK SMALL LETTER MU
4222	<n*>	<U03BD>	GREEK SMALL LETTER NU
4223	<c*>	<U03BE>	GREEK SMALL LETTER XI
4224	<o*>	<U03BF>	GREEK SMALL LETTER OMICRON
4225	<p*>	<U03C0>	GREEK SMALL LETTER PI
4226	<r*>	<U03C1>	GREEK SMALL LETTER RHO
4227	<*s>	<U03C2>	GREEK SMALL LETTER FINAL SIGMA
4228	<s*>	<U03C3>	GREEK SMALL LETTER SIGMA
4229	<t*>	<U03C4>	GREEK SMALL LETTER TAU
4230	<u*>	<U03C5>	GREEK SMALL LETTER UPSILON
4231	<f*>	<U03C6>	GREEK SMALL LETTER PHI
4232	<x*>	<U03C7>	GREEK SMALL LETTER CHI
4233	<q*>	<U03C8>	GREEK SMALL LETTER PSI
4234	<w*>	<U03C9>	GREEK SMALL LETTER OMEGA
4235	<j*>	<U03CA>	GREEK SMALL LETTER IOTA WITH DIALYTIKA
4236	<v*>	<U03CB>	GREEK SMALL LETTER UPSILON WITH DIALYTIKA
4237	<o%>	<U03CC>	GREEK SMALL LETTER OMICRON WITH TONOS
4238	<u%>	<U03CD>	GREEK SMALL LETTER UPSILON WITH TONOS
4239	<w%>	<U03CE>	GREEK SMALL LETTER OMEGA WITH TONOS
4240	<b3>	<U03D0>	GREEK BETA SYMBOL
4241	<T3>	<U03DA>	GREEK LETTER STIGMA
4242	<M3>	<U03DC>	GREEK LETTER DIGAMMA
4243	<K3>	<U03DE>	GREEK LETTER KOPPA
4244	<P3>	<U03E0>	GREEK LETTER SAMPI
4245	<IO>	<U0401>	CYRILLIC CAPITAL LETTER IO
4246	<D%>	<U0402>	CYRILLIC CAPITAL LETTER DJE (Serbocroatian)
4247	<G%>	<U0403>	CYRILLIC CAPITAL LETTER GJE
4248	<IE>	<U0404>	CYRILLIC CAPITAL LETTER UKRAINIAN IE
4249	<DS>	<U0405>	CYRILLIC CAPITAL LETTER DZE
4250	<II>	<U0406>	CYRILLIC CAPITAL LETTER BYELORUSSIAN-UKRAINIAN I
4251	<YI>	<U0407>	CYRILLIC CAPITAL LETTER YI (Ukrainian)
4252	<J%>	<U0408>	CYRILLIC CAPITAL LETTER JE
4253	<LJ>	<U0409>	CYRILLIC CAPITAL LETTER LJE
4254	<NJ>	<U040A>	CYRILLIC CAPITAL LETTER NJE
4255	<Ts>	<U040B>	CYRILLIC CAPITAL LETTER TSHE (Serbocroatian)
4256	<KJ>	<U040C>	CYRILLIC CAPITAL LETTER KJE
4257	<V%>	<U040E>	CYRILLIC CAPITAL LETTER SHORT U (Byelorussian)
4258	<DZ>	<U040F>	CYRILLIC CAPITAL LETTER DZHE
4259	<A=>	<U0410>	CYRILLIC CAPITAL LETTER A
4260	<B=>	<U0411>	CYRILLIC CAPITAL LETTER BE
4261	<V=>	<U0412>	CYRILLIC CAPITAL LETTER VE
4262	<G=>	<U0413>	CYRILLIC CAPITAL LETTER GHE
4263	<D=>	<U0414>	CYRILLIC CAPITAL LETTER DE
4264	<E=>	<U0415>	CYRILLIC CAPITAL LETTER IE
4265	<Z%>	<U0416>	CYRILLIC CAPITAL LETTER ZHE
4266	<Z=>	<U0417>	CYRILLIC CAPITAL LETTER ZE
4267	<I=>	<U0418>	CYRILLIC CAPITAL LETTER I
4268	<J=>	<U0419>	CYRILLIC CAPITAL LETTER SHORT I
4269	<K=>	<U041A>	CYRILLIC CAPITAL LETTER KA
4270	<L=>	<U041B>	CYRILLIC CAPITAL LETTER EL
4271	<M=>	<U041C>	CYRILLIC CAPITAL LETTER EM
4272	<N=>	<U041D>	CYRILLIC CAPITAL LETTER EN
4273	<O=>	<U041E>	CYRILLIC CAPITAL LETTER O
4274	<P=>	<U041F>	CYRILLIC CAPITAL LETTER PE
4275	<R=>	<U0420>	CYRILLIC CAPITAL LETTER ER
4276	<S=>	<U0421>	CYRILLIC CAPITAL LETTER ES
4277	<T=>	<U0422>	CYRILLIC CAPITAL LETTER TE
4278	<U=>	<U0423>	CYRILLIC CAPITAL LETTER U
4279	<F=>	<U0424>	CYRILLIC CAPITAL LETTER EF
4280	<H=>	<U0425>	CYRILLIC CAPITAL LETTER HA
4281	<C=>	<U0426>	CYRILLIC CAPITAL LETTER TSE
4282	<C%>	<U0427>	CYRILLIC CAPITAL LETTER CHE
4283	<S%>	<U0428>	CYRILLIC CAPITAL LETTER SHA
4284	<Sc>	<U0429>	CYRILLIC CAPITAL LETTER SHCHA
4285	<=">	<U042A>	CYRILLIC CAPITAL LETTER HARD SIGN
4286	<Y=>	<U042B>	CYRILLIC CAPITAL LETTER YERU
4287	<% ">	<U042C>	CYRILLIC CAPITAL LETTER SOFT SIGN
4288	<JE>	<U042D>	CYRILLIC CAPITAL LETTER E
4289	<JU>	<U042E>	CYRILLIC CAPITAL LETTER YU
4290	<JA>	<U042F>	CYRILLIC CAPITAL LETTER YA
4291	<a=>	<U0430>	CYRILLIC SMALL LETTER A
4292	<b=>	<U0431>	CYRILLIC SMALL LETTER BE
4293	<v=>	<U0432>	CYRILLIC SMALL LETTER VE



<g=>	<U0433>	CYRILLIC SMALL LETTER GHE
<d=>	<U0434>	CYRILLIC SMALL LETTER DE
<e=>	<U0435>	CYRILLIC SMALL LETTER IE
<z%>	<U0436>	CYRILLIC SMALL LETTER ZHE
<z=>	<U0437>	CYRILLIC SMALL LETTER ZE
<i=>	<U0438>	CYRILLIC SMALL LETTER I
<j=>	<U0439>	CYRILLIC SMALL LETTER SHORT I
<k=>	<U043A>	CYRILLIC SMALL LETTER KA
<l=>	<U043B>	CYRILLIC SMALL LETTER EL
<m=>	<U043C>	CYRILLIC SMALL LETTER EM
<n=>	<U043D>	CYRILLIC SMALL LETTER EN
<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 SUPERScript 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

44	<e, />	<U1E1D>	LATIN SMALL LETTER E WITH CEDILLA AND BREVE
44	<F.>	<U1E1E>	LATIN CAPITAL LETTER F WITH DOT ABOVE
44	<f.>	<U1E1F>	LATIN SMALL LETTER F WITH DOT ABOVE
44	<G->	<U1E20>	LATIN CAPITAL LETTER G WITH MACRON
44	<g->	<U1E21>	LATIN SMALL LETTER G WITH MACRON
44	<H.>	<U1E22>	LATIN CAPITAL LETTER H WITH DOT ABOVE
44	<h.>	<U1E23>	LATIN SMALL LETTER H WITH DOT ABOVE
44	<H-.>	<U1E24>	LATIN CAPITAL LETTER H WITH DOT BELOW
44	<h-.>	<U1E25>	LATIN SMALL LETTER H WITH DOT BELOW
44	<H:>	<U1E26>	LATIN CAPITAL LETTER H WITH DIAERESIS
44	<h:>	<U1E27>	LATIN SMALL LETTER H WITH DIAERESIS
44	<H, >	<U1E28>	LATIN CAPITAL LETTER H WITH CEDILLA
44	<h, >	<U1E29>	LATIN SMALL LETTER H WITH CEDILLA
44	<H-(>	<U1E2A>	LATIN CAPITAL LETTER H WITH BREVE BELOW
44	<h-(>	<U1E2B>	LATIN SMALL LETTER H WITH BREVE BELOW
44	<I-?>	<U1E2C>	LATIN CAPITAL LETTER I WITH TILDE BELOW
44	<i-?>	<U1E2D>	LATIN SMALL LETTER I WITH TILDE BELOW
44	<I:'>	<U1E2E>	LATIN CAPITAL LETTER I WITH DIAERESIS AND ACUTE
44	<i:'>	<U1E2F>	LATIN SMALL LETTER I WITH DIAERESIS AND ACUTE
44	<K'>	<U1E30>	LATIN CAPITAL LETTER K WITH ACUTE
44	<k'>	<U1E31>	LATIN SMALL LETTER K WITH ACUTE
44	<K-.>	<U1E32>	LATIN CAPITAL LETTER K WITH DOT BELOW
44	<k-.>	<U1E33>	LATIN SMALL LETTER K WITH DOT BELOW
44	<K_>	<U1E34>	LATIN CAPITAL LETTER K WITH LINE BELOW
44	<k_>	<U1E35>	LATIN SMALL LETTER K WITH LINE BELOW
44	<L-.>	<U1E36>	LATIN CAPITAL LETTER L WITH DOT BELOW
44	<l-.>	<U1E37>	LATIN SMALL LETTER L WITH DOT BELOW
44	<L--.>	<U1E38>	LATIN CAPITAL LETTER L WITH DOT BELOW AND MACRON
44	<l--.>	<U1E39>	LATIN SMALL LETTER L WITH DOT BELOW AND MACRON
44	<L_>	<U1E3A>	LATIN CAPITAL LETTER L WITH LINE BELOW
44	<l_>	<U1E3B>	LATIN SMALL LETTER L WITH LINE BELOW
44	<L-/>>	<U1E3C>	LATIN CAPITAL LETTER L WITH CIRCUMFLEX BELOW
44	<l-/>>	<U1E3D>	LATIN SMALL LETTER L WITH CIRCUMFLEX BELOW
44	<M'>	<U1E3E>	LATIN CAPITAL LETTER M WITH ACUTE
44	<m'>	<U1E3F>	LATIN SMALL LETTER M WITH ACUTE
44	<M.>	<U1E40>	LATIN CAPITAL LETTER M WITH DOT ABOVE
44	<m.>	<U1E41>	LATIN SMALL LETTER M WITH DOT ABOVE
44	<M-.>	<U1E42>	LATIN CAPITAL LETTER M WITH DOT BELOW
44	<m-.>	<U1E43>	LATIN SMALL LETTER M WITH DOT BELOW
44	<N.>	<U1E44>	LATIN CAPITAL LETTER N WITH DOT ABOVE
44	<n.>	<U1E45>	LATIN SMALL LETTER N WITH DOT ABOVE
44	<N-.>	<U1E46>	LATIN CAPITAL LETTER N WITH DOT BELOW
44	<n-.>	<U1E47>	LATIN SMALL LETTER N WITH DOT BELOW
44	<N_>	<U1E48>	LATIN CAPITAL LETTER N WITH LINE BELOW
44	<n_>	<U1E49>	LATIN SMALL LETTER N WITH LINE BELOW
44	<N-/>>	<U1E4A>	LATIN CAPITAL LETTER N WITH CIRCUMFLEX BELOW
44	<n-/>>	<U1E4B>	LATIN SMALL LETTER N WITH CIRCUMFLEX BELOW
44	<O?>'	<U1E4C>	LATIN CAPITAL LETTER O WITH TILDE AND ACUTE
44	<o?>'	<U1E4D>	LATIN SMALL LETTER O WITH TILDE AND ACUTE
44	<O?:>	<U1E4E>	LATIN CAPITAL LETTER O WITH TILDE AND DIAERESIS
44	<o?:>	<U1E4F>	LATIN SMALL LETTER O WITH TILDE AND DIAERESIS
44	<O-!>	<U1E50>	LATIN CAPITAL LETTER O WITH MACRON AND GRAVE
44	<o-!>	<U1E51>	LATIN SMALL LETTER O WITH MACRON AND GRAVE
44	<O-'>	<U1E52>	LATIN CAPITAL LETTER O WITH MACRON AND ACUTE
44	<o-'>	<U1E53>	LATIN SMALL LETTER O WITH MACRON AND ACUTE
44	<P'>	<U1E54>	LATIN CAPITAL LETTER P WITH ACUTE
44	<p'>	<U1E55>	LATIN SMALL LETTER P WITH ACUTE
44	<P.>	<U1E56>	LATIN CAPITAL LETTER P WITH DOT ABOVE
44	<p.>	<U1E57>	LATIN SMALL LETTER P WITH DOT ABOVE
44	<R.>	<U1E58>	LATIN CAPITAL LETTER R WITH DOT ABOVE
44	<r.>	<U1E59>	LATIN SMALL LETTER R WITH DOT ABOVE
44	<R-.>	<U1E5A>	LATIN CAPITAL LETTER R WITH DOT BELOW
44	<r-.>	<U1E5B>	LATIN SMALL LETTER R WITH DOT BELOW
44	<R--.>	<U1E5C>	LATIN CAPITAL LETTER R WITH DOT BELOW AND MACRON
44	<r--.>	<U1E5D>	LATIN SMALL LETTER R WITH DOT BELOW AND MACRON
44	<R_>	<U1E5E>	LATIN CAPITAL LETTER R WITH LINE BELOW
44	<r_>	<U1E5F>	LATIN SMALL LETTER R WITH LINE BELOW
44	<S.>	<U1E60>	LATIN CAPITAL LETTER S WITH DOT ABOVE
44	<s.>	<U1E61>	LATIN SMALL LETTER S WITH DOT ABOVE
44	<S-.>	<U1E62>	LATIN CAPITAL LETTER S WITH DOT BELOW
44	<s-.>	<U1E63>	LATIN SMALL LETTER S WITH DOT BELOW
44	<S' .>	<U1E64>	LATIN CAPITAL LETTER S WITH ACUTE AND DOT ABOVE
44	<s' .>	<U1E65>	LATIN SMALL LETTER S WITH ACUTE AND DOT ABOVE
44	<S<.>	<U1E66>	LATIN CAPITAL LETTER S WITH CARON AND DOT ABOVE
44	<s<.>	<U1E67>	LATIN SMALL LETTER S WITH CARON AND DOT ABOVE
44	<S-.-.>	<U1E68>	LATIN CAPITAL LETTER S WITH DOT BELOW AND DOT ABOVE
44	<s-.-.>	<U1E69>	LATIN SMALL LETTER S WITH DOT BELOW AND DOT ABOVE
44	<T.>	<U1E6A>	LATIN CAPITAL LETTER T WITH DOT ABOVE
44	<t.>	<U1E6B>	LATIN SMALL LETTER T WITH DOT ABOVE
44	<T-.>	<U1E6C>	LATIN CAPITAL LETTER T WITH DOT BELOW
44	<t-.>	<U1E6D>	LATIN SMALL LETTER T WITH DOT BELOW
44	<T_>	<U1E6E>	LATIN CAPITAL LETTER T WITH LINE BELOW
44	<t_>	<U1E6F>	LATIN SMALL LETTER T WITH LINE BELOW
44	<T-/>>	<U1E70>	LATIN CAPITAL LETTER T WITH CIRCUMFLEX BELOW
44	<t-/>>	<U1E71>	LATIN SMALL LETTER T WITH CIRCUMFLEX BELOW
44	<U--:>	<U1E72>	LATIN CAPITAL LETTER U WITH DIAERESIS BELOW
44	<u--:>	<U1E73>	LATIN SMALL LETTER U WITH DIAERESIS BELOW
44	<U-?>	<U1E74>	LATIN CAPITAL LETTER U WITH TILDE BELOW

<u-?>	<U1E75>	LATIN SMALL LETTER U WITH TILDE BELOW
<U-/>	<U1E76>	LATIN CAPITAL LETTER U WITH CIRCUMFLEX BELOW
<u-/>	<U1E77>	LATIN SMALL LETTER U WITH CIRCUMFLEX BELOW
<U?´>	<U1E78>	LATIN CAPITAL LETTER U WITH TILDE AND ACUTE
<u?´>	<U1E79>	LATIN SMALL LETTER U WITH TILDE AND ACUTE
<U-:~>	<U1E7A>	LATIN CAPITAL LETTER U WITH MACRON AND DIAERESIS
<u-:~>	<U1E7B>	LATIN SMALL LETTER U WITH MACRON AND DIAERESIS
<V?>	<U1E7C>	LATIN CAPITAL LETTER V WITH TILDE
<v?>	<U1E7D>	LATIN SMALL LETTER V WITH TILDE
<V-·>	<U1E7E>	LATIN CAPITAL LETTER V WITH DOT BELOW
<v-·>	<U1E7F>	LATIN SMALL LETTER V WITH DOT BELOW
<W!>	<U1E80>	LATIN CAPITAL LETTER W WITH GRAVE
<w!>	<U1E81>	LATIN SMALL LETTER W WITH GRAVE
<W´>	<U1E82>	LATIN CAPITAL LETTER W WITH ACUTE
<w´>	<U1E83>	LATIN SMALL LETTER W WITH ACUTE
<W:~>	<U1E84>	LATIN CAPITAL LETTER W WITH DIAERESIS
<w:~>	<U1E85>	LATIN SMALL LETTER W WITH DIAERESIS
<W-·>	<U1E86>	LATIN CAPITAL LETTER W WITH DOT ABOVE
<w-·>	<U1E87>	LATIN SMALL LETTER W WITH DOT ABOVE
<W-·~>	<U1E88>	LATIN CAPITAL LETTER W WITH DOT BELOW
<w-·~>	<U1E89>	LATIN SMALL LETTER W WITH DOT BELOW
<X·>	<U1E8A>	LATIN CAPITAL LETTER X WITH DOT ABOVE
<x·>	<U1E8B>	LATIN SMALL LETTER X WITH DOT ABOVE
<X:~>	<U1E8C>	LATIN CAPITAL LETTER X WITH DIAERESIS
<x:~>	<U1E8D>	LATIN SMALL LETTER X WITH DIAERESIS
<Y·>	<U1E8E>	LATIN CAPITAL LETTER Y WITH DOT ABOVE
<y·>	<U1E8F>	LATIN SMALL LETTER Y WITH DOT ABOVE
<Z/>>	<U1E90>	LATIN CAPITAL LETTER Z WITH CIRCUMFLEX
<z/>>	<U1E91>	LATIN SMALL LETTER Z WITH CIRCUMFLEX
<Z-·>	<U1E92>	LATIN CAPITAL LETTER Z WITH DOT BELOW
<z-·>	<U1E93>	LATIN SMALL LETTER Z WITH DOT BELOW
<Z-·~>	<U1E94>	LATIN CAPITAL LETTER Z WITH LINE BELOW
<z-·~>	<U1E95>	LATIN SMALL LETTER Z WITH LINE BELOW
<A-·>	<U1EA0>	LATIN CAPITAL LETTER A WITH DOT BELOW
<a-·>	<U1EA1>	LATIN SMALL LETTER A WITH DOT BELOW
<A2>	<U1EA2>	LATIN CAPITAL LETTER A WITH HOOK ABOVE
<a2>	<U1EA3>	LATIN SMALL LETTER A WITH HOOK ABOVE
<A/>´>	<U1EA4>	LATIN CAPITAL LETTER A WITH CIRCUMFLEX AND ACUTE
<a/>´>	<U1EA5>	LATIN SMALL LETTER A WITH CIRCUMFLEX AND ACUTE
<A/>!>	<U1EA6>	LATIN CAPITAL LETTER A WITH CIRCUMFLEX AND GRAVE
<a/>!>	<U1EA7>	LATIN SMALL LETTER A WITH CIRCUMFLEX AND GRAVE
<A/>2>	<U1EA8>	LATIN CAPITAL LETTER A WITH CIRCUMFLEX AND HOOK ABOVE
<a/>2>	<U1EA9>	LATIN SMALL LETTER A WITH CIRCUMFLEX AND HOOK ABOVE
<A/>?>	<U1EAA>	LATIN CAPITAL LETTER A WITH CIRCUMFLEX AND TILDE
<a/>?>	<U1EAB>	LATIN SMALL LETTER A WITH CIRCUMFLEX AND TILDE
<A/>-·>	<U1EAC>	LATIN CAPITAL LETTER A WITH CIRCUMFLEX AND DOT BELOW
<a/>-·>	<U1EAD>	LATIN SMALL LETTER A WITH CIRCUMFLEX AND DOT BELOW
<A´>	<U1EAE>	LATIN CAPITAL LETTER A WITH BREVE AND ACUTE
<a´>	<U1EAF>	LATIN SMALL LETTER A WITH BREVE AND ACUTE
<A!>	<U1EB0>	LATIN CAPITAL LETTER A WITH BREVE AND GRAVE
<a!>	<U1EB1>	LATIN SMALL LETTER A WITH BREVE AND GRAVE
<A(2)>	<U1EB2>	LATIN CAPITAL LETTER A WITH BREVE AND HOOK ABOVE
<a(2)>	<U1EB3>	LATIN SMALL LETTER A WITH BREVE AND HOOK ABOVE
<A(?)>	<U1EB4>	LATIN CAPITAL LETTER A WITH BREVE AND TILDE
<a(?)>	<U1EB5>	LATIN SMALL LETTER A WITH BREVE AND TILDE
<A(-·>	<U1EB6>	LATIN CAPITAL LETTER A WITH BREVE AND DOT BELOW
<a(-·>	<U1EB7>	LATIN SMALL LETTER A WITH BREVE AND DOT BELOW
<E-·>	<U1EB8>	LATIN CAPITAL LETTER E WITH DOT BELOW
<e-·>	<U1EB9>	LATIN SMALL LETTER E WITH DOT BELOW
<E2>	<U1EBA>	LATIN CAPITAL LETTER E WITH HOOK ABOVE
<e2>	<U1EBB>	LATIN SMALL LETTER E WITH HOOK ABOVE
<E?>	<U1EBC>	LATIN CAPITAL LETTER E WITH TILDE
<e?>	<U1EBD>	LATIN SMALL LETTER E WITH TILDE
<E/>´>	<U1EBE>	LATIN CAPITAL LETTER E WITH CIRCUMFLEX AND ACUTE
<e/>´>	<U1EBF>	LATIN SMALL LETTER E WITH CIRCUMFLEX AND ACUTE
<E/>!>	<U1EC0>	LATIN CAPITAL LETTER E WITH CIRCUMFLEX AND GRAVE
<e/>!>	<U1EC1>	LATIN SMALL LETTER E WITH CIRCUMFLEX AND GRAVE
<E/>2>	<U1EC2>	LATIN CAPITAL LETTER E WITH CIRCUMFLEX AND HOOK ABOVE
<e/>2>	<U1EC3>	LATIN SMALL LETTER E WITH CIRCUMFLEX AND HOOK ABOVE
<E/>?>	<U1EC4>	LATIN CAPITAL LETTER E WITH CIRCUMFLEX AND TILDE
<e/>?>	<U1EC5>	LATIN SMALL LETTER E WITH CIRCUMFLEX AND TILDE
<E/>-·>	<U1EC6>	LATIN CAPITAL LETTER E WITH CIRCUMFLEX AND DOT BELOW
<e/>-·>	<U1EC7>	LATIN SMALL LETTER E WITH CIRCUMFLEX AND DOT BELOW
<i2>	<U1EC8>	LATIN CAPITAL LETTER I WITH HOOK ABOVE
<i2>	<U1EC9>	LATIN SMALL LETTER I WITH HOOK ABOVE
<I-·>	<U1ECA>	LATIN CAPITAL LETTER I WITH DOT BELOW
<i-·>	<U1ECB>	LATIN SMALL LETTER I WITH DOT BELOW
<O-·>	<U1ECC>	LATIN CAPITAL LETTER O WITH DOT BELOW
<o-·>	<U1ECD>	LATIN SMALL LETTER O WITH DOT BELOW
<O2>	<U1ECE>	LATIN CAPITAL LETTER O WITH HOOK ABOVE
<o2>	<U1ECF>	LATIN SMALL LETTER O WITH HOOK ABOVE
<O/>´>	<U1ED0>	LATIN CAPITAL LETTER O WITH CIRCUMFLEX AND ACUTE
<o/>´>	<U1ED1>	LATIN SMALL LETTER O WITH CIRCUMFLEX AND ACUTE
<O/>!>	<U1ED2>	LATIN CAPITAL LETTER O WITH CIRCUMFLEX AND GRAVE
<o/>!>	<U1ED3>	LATIN SMALL LETTER O WITH CIRCUMFLEX AND GRAVE
<O/>2>	<U1ED4>	LATIN CAPITAL LETTER O WITH CIRCUMFLEX AND HOOK ABOVE
<o/>2>	<U1ED5>	LATIN SMALL LETTER O WITH CIRCUMFLEX AND HOOK ABOVE
<O/>?>	<U1ED6>	LATIN CAPITAL LETTER O WITH CIRCUMFLEX AND TILDE

464	<o/>?>	<U1ED7>	LATIN SMALL LETTER O WITH CIRCUMFLEX AND TILDE
464	<O/>-.>	<U1ED8>	LATIN CAPITAL LETTER O WITH CIRCUMFLEX AND DOT BELOW
464	<o/>-.>	<U1ED9>	LATIN SMALL LETTER O WITH CIRCUMFLEX AND DOT BELOW
464	<O9'>	<U1EDA>	LATIN CAPITAL LETTER O WITH HORN AND ACUTE
464	<o9'>	<U1EDB>	LATIN SMALL LETTER O WITH HORN AND ACUTE
464	<O9!>	<U1EDC>	LATIN CAPITAL LETTER O WITH HORN AND GRAVE
464	<o9!>	<U1EDD>	LATIN SMALL LETTER O WITH HORN AND GRAVE
464	<O92>	<U1EDE>	LATIN CAPITAL LETTER O WITH HORN AND HOOK ABOVE
464	<o92>	<U1EDF>	LATIN SMALL LETTER O WITH HORN AND HOOK ABOVE
464	<O9?>	<U1EE0>	LATIN CAPITAL LETTER O WITH HORN AND TILDE
464	<o9?>	<U1EE1>	LATIN SMALL LETTER O WITH HORN AND TILDE
464	<O9-.>	<U1EE2>	LATIN CAPITAL LETTER O WITH HORN AND DOT BELOW
464	<o9-.>	<U1EE3>	LATIN SMALL LETTER O WITH HORN AND DOT BELOW
464	<U-.>	<U1EE4>	LATIN CAPITAL LETTER U WITH DOT BELOW
464	<u-.>	<U1EE5>	LATIN SMALL LETTER U WITH DOT BELOW
464	<U2>	<U1EE6>	LATIN CAPITAL LETTER U WITH HOOK ABOVE
464	<u2>	<U1EE7>	LATIN SMALL LETTER U WITH HOOK ABOVE
464	<U9'>	<U1EE8>	LATIN CAPITAL LETTER U WITH HORN AND ACUTE
464	<u9'>	<U1EE9>	LATIN SMALL LETTER U WITH HORN AND ACUTE
464	<U9!>	<U1EEA>	LATIN CAPITAL LETTER U WITH HORN AND GRAVE
464	<u9!>	<U1EEB>	LATIN SMALL LETTER U WITH HORN AND GRAVE
464	<U92>	<U1EEC>	LATIN CAPITAL LETTER U WITH HORN AND HOOK ABOVE
464	<u92>	<U1EED>	LATIN SMALL LETTER U WITH HORN AND HOOK ABOVE
464	<U9?>	<U1EEE>	LATIN CAPITAL LETTER U WITH HORN AND TILDE
464	<u9?>	<U1EEF>	LATIN SMALL LETTER U WITH HORN AND TILDE
464	<U9-.>	<U1EF0>	LATIN CAPITAL LETTER U WITH HORN AND DOT BELOW
464	<u9-.>	<U1EF1>	LATIN SMALL LETTER U WITH HORN AND DOT BELOW
464	<Y!>	<U1EF2>	LATIN CAPITAL LETTER Y WITH GRAVE
464	<y!>	<U1EF3>	LATIN SMALL LETTER Y WITH GRAVE
464	<Y-.>	<U1EF4>	LATIN CAPITAL LETTER Y WITH DOT BELOW
464	<y-.>	<U1EF5>	LATIN SMALL LETTER Y WITH DOT BELOW
464	<Y2>	<U1EF6>	LATIN CAPITAL LETTER Y WITH HOOK ABOVE
464	<y2>	<U1EF7>	LATIN SMALL LETTER Y WITH HOOK ABOVE
464	<Y?>	<U1EF8>	LATIN CAPITAL LETTER Y WITH TILDE
464	<y?>	<U1EF9>	LATIN SMALL LETTER Y WITH TILDE
464	<a*,>	<U1F00>	GREEK SMALL LETTER ALPHA WITH PSILI
464	<a*,>	<U1F01>	GREEK SMALL LETTER ALPHA WITH DASIA
464	<a*;!>	<U1F02>	GREEK SMALL LETTER ALPHA WITH PSILI AND VARIA
464	<a*;!>	<U1F03>	GREEK SMALL LETTER ALPHA WITH DASIA AND VARIA
464	<a*,'>	<U1F04>	GREEK SMALL LETTER ALPHA WITH PSILI AND OXIA
464	<a*,'>	<U1F05>	GREEK SMALL LETTER ALPHA WITH DASIA AND OXIA
464	<a*,>?>	<U1F06>	GREEK SMALL LETTER ALPHA WITH PSILI AND PERISPOMENI
464	<a*,>?>	<U1F07>	GREEK SMALL LETTER ALPHA WITH DASIA AND PERISPOMENI
464	<A*,>	<U1F08>	GREEK CAPITAL LETTER ALPHA WITH PSILI
464	<A*,>	<U1F09>	GREEK CAPITAL LETTER ALPHA WITH DASIA
464	<A*;!>	<U1F0A>	GREEK CAPITAL LETTER ALPHA WITH PSILI AND VARIA
464	<A*;!>	<U1F0B>	GREEK CAPITAL LETTER ALPHA WITH DASIA AND VARIA
464	<A*,'>	<U1F0C>	GREEK CAPITAL LETTER ALPHA WITH PSILI AND OXIA
464	<A*,'>	<U1F0D>	GREEK CAPITAL LETTER ALPHA WITH DASIA AND OXIA
464	<A*,>?>	<U1F0E>	GREEK CAPITAL LETTER ALPHA WITH PSILI AND PERISPOMENI
464	<A*,>?>	<U1F0F>	GREEK CAPITAL LETTER ALPHA WITH DASIA AND PERISPOMENI
464	<e*,>	<U1F10>	GREEK SMALL LETTER EPSILON WITH PSILI
464	<e*,>	<U1F11>	GREEK SMALL LETTER EPSILON WITH DASIA
464	<e*;!>	<U1F12>	GREEK SMALL LETTER EPSILON WITH PSILI AND VARIA
464	<e*;!>	<U1F13>	GREEK SMALL LETTER EPSILON WITH DASIA AND VARIA
464	<e*,'>	<U1F14>	GREEK SMALL LETTER EPSILON WITH PSILI AND OXIA
464	<e*,'>	<U1F15>	GREEK SMALL LETTER EPSILON WITH DASIA AND OXIA
464	<E*,>	<U1F18>	GREEK CAPITAL LETTER EPSILON WITH PSILI
464	<E*,>	<U1F19>	GREEK CAPITAL LETTER EPSILON WITH DASIA
464	<E*;!>	<U1F1A>	GREEK CAPITAL LETTER EPSILON WITH PSILI AND VARIA
464	<E*;!>	<U1F1B>	GREEK CAPITAL LETTER EPSILON WITH DASIA AND VARIA
464	<E*,'>	<U1F1C>	GREEK CAPITAL LETTER EPSILON WITH PSILI AND OXIA
464	<E*,'>	<U1F1D>	GREEK CAPITAL LETTER EPSILON WITH DASIA AND OXIA
464	<y*,>	<U1F20>	GREEK SMALL LETTER ETA WITH PSILI
464	<y*,>	<U1F21>	GREEK SMALL LETTER ETA WITH DASIA
464	<y*;!>	<U1F22>	GREEK SMALL LETTER ETA WITH PSILI AND VARIA
464	<y*;!>	<U1F23>	GREEK SMALL LETTER ETA WITH DASIA AND VARIA
464	<y*,'>	<U1F24>	GREEK SMALL LETTER ETA WITH PSILI AND OXIA
464	<y*,'>	<U1F25>	GREEK SMALL LETTER ETA WITH DASIA AND OXIA
464	<y*,>?>	<U1F26>	GREEK SMALL LETTER ETA WITH PSILI AND PERISPOMENI
464	<y*,>?>	<U1F27>	GREEK SMALL LETTER ETA WITH DASIA AND PERISPOMENI
464	<Y*,>	<U1F28>	GREEK CAPITAL LETTER ETA WITH PSILI
464	<Y*,>	<U1F29>	GREEK CAPITAL LETTER ETA WITH DASIA
464	<Y*;!>	<U1F2A>	GREEK CAPITAL LETTER ETA WITH PSILI AND VARIA
464	<Y*;!>	<U1F2B>	GREEK CAPITAL LETTER ETA WITH DASIA AND VARIA
464	<Y*,'>	<U1F2C>	GREEK CAPITAL LETTER ETA WITH PSILI AND OXIA
464	<Y*,'>	<U1F2D>	GREEK CAPITAL LETTER ETA WITH DASIA AND OXIA
464	<Y*,>?>	<U1F2E>	GREEK CAPITAL LETTER ETA WITH PSILI AND PERISPOMENI
464	<Y*,>?>	<U1F2F>	GREEK CAPITAL LETTER ETA WITH DASIA AND PERISPOMENI
464	<i*,>	<U1F30>	GREEK SMALL LETTER IOTA WITH PSILI
464	<i*,>	<U1F31>	GREEK SMALL LETTER IOTA WITH DASIA
464	<i*;!>	<U1F32>	GREEK SMALL LETTER IOTA WITH PSILI AND VARIA
464	<i*;!>	<U1F33>	GREEK SMALL LETTER IOTA WITH DASIA AND VARIA
464	<i*,'>	<U1F34>	GREEK SMALL LETTER IOTA WITH PSILI AND OXIA
464	<i*,'>	<U1F35>	GREEK SMALL LETTER IOTA WITH DASIA AND OXIA
464	<i*,>?>	<U1F36>	GREEK SMALL LETTER IOTA WITH PSILI AND PERISPOMENI
464	<i*,>?>	<U1F37>	GREEK SMALL LETTER IOTA WITH DASIA AND PERISPOMENI
464	<I*,>	<U1F38>	GREEK CAPITAL LETTER IOTA WITH PSILI

47334	<I*;>	<U1F39>	GREEK CAPITAL LETTER IOTA WITH DASIA
47335	<I*;!>	<U1F3A>	GREEK CAPITAL LETTER IOTA WITH PSILI AND VARIA
47336	<I*;!>	<U1F3B>	GREEK CAPITAL LETTER IOTA WITH DASIA AND VARIA
47337	<I*;'>	<U1F3C>	GREEK CAPITAL LETTER IOTA WITH PSILI AND OXIA
47338	<I*;'>	<U1F3D>	GREEK CAPITAL LETTER IOTA WITH DASIA AND OXIA
47339	<I*;?>	<U1F3E>	GREEK CAPITAL LETTER IOTA WITH PSILI AND PERISPOMENI
47400	<I*;?>	<U1F3F>	GREEK CAPITAL LETTER IOTA WITH DASIA AND PERISPOMENI
47410	<o*;>	<U1F40>	GREEK SMALL LETTER OMICRON WITH PSILI
47411	<o*;>	<U1F41>	GREEK SMALL LETTER OMICRON WITH DASIA
47413	<o*;!>	<U1F42>	GREEK SMALL LETTER OMICRON WITH PSILI AND VARIA
47414	<o*;!>	<U1F43>	GREEK SMALL LETTER OMICRON WITH DASIA AND VARIA
47415	<o*;'>	<U1F44>	GREEK SMALL LETTER OMICRON WITH PSILI AND OXIA
47416	<o*;'>	<U1F45>	GREEK SMALL LETTER OMICRON WITH DASIA AND OXIA
47417	<O*;>	<U1F48>	GREEK CAPITAL LETTER OMICRON WITH PSILI
47418	<O*;>	<U1F49>	GREEK CAPITAL LETTER OMICRON WITH DASIA
47490	<O*;!>	<U1F4A>	GREEK CAPITAL LETTER OMICRON WITH PSILI AND VARIA
47500	<O*;!>	<U1F4B>	GREEK CAPITAL LETTER OMICRON WITH DASIA AND VARIA
47501	<O*;'>	<U1F4C>	GREEK CAPITAL LETTER OMICRON WITH PSILI AND OXIA
47502	<O*;'>	<U1F4D>	GREEK CAPITAL LETTER OMICRON WITH DASIA AND OXIA
47503	<u*;>	<U1F50>	GREEK SMALL LETTER UPSILON WITH PSILI
47504	<u*;>	<U1F51>	GREEK SMALL LETTER UPSILON WITH DASIA
47506	<u*;!>	<U1F52>	GREEK SMALL LETTER UPSILON WITH PSILI AND VARIA
47507	<u*;!>	<U1F53>	GREEK SMALL LETTER UPSILON WITH DASIA AND VARIA
47508	<u*;'>	<U1F54>	GREEK SMALL LETTER UPSILON WITH PSILI AND OXIA
47509	<u*;'>	<U1F55>	GREEK SMALL LETTER UPSILON WITH DASIA AND OXIA
47598	<u*;?>	<U1F56>	GREEK SMALL LETTER UPSILON WITH PSILI AND PERISPOMENI
47600	<u*;?>	<U1F57>	GREEK SMALL LETTER UPSILON WITH DASIA AND PERISPOMENI
47610	<U*;>	<U1F59>	GREEK CAPITAL LETTER UPSILON WITH DASIA
47611	<U*;!>	<U1F5B>	GREEK CAPITAL LETTER UPSILON WITH DASIA AND VARIA
47612	<U*;'>	<U1F5D>	GREEK CAPITAL LETTER UPSILON WITH DASIA AND OXIA
47614	<U*;'>	<U1F5F>	GREEK CAPITAL LETTER UPSILON WITH DASIA AND PERISPOMENI
47650	<w*;>	<U1F60>	GREEK SMALL LETTER OMEGA WITH PSILI
47651	<w*;>	<U1F61>	GREEK SMALL LETTER OMEGA WITH DASIA
47652	<w*;!>	<U1F62>	GREEK SMALL LETTER OMEGA WITH PSILI AND VARIA
47653	<w*;!>	<U1F63>	GREEK SMALL LETTER OMEGA WITH DASIA AND VARIA
47654	<w*;'>	<U1F64>	GREEK SMALL LETTER OMEGA WITH PSILI AND OXIA
47655	<w*;'>	<U1F65>	GREEK SMALL LETTER OMEGA WITH DASIA AND OXIA
47700	<w*;?>	<U1F66>	GREEK SMALL LETTER OMEGA WITH PSILI AND PERISPOMENI
47701	<w*;?>	<U1F67>	GREEK SMALL LETTER OMEGA WITH DASIA AND PERISPOMENI
47730	<W*;>	<U1F68>	GREEK CAPITAL LETTER OMEGA WITH PSILI
47731	<W*;>	<U1F69>	GREEK CAPITAL LETTER OMEGA WITH DASIA
47732	<W*;!>	<U1F6A>	GREEK CAPITAL LETTER OMEGA WITH PSILI AND VARIA
47733	<W*;!>	<U1F6B>	GREEK CAPITAL LETTER OMEGA WITH DASIA AND VARIA
47734	<W*;'>	<U1F6C>	GREEK CAPITAL LETTER OMEGA WITH PSILI AND OXIA
47735	<W*;'>	<U1F6D>	GREEK CAPITAL LETTER OMEGA WITH DASIA AND OXIA
47736	<W*;'>	<U1F6E>	GREEK CAPITAL LETTER OMEGA WITH PSILI AND PERISPOMENI
47737	<W*;'>	<U1F6F>	GREEK CAPITAL LETTER OMEGA WITH DASIA AND PERISPOMENI
47738	<a*!>	<U1F70>	GREEK SMALL LETTER ALPHA WITH VARIA
47739	<a*!>	<U1F71>	GREEK SMALL LETTER ALPHA WITH OXIA
47740	<e*!>	<U1F72>	GREEK SMALL LETTER EPSILON WITH VARIA
47741	<e*!>	<U1F73>	GREEK SMALL LETTER EPSILON WITH OXIA
47742	<y*!>	<U1F74>	GREEK SMALL LETTER ETA WITH VARIA
47743	<y*!>	<U1F75>	GREEK SMALL LETTER ETA WITH OXIA
47744	<i*!>	<U1F76>	GREEK SMALL LETTER IOTA WITH VARIA
47745	<i*!>	<U1F77>	GREEK SMALL LETTER IOTA WITH OXIA
47746	<o*!>	<U1F78>	GREEK SMALL LETTER OMICRON WITH VARIA
47747	<o*!>	<U1F79>	GREEK SMALL LETTER OMICRON WITH OXIA
47748	<u*!>	<U1F7A>	GREEK SMALL LETTER UPSILON WITH VARIA
47749	<u*!>	<U1F7B>	GREEK SMALL LETTER UPSILON WITH OXIA
47750	<w*!>	<U1F7C>	GREEK SMALL LETTER OMEGA WITH VARIA
47751	<w*!>	<U1F7D>	GREEK SMALL LETTER OMEGA WITH OXIA
47752	<a*,>	<U1F80>	GREEK SMALL LETTER ALPHA WITH PSILI AND YPOGEGRAMMENI
47753	<a*,>	<U1F81>	GREEK SMALL LETTER ALPHA WITH DASIA AND YPOGEGRAMMENI
47754	<a*;!>	<U1F82>	GREEK SMALL LETTER ALPHA WITH PSILI AND VARIA AND YPOGEGRAMMENI
47755	<a*;!>	<U1F83>	GREEK SMALL LETTER ALPHA WITH DASIA AND VARIA AND YPOGEGRAMMENI
47756	<a*;'>	<U1F84>	GREEK SMALL LETTER ALPHA WITH PSILI AND OXIA AND YPOGEGRAMMENI
47757	<a*;'>	<U1F85>	GREEK SMALL LETTER ALPHA WITH DASIA AND OXIA AND YPOGEGRAMMENI
47758	<a*;'>	<U1F86>	GREEK SMALL LETTER ALPHA WITH PSILI AND PERISPOMENI AND YPOGEGRAMMENI
47759	<a*;'>	<U1F87>	GREEK SMALL LETTER ALPHA WITH DASIA AND PERISPOMENI AND YPOGEGRAMMENI
47760	<A*,>	<U1F88>	GREEK CAPITAL LETTER ALPHA WITH PSILI AND PROSGEGRAMMENI
47761	<A*,>	<U1F89>	GREEK CAPITAL LETTER ALPHA WITH DASIA AND PROSGEGRAMMENI
47762	<A*;!>	<U1F8A>	GREEK CAPITAL LETTER ALPHA WITH PSILI AND VARIA AND PROSGEGRAMMENI
47763	<A*;!>	<U1F8B>	GREEK CAPITAL LETTER ALPHA WITH DASIA AND VARIA AND PROSGEGRAMMENI
47764	<A*;'>	<U1F8C>	GREEK CAPITAL LETTER ALPHA WITH PSILI AND OXIA AND PROSGEGRAMMENI
47765	<A*;'>	<U1F8D>	GREEK CAPITAL LETTER ALPHA WITH DASIA AND OXIA AND PROSGEGRAMMENI
47766	<A*;'>	<U1F8E>	GREEK CAPITAL LETTER ALPHA WITH PSILI AND PERISPOMENI AND PROSGEGRAMMENI
47767	<A*;'>	<U1F8F>	GREEK CAPITAL LETTER ALPHA WITH DASIA AND PERISPOMENI AND PROSGEGRAMMENI
47768	<y*,>	<U1F90>	GREEK SMALL LETTER ETA WITH PSILI AND YPOGEGRAMMENI
47769	<y*,>	<U1F91>	GREEK SMALL LETTER ETA WITH DASIA AND YPOGEGRAMMENI
47770	<y*;!>	<U1F92>	GREEK SMALL LETTER ETA WITH PSILI AND VARIA AND YPOGEGRAMMENI
47771	<y*;!>	<U1F93>	GREEK SMALL LETTER ETA WITH DASIA AND VARIA AND YPOGEGRAMMENI
47772	<y*;'>	<U1F94>	GREEK SMALL LETTER ETA WITH PSILI AND OXIA AND YPOGEGRAMMENI
47773	<y*;'>	<U1F95>	GREEK SMALL LETTER ETA WITH DASIA AND OXIA AND YPOGEGRAMMENI
47774	<y*;'>	<U1F96>	GREEK SMALL LETTER ETA WITH PSILI AND PERISPOMENI AND YPOGEGRAMMENI
47775	<y*;'>	<U1F97>	GREEK SMALL LETTER ETA WITH DASIA AND PERISPOMENI AND YPOGEGRAMMENI
47776	<Y*,>	<U1F98>	GREEK CAPITAL LETTER ETA WITH PSILI AND PROSGEGRAMMENI
47777	<Y*,>	<U1F99>	GREEK CAPITAL LETTER ETA WITH DASIA AND PROSGEGRAMMENI
47778	<Y*;!>	<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

49	<W*´>	<U1FFB>	GREEK CAPITAL LETTER OMEGA WITH OXIA
49	<W*J>	<U1FFC>	GREEK CAPITAL LETTER OMEGA WITH PROSGEGRAMMENI
49	</// <sup>*</sup> >	<U1FFD>	GREEK OXIA
49	<¡¡>	<U1FFE>	GREEK DASIA
49	<1N>	<U2002>	EN SPACE
49	<1M>	<U2003>	EM SPACE
49	<3M>	<U2004>	THREE-PER-EM SPACE
49	<4M>	<U2005>	FOUR-PER-EM SPACE
49	<6M>	<U2006>	SIX-PER-EM SPACE
49	<LR>	<U200E>	LEFT-TO-RIGHT MARK
49	<RL>	<U200F>	RIGHT-TO-LEFT MARK
49	<1T>	<U2009>	THIN SPACE
49	<1H>	<U200A>	HAIR SPACE
49	<-1>	<U2010>	HYPHEN
49	<-N>	<U2013>	EN DASH
49	<-M>	<U2014>	EM DASH
49	<-3>	<U2015>	HORIZONTAL BAR
49	<¡2>	<U2016>	DOUBLE VERTICAL LINE
49	<=2>	<U2017>	DOUBLE LOW LINE
49	<'6>	<U2018>	LEFT SINGLE QUOTATION MARK
49	<'9>	<U2019>	RIGHT SINGLE QUOTATION MARK
49	<.9>	<U201A>	SINGLE LOW-9 QUOTATION MARK
49	<9'>	<U201B>	SINGLE HIGH-REVERSED-9 QUOTATION MARK
49	<"6>	<U201C>	LEFT DOUBLE QUOTATION MARK
49	<"9>	<U201D>	RIGHT DOUBLE QUOTATION MARK
49	<:9>	<U201E>	DOUBLE LOW-9 QUOTATION MARK
49	<9">	<U201F>	DOUBLE HIGH-REVERSED-9 QUOTATION MARK
49	</// <sup>-</sup> >	<U2020>	DAGGER
49	</// <sup>=</sup> >	<U2021>	DOUBLE DAGGER
49	<sb>	<U2022>	BULLET
49	<3b>	<U2023>	TRIANGULAR BULLET
49	<..>	<U2025>	TWO DOT LEADER
49	<.3>	<U2026>	HORIZONTAL ELLIPSIS
49	<.->	<U2027>	HYPHENATION POINT
49	<linesep>	<U2028>	LINE SEPARATOR
49	<parasep>	<U2029>	PARAGRAPH SEPARATOR
49	<%0>	<U2030>	PER MILLE SIGN
49	<1'>	<U2032>	PRIME
49	<2'>	<U2033>	DOUBLE PRIME
49	<3'>	<U2034>	TRIPLE PRIME
49	<1">	<U2035>	REVERSED PRIME
49	<2">	<U2036>	REVERSED DOUBLE PRIME
49	<3">	<U2037>	REVERSED TRIPLE PRIME
49	<Ca>	<U2038>	CARET
49	<<1>	<U2039>	SINGLE LEFT-POINTING ANGLE QUOTATION MARK
49	</>1>	<U203A>	SINGLE RIGHT-POINTING ANGLE QUOTATION MARK
49	<:X>	<U203B>	REFERENCE MARK
49	<!*2>	<U203C>	DOUBLE EXCLAMATION MARK
49	<' ->	<U203E>	OVERLINE
49	<-b>	<U2043>	HYPHEN BULLET
49	</// <sup>f</sup> >	<U2044>	FRACTION SLASH
49	<0S>	<U2070>	SUPERSCRIPIT ZERO
49	<4S>	<U2074>	SUPERSCRIPIT FOUR
49	<5S>	<U2075>	SUPERSCRIPIT FIVE
49	<6S>	<U2076>	SUPERSCRIPIT SIX
49	<7S>	<U2077>	SUPERSCRIPIT SEVEN
49	<8S>	<U2078>	SUPERSCRIPIT EIGHT
49	<9S>	<U2079>	SUPERSCRIPIT NINE
49	<+S>	<U207A>	SUPERSCRIPIT PLUS SIGN
49	<-S>	<U207B>	SUPERSCRIPIT MINUS
49	<=S>	<U207C>	SUPERSCRIPIT EQUALS SIGN
49	<(S>	<U207D>	SUPERSCRIPIT LEFT PARENTHESIS
49	<)S>	<U207E>	SUPERSCRIPIT RIGHT PARENTHESIS
49	<nS>	<U207F>	SUPERSCRIPIT LATIN SMALL LETTER N
49	<0s>	<U2080>	SUBSCRIPT ZERO
49	<1s>	<U2081>	SUBSCRIPT ONE
49	<2s>	<U2082>	SUBSCRIPT TWO
49	<3s>	<U2083>	SUBSCRIPT THREE
49	<4s>	<U2084>	SUBSCRIPT FOUR
49	<5s>	<U2085>	SUBSCRIPT FIVE
49	<6s>	<U2086>	SUBSCRIPT SIX
49	<7s>	<U2087>	SUBSCRIPT SEVEN
49	<8s>	<U2088>	SUBSCRIPT EIGHT
49	<9s>	<U2089>	SUBSCRIPT NINE
49	<+s>	<U208A>	SUBSCRIPT PLUS SIGN
49	<-s>	<U208B>	SUBSCRIPT MINUS
49	<=s>	<U208C>	SUBSCRIPT EQUALS SIGN
49	<(s>	<U208D>	SUBSCRIPT LEFT PARENTHESIS
49	<)s>	<U208E>	SUBSCRIPT RIGHT PARENTHESIS
49	<Ff>	<U20A3>	FRENCH FRANC SIGN
49	<Li>	<U20A4>	LIRA SIGN
49	<Pt>	<U20A7>	PESETA SIGN
49	<W=>	<U20A9>	WON SIGN
49	<"7>	<U20D1>	COMBINING RIGHT HARPOON ABOVE
49	<oC>	<U2103>	DEGREE CELSIUS
49	<co>	<U2105>	CARE OF
49	<oF>	<U2109>	DEGREE FAHRENHEIT
49	<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
<45>	<U2158>	VULGAR FRACTION FOUR FIFTHS
<16>	<U2159>	VULGAR FRACTION ONE SIXTH
<56>	<U215A>	VULGAR FRACTION FIVE SIXTHS
<18>	<U215B>	VULGAR FRACTION ONE EIGHTH
<38>	<U215C>	VULGAR FRACTION THREE EIGHTHS
<58>	<U215D>	VULGAR FRACTION FIVE EIGHTHS
<78>	<U215E>	VULGAR FRACTION SEVEN EIGHTHS
<1R>	<U2160>	ROMAN NUMERAL ONE
<2R>	<U2161>	ROMAN NUMERAL TWO
<3R>	<U2162>	ROMAN NUMERAL THREE
<4R>	<U2163>	ROMAN NUMERAL FOUR
<5R>	<U2164>	ROMAN NUMERAL FIVE
<6R>	<U2165>	ROMAN NUMERAL SIX
<7R>	<U2166>	ROMAN NUMERAL SEVEN
<8R>	<U2167>	ROMAN NUMERAL EIGHT
<9R>	<U2168>	ROMAN NUMERAL NINE
<aR>	<U2169>	ROMAN NUMERAL TEN
 	<U216A>	ROMAN NUMERAL ELEVEN
<cR>	<U216B>	ROMAN NUMERAL TWELVE
<50R>	<U216C>	ROMAN NUMERAL FIFTY
<100R>	<U216D>	ROMAN NUMERAL ONE HUNDRED
<500R>	<U216E>	ROMAN NUMERAL FIVE HUNDRED
<1000R>	<U216F>	ROMAN NUMERAL ONE THOUSAND
<1r>	<U2170>	SMALL ROMAN NUMERAL ONE
<2r>	<U2171>	SMALL ROMAN NUMERAL TWO
<3r>	<U2172>	SMALL ROMAN NUMERAL THREE
<4r>	<U2173>	SMALL ROMAN NUMERAL FOUR
<5r>	<U2174>	SMALL ROMAN NUMERAL FIVE
<6r>	<U2175>	SMALL ROMAN NUMERAL SIX
<7r>	<U2176>	SMALL ROMAN NUMERAL SEVEN
<8r>	<U2177>	SMALL ROMAN NUMERAL EIGHT
<9r>	<U2178>	SMALL ROMAN NUMERAL NINE
<ar>	<U2179>	SMALL ROMAN NUMERAL TEN
 	<U217A>	SMALL ROMAN NUMERAL ELEVEN
<cr>	<U217B>	SMALL ROMAN NUMERAL TWELVE
<50r>	<U217C>	SMALL ROMAN NUMERAL FIFTY
<100r>	<U217D>	SMALL ROMAN NUMERAL ONE HUNDRED
<500r>	<U217E>	SMALL ROMAN NUMERAL FIVE HUNDRED
<1000r>	<U217F>	SMALL ROMAN NUMERAL ONE THOUSAND
<1000RCD>	<U2180>	ROMAN NUMERAL ONE THOUSAND C D
<5000R>	<U2181>	ROMAN NUMERAL FIVE THOUSAND
<10000R>	<U2182>	ROMAN NUMERAL TEN THOUSAND
<->	<U2190>	LEFTWARDS ARROW
<-↑>	<U2191>	UPWARDS ARROW
<-/>	<U2192>	RIGHTWARDS ARROW
<-v>	<U2193>	DOWNWARDS ARROW
<</>	<U2194>	LEFT RIGHT ARROW
<UD>	<U2195>	UP DOWN ARROW
<<!!>	<U2196>	NORTH WEST ARROW
</////>	<U2197>	NORTH EAST ARROW
<!!/>	<U2198>	SOUTH EAST ARROW
<</////>	<U2199>	SOUTH WEST ARROW
<UD->	<U21A8>	UP DOWN ARROW WITH BASE
</>v>	<U21C0>	RIGHTWARDS HARPOON WITH BARB UPWARDS
<<=>	<U21D0>	LEFTWARDS DOUBLE ARROW
<=>	<U21D2>	RIGHTWARDS DOUBLE ARROW
<==>	<U21D4>	LEFT RIGHT DOUBLE ARROW
<FA>	<U2200>	FOR ALL
<dP>	<U2202>	PARTIAL DIFFERENTIAL
<TE>	<U2203>	THERE EXISTS
<//0>	<U2205>	EMPTY SET
<DE>	<U2206>	INCREMENT
<NB>	<U2207>	NABLA
<(->	<U2208>	ELEMENT OF
<(-)>	<U220B>	CONTAINS AS MEMBER
<FP>	<U220E>	END OF PROOF
<*P>	<U220F>	N-ARY PRODUCT
<+Z>	<U2211>	N-ARY SUMMATION
<-2>	<U2212>	MINUS SIGN
<-+>	<U2213>	MINUS-OR-PLUS SIGN
<. +>	<U2214>	DOT PLUS
<*->	<U2217>	ASTERISK OPERATOR
<Ob>	<U2218>	RING OPERATOR
<Sb>	<U2219>	BULLET OPERATOR
<RT>	<U221A>	SQUARE ROOT
<0(>	<U221D>	PROPORTIONAL TO

<00>	<U221E>	INFINITY
<-L>	<U221F>	RIGHT ANGLE
<-V>	<U2220>	ANGLE
<PP>	<U2225>	PARALLEL TO
<AN>	<U2227>	LOGICAL AND
<OR>	<U2228>	LOGICAL OR
<(U>	<U2229>	INTERSECTION
<)U>	<U222A>	UNION
<In>	<U222B>	INTEGRAL
<DI>	<U222C>	DOUBLE INTEGRAL
<Io>	<U222E>	CONTOUR INTEGRAL
<. : >	<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
<p-o>	<U24DF>	CIRCLED LATIN SMALL LETTER P
<q-o>	<U24E0>	CIRCLED LATIN SMALL LETTER Q
<r-o>	<U24E1>	CIRCLED LATIN SMALL LETTER R
<s-o>	<U24E2>	CIRCLED LATIN SMALL LETTER S
<t-o>	<U24E3>	CIRCLED LATIN SMALL LETTER T
<u-o>	<U24E4>	CIRCLED LATIN SMALL LETTER U
<v-o>	<U24E5>	CIRCLED LATIN SMALL LETTER V
<w-o>	<U24E6>	CIRCLED LATIN SMALL LETTER W
<x-o>	<U24E7>	CIRCLED LATIN SMALL LETTER X
<y-o>	<U24E8>	CIRCLED LATIN SMALL LETTER Y
<z-o>	<U24E9>	CIRCLED LATIN SMALL LETTER Z
<0-o>	<U24EA>	CIRCLED DIGIT ZERO
<hh>	<U2500>	BOX DRAWINGS LIGHT HORIZONTAL
<HH->	<U2501>	BOX DRAWINGS HEAVY HORIZONTAL
<vv>	<U2502>	BOX DRAWINGS LIGHT VERTICAL
<VV->	<U2503>	BOX DRAWINGS HEAVY VERTICAL
<3->	<U2504>	BOX DRAWINGS LIGHT TRIPLE DASH HORIZONTAL
<3_>	<U2505>	BOX DRAWINGS HEAVY TRIPLE DASH HORIZONTAL
<3!>	<U2506>	BOX DRAWINGS LIGHT TRIPLE DASH VERTICAL
<3//>	<U2507>	BOX DRAWINGS HEAVY TRIPLE DASH VERTICAL
<4->	<U2508>	BOX DRAWINGS LIGHT QUADRUPLE DASH HORIZONTAL
<4_>	<U2509>	BOX DRAWINGS HEAVY QUADRUPLE DASH HORIZONTAL
<4!>	<U250A>	BOX DRAWINGS LIGHT QUADRUPLE DASH VERTICAL
<4//>	<U250B>	BOX DRAWINGS HEAVY QUADRUPLE DASH VERTICAL
<dr>	<U250C>	BOX DRAWINGS LIGHT DOWN AND RIGHT
<dR->	<U250D>	BOX DRAWINGS DOWN LIGHT AND RIGHT HEAVY
<Dr->	<U250E>	BOX DRAWINGS DOWN HEAVY AND RIGHT LIGHT
<DR->	<U250F>	BOX DRAWINGS HEAVY DOWN AND RIGHT
<dl>	<U2510>	BOX DRAWINGS LIGHT DOWN AND LEFT
<dL->	<U2511>	BOX DRAWINGS DOWN LIGHT AND LEFT HEAVY
<Dl->	<U2512>	BOX DRAWINGS DOWN HEAVY AND LEFT LIGHT
<LD->	<U2513>	BOX DRAWINGS HEAVY DOWN AND LEFT
<ur>	<U2514>	BOX DRAWINGS LIGHT UP AND RIGHT
<uR->	<U2515>	BOX DRAWINGS UP LIGHT AND RIGHT HEAVY
<Ur->	<U2516>	BOX DRAWINGS UP HEAVY AND RIGHT LIGHT
<UR->	<U2517>	BOX DRAWINGS HEAVY UP AND RIGHT
<ul>	<U2518>	BOX DRAWINGS LIGHT UP AND LEFT
<uL->	<U2519>	BOX DRAWINGS UP LIGHT AND LEFT HEAVY
<Ul->	<U251A>	BOX DRAWINGS UP HEAVY AND LEFT LIGHT
<UL->	<U251B>	BOX DRAWINGS HEAVY UP AND LEFT
<vr>	<U251C>	BOX DRAWINGS LIGHT VERTICAL AND RIGHT
<vR->	<U251D>	BOX DRAWINGS VERTICAL LIGHT AND RIGHT HEAVY
<Udr>	<U251E>	BOX DRAWINGS UP HEAVY AND RIGHT DOWN LIGHT
<uDr>	<U251F>	BOX DRAWINGS DOWN HEAVY AND RIGHT UP LIGHT
<Vr->	<U2520>	BOX DRAWINGS VERTICAL HEAVY AND RIGHT LIGHT
<UdR>	<U2521>	BOX DRAWINGS DOWN LIGHT AND RIGHT UP HEAVY
<uDR>	<U2522>	BOX DRAWINGS UP LIGHT AND RIGHT DOWN HEAVY
<VR->	<U2523>	BOX DRAWINGS HEAVY VERTICAL AND RIGHT
<vl>	<U2524>	BOX DRAWINGS LIGHT VERTICAL AND LEFT
<vL->	<U2525>	BOX DRAWINGS VERTICAL LIGHT AND LEFT HEAVY
<Udl>	<U2526>	BOX DRAWINGS UP HEAVY AND LEFT DOWN LIGHT
<udl>	<U2527>	BOX DRAWINGS DOWN HEAVY AND LEFT UP LIGHT
<vL->	<U2528>	BOX DRAWINGS VERTICAL HEAVY AND LEFT LIGHT
<UdL>	<U2529>	BOX DRAWINGS DOWN LIGHT AND LEFT UP HEAVY
<uDL>	<U252A>	BOX DRAWINGS UP LIGHT AND LEFT DOWN HEAVY
<VL->	<U252B>	BOX DRAWINGS HEAVY VERTICAL AND LEFT
<dh>	<U252C>	BOX DRAWINGS LIGHT DOWN AND HORIZONTAL
<dLr>	<U252D>	BOX DRAWINGS LEFT HEAVY AND RIGHT DOWN LIGHT
<dLr>	<U252E>	BOX DRAWINGS RIGHT HEAVY AND LEFT DOWN LIGHT
<dH->	<U252F>	BOX DRAWINGS DOWN LIGHT AND HORIZONTAL HEAVY
<Dh->	<U2530>	BOX DRAWINGS DOWN HEAVY AND HORIZONTAL LIGHT
<DLr>	<U2531>	BOX DRAWINGS RIGHT LIGHT AND LEFT DOWN HEAVY
<DLr>	<U2532>	BOX DRAWINGS LEFT LIGHT AND RIGHT DOWN HEAVY
<DH->	<U2533>	BOX DRAWINGS HEAVY DOWN AND HORIZONTAL
<uh>	<U2534>	BOX DRAWINGS LIGHT UP AND HORIZONTAL
<uLr>	<U2535>	BOX DRAWINGS LEFT HEAVY AND RIGHT UP LIGHT
<uLr>	<U2536>	BOX DRAWINGS RIGHT HEAVY AND LEFT UP LIGHT
<uH->	<U2537>	BOX DRAWINGS UP LIGHT AND HORIZONTAL HEAVY
<UH->	<U2538>	BOX DRAWINGS UP HEAVY AND HORIZONTAL LIGHT
<ULr>	<U2539>	BOX DRAWINGS RIGHT LIGHT AND LEFT UP HEAVY
<ULr>	<U253A>	BOX DRAWINGS LEFT LIGHT AND RIGHT UP HEAVY
<UH->	<U253B>	BOX DRAWINGS HEAVY UP AND HORIZONTAL
<vh>	<U253C>	BOX DRAWINGS LIGHT VERTICAL AND HORIZONTAL
<vLr>	<U253D>	BOX DRAWINGS LEFT HEAVY AND RIGHT VERTICAL LIGHT
<vLr>	<U253E>	BOX DRAWINGS RIGHT HEAVY AND LEFT VERTICAL LIGHT
<vH->	<U253F>	BOX DRAWINGS VERTICAL LIGHT AND HORIZONTAL HEAVY

<Udh>	<U2540>	BOX DRAWINGS UP HEAVY AND DOWN HORIZONTAL LIGHT
<udh>	<U2541>	BOX DRAWINGS DOWN HEAVY AND UP HORIZONTAL LIGHT
<Vh->	<U2542>	BOX DRAWINGS VERTICAL HEAVY AND HORIZONTAL LIGHT
<Udlr>	<U2543>	BOX DRAWINGS LEFT UP HEAVY AND RIGHT DOWN LIGHT
<Udlr>	<U2544>	BOX DRAWINGS RIGHT UP HEAVY AND LEFT DOWN LIGHT
<uDLr>	<U2545>	BOX DRAWINGS LEFT DOWN HEAVY AND RIGHT UP LIGHT
<uDlR>	<U2546>	BOX DRAWINGS RIGHT DOWN HEAVY AND LEFT UP LIGHT
<UdH>	<U2547>	BOX DRAWINGS DOWN LIGHT AND UP HORIZONTAL HEAVY
<uDh>	<U2548>	BOX DRAWINGS UP LIGHT AND DOWN HORIZONTAL HEAVY
<VLr>	<U2549>	BOX DRAWINGS RIGHT LIGHT AND LEFT VERTICAL HEAVY
<VlR>	<U254A>	BOX DRAWINGS LEFT LIGHT AND RIGHT VERTICAL HEAVY
<VH->	<U254B>	BOX DRAWINGS HEAVY VERTICAL AND HORIZONTAL
<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
<uL>	<U255B>	BOX DRAWINGS UP SINGLE AND LEFT DOUBLE
<UL>	<U255C>	BOX DRAWINGS UP DOUBLE AND LEFT SINGLE
<UL>	<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
<OL>	<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
<0u>	<U263A>	WHITE SMILING FACE
<0U>	<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
<cS->	<U2664>	WHITE SPADE SUIT
<cH->	<U2665>	BLACK HEART SUIT
<cD->	<U2666>	BLACK DIAMOND SUIT
<cC->	<U2667>	WHITE CLUB SUIT
<Md>	<U2669>	QUARTER NOTE
<M8>	<U266A>	EIGHTH NOTE
<M2>	<U266B>	BEAMED EIGHTH NOTES
<M16>	<U266C>	BEAMED SIXTEENTH NOTES
<Mb>	<U266D>	MUSIC FLAT SIGN
<Mx>	<U266E>	MUSIC NATURAL SIGN
<MX>	<U266F>	MUSIC SHARP SIGN
<OK>	<U2713>	CHECK MARK
<XX>	<U2717>	BALLOT X
<-X>	<U2720>	MALTESE CROSS
<IS>	<U3000>	IDEOGRAPHIC SPACE
< , _>	<U3001>	IDEOGRAPHIC COMMA
< . _>	<U3002>	IDEOGRAPHIC FULL STOP
<+ ">	<U3003>	DITTO MARK
<JIS>	<U3004>	JAPANESE INDUSTRIAL STANDARD SYMBOL
<* _>	<U3005>	IDEOGRAPHIC ITERATION MARK
< ; _>	<U3006>	IDEOGRAPHIC CLOSING MARK
< 0 _>	<U3007>	IDEOGRAPHIC NUMBER ZERO
<<+>	<U300A>	LEFT DOUBLE ANGLE BRACKET
</>+>	<U300B>	RIGHT DOUBLE ANGLE BRACKET
<<'>	<U300C>	LEFT CORNER BRACKET
</>'>	<U300D>	RIGHT CORNER BRACKET
<<">	<U300E>	LEFT WHITE CORNER BRACKET
</>">	<U300F>	RIGHT WHITE CORNER BRACKET
< (">	<U3010>	LEFT BLACK LENTICULAR BRACKET
< (">	<U3011>	RIGHT BLACK LENTICULAR BRACKET
<=T>	<U3012>	POSTAL MARK
<= _>	<U3013>	GETA MARK
< ('>	<U3014>	LEFT TORTOISE SHELL BRACKET
< )'>	<U3015>	RIGHT TORTOISE SHELL BRACKET
< (I>	<U3016>	LEFT WHITE LENTICULAR BRACKET
< )I>	<U3017>	RIGHT WHITE LENTICULAR BRACKET
<-?>	<U301C>	WAVE DASH
<=T: )>	<U3020>	POSTAL MARK FACE
<A5>	<U3041>	HIRAGANA LETTER SMALL A
<a5>	<U3042>	HIRAGANA LETTER A
<I5>	<U3043>	HIRAGANA LETTER SMALL I
<i5>	<U3044>	HIRAGANA LETTER I
<U5>	<U3045>	HIRAGANA LETTER SMALL U
<u5>	<U3046>	HIRAGANA LETTER U
<E5>	<U3047>	HIRAGANA LETTER SMALL E
<e5>	<U3048>	HIRAGANA LETTER E
<O5>	<U3049>	HIRAGANA LETTER SMALL O
<o5>	<U304A>	HIRAGANA LETTER O
<ka>	<U304B>	HIRAGANA LETTER KA
<ga>	<U304C>	HIRAGANA LETTER GA
<ki>	<U304D>	HIRAGANA LETTER KI
<gi>	<U304E>	HIRAGANA LETTER GI
<ku>	<U304F>	HIRAGANA LETTER KU
<gu>	<U3050>	HIRAGANA LETTER GU
<ke>	<U3051>	HIRAGANA LETTER KE
<ge>	<U3052>	HIRAGANA LETTER GE
<ko>	<U3053>	HIRAGANA LETTER KO
<go>	<U3054>	HIRAGANA LETTER GO
<sa>	<U3055>	HIRAGANA LETTER SA
<za>	<U3056>	HIRAGANA LETTER ZA
<si>	<U3057>	HIRAGANA LETTER SI
<zi>	<U3058>	HIRAGANA LETTER ZI
<su>	<U3059>	HIRAGANA LETTER SU
<zu>	<U305A>	HIRAGANA LETTER ZU
<se>	<U305B>	HIRAGANA LETTER SE
<ze>	<U305C>	HIRAGANA LETTER ZE
<so>	<U305D>	HIRAGANA LETTER SO
<zo>	<U305E>	HIRAGANA LETTER ZO
<ta>	<U305F>	HIRAGANA LETTER TA
<da>	<U3060>	HIRAGANA LETTER DA
<ti>	<U3061>	HIRAGANA LETTER TI
<di>	<U3062>	HIRAGANA LETTER DI
<tU>	<U3063>	HIRAGANA LETTER SMALL TU
<tu>	<U3064>	HIRAGANA LETTER TU
<du>	<U3065>	HIRAGANA LETTER DU
<te>	<U3066>	HIRAGANA LETTER TE

<de>	<U3067>	HIRAGANA LETTER DE
<to>	<U3068>	HIRAGANA LETTER TO
<do>	<U3069>	HIRAGANA LETTER DO
<na>	<U306A>	HIRAGANA LETTER NA
<ni>	<U306B>	HIRAGANA LETTER NI
<nu>	<U306C>	HIRAGANA LETTER NU
<ne>	<U306D>	HIRAGANA LETTER NE
<no>	<U306E>	HIRAGANA LETTER NO
<ha>	<U306F>	HIRAGANA LETTER HA
<ba>	<U3070>	HIRAGANA LETTER BA
<pa>	<U3071>	HIRAGANA LETTER PA
<hi>	<U3072>	HIRAGANA LETTER HI
<bi>	<U3073>	HIRAGANA LETTER BI
<pi>	<U3074>	HIRAGANA LETTER PI
<hu>	<U3075>	HIRAGANA LETTER HU
<bu>	<U3076>	HIRAGANA LETTER BU
<pu>	<U3077>	HIRAGANA LETTER PU
<he>	<U3078>	HIRAGANA LETTER HE
<be>	<U3079>	HIRAGANA LETTER BE
<pe>	<U307A>	HIRAGANA LETTER PE
<ho>	<U307B>	HIRAGANA LETTER HO
<bo>	<U307C>	HIRAGANA LETTER BO
<po>	<U307D>	HIRAGANA LETTER PO
<ma>	<U307E>	HIRAGANA LETTER MA
<mi>	<U307F>	HIRAGANA LETTER MI
<mu>	<U3080>	HIRAGANA LETTER MU
<me>	<U3081>	HIRAGANA LETTER ME
<mo>	<U3082>	HIRAGANA LETTER MO
<yA>	<U3083>	HIRAGANA LETTER SMALL YA
<ya>	<U3084>	HIRAGANA LETTER YA
<yU>	<U3085>	HIRAGANA LETTER SMALL YU
<yu>	<U3086>	HIRAGANA LETTER YU
<yO>	<U3087>	HIRAGANA LETTER SMALL YO
<yo>	<U3088>	HIRAGANA LETTER YO
<ra>	<U3089>	HIRAGANA LETTER RA
<ri>	<U308A>	HIRAGANA LETTER RI
<ru>	<U308B>	HIRAGANA LETTER RU
<re>	<U308C>	HIRAGANA LETTER RE
<ro>	<U308D>	HIRAGANA LETTER RO
<wA>	<U308E>	HIRAGANA LETTER SMALL WA
<wa>	<U308F>	HIRAGANA LETTER WA
<wi>	<U3090>	HIRAGANA LETTER WI
<we>	<U3091>	HIRAGANA LETTER WE
<wo>	<U3092>	HIRAGANA LETTER WO
<n5>	<U3093>	HIRAGANA LETTER N
<vu>	<U3094>	HIRAGANA LETTER VU
<"5>	<U309B>	KATAKANA-HIRAGANA VOICED SOUND MARK
<05>	<U309C>	KATAKANA-HIRAGANA SEMI-VOICED SOUND MARK
<*5>	<U309D>	HIRAGANA ITERATION MARK
<+5>	<U309E>	HIRAGANA VOICED ITERATION MARK
<a6>	<U30A1>	KATAKANA LETTER SMALL A
<A6>	<U30A2>	KATAKANA LETTER A
<i6>	<U30A3>	KATAKANA LETTER SMALL I
<I6>	<U30A4>	KATAKANA LETTER I
<u6>	<U30A5>	KATAKANA LETTER SMALL U
<U6>	<U30A6>	KATAKANA LETTER U
<e6>	<U30A7>	KATAKANA LETTER SMALL E
<E6>	<U30A8>	KATAKANA LETTER E
<o6>	<U30A9>	KATAKANA LETTER SMALL O
<O6>	<U30AA>	KATAKANA LETTER O
<Ka>	<U30AB>	KATAKANA LETTER KA
<Ga>	<U30AC>	KATAKANA LETTER GA
<Ki>	<U30AD>	KATAKANA LETTER KI
<Gi>	<U30AE>	KATAKANA LETTER GI
<Ku>	<U30AF>	KATAKANA LETTER KU
<Gu>	<U30B0>	KATAKANA LETTER GU
<Ke>	<U30B1>	KATAKANA LETTER KE
<Ge>	<U30B2>	KATAKANA LETTER GE
<Ko>	<U30B3>	KATAKANA LETTER KO
<Go>	<U30B4>	KATAKANA LETTER GO
<Sa>	<U30B5>	KATAKANA LETTER SA
<Za>	<U30B6>	KATAKANA LETTER ZA
<Si>	<U30B7>	KATAKANA LETTER SI
<Zi>	<U30B8>	KATAKANA LETTER ZI
<Su>	<U30B9>	KATAKANA LETTER SU
<Zu>	<U30BA>	KATAKANA LETTER ZU
<Se>	<U30BB>	KATAKANA LETTER SE
<Ze>	<U30BC>	KATAKANA LETTER ZE
<So>	<U30BD>	KATAKANA LETTER SO
<Zo>	<U30BE>	KATAKANA LETTER ZO
<Ta>	<U30BF>	KATAKANA LETTER TA
<Da>	<U30C0>	KATAKANA LETTER DA
<Ti>	<U30C1>	KATAKANA LETTER TI
<Di>	<U30C2>	KATAKANA LETTER DI
<TU>	<U30C3>	KATAKANA LETTER SMALL TU
<Tu>	<U30C4>	KATAKANA LETTER TU
<Du>	<U30C5>	KATAKANA LETTER DU
<Te>	<U30C6>	KATAKANA LETTER TE

<De>	<U30C7>	KATAKANA LETTER DE
<To>	<U30C8>	KATAKANA LETTER TO
<Do>	<U30C9>	KATAKANA LETTER DO
<Na>	<U30CA>	KATAKANA LETTER NA
<Ni>	<U30CB>	KATAKANA LETTER NI
<Nu>	<U30CC>	KATAKANA LETTER NU
<Ne>	<U30CD>	KATAKANA LETTER NE
<No>	<U30CE>	KATAKANA LETTER NO
<Ha>	<U30CF>	KATAKANA LETTER HA
<Ba>	<U30D0>	KATAKANA LETTER BA
<Pa>	<U30D1>	KATAKANA LETTER PA
<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



5702	<eN>	<U3125>	BOPOMOFO LETTER ENG
5703	<er>	<U3126>	BOPOMOFO LETTER ER
5704	<i4>	<U3127>	BOPOMOFO LETTER I
5705	<u4>	<U3128>	BOPOMOFO LETTER U
5706	<iu>	<U3129>	BOPOMOFO LETTER IU
5707	<v4>	<U312A>	BOPOMOFO LETTER V
5708	<nG>	<U312B>	BOPOMOFO LETTER NG
5709	<gn>	<U312C>	BOPOMOFO LETTER GN
5710	<(JU)>	<U321C>	PARENTHESESIZED HANGUL CIEUC U
5711	<1c>	<U3220>	PARENTHESESIZED IDEOGRAPH ONE
5712	<2c>	<U3221>	PARENTHESESIZED IDEOGRAPH TWO
5713	<3c>	<U3222>	PARENTHESESIZED IDEOGRAPH THREE
5714	<4c>	<U3223>	PARENTHESESIZED IDEOGRAPH FOUR
5715	<5c>	<U3224>	PARENTHESESIZED IDEOGRAPH FIVE
5716	<6c>	<U3225>	PARENTHESESIZED IDEOGRAPH SIX
5717	<7c>	<U3226>	PARENTHESESIZED IDEOGRAPH SEVEN
5718	<8c>	<U3227>	PARENTHESESIZED IDEOGRAPH EIGHT
5719	<9c>	<U3228>	PARENTHESESIZED IDEOGRAPH NINE
5720	<10c>	<U3229>	PARENTHESESIZED IDEOGRAPH TEN
5721	<KSC>	<U327F>	KOREAN STANDARD SYMBOL
5722	<am>	<U33C2>	SQUARE AM
5723	<pm>	<U33D8>	SQUARE PM
5724	<ff>	<UFB00>	LATIN SMALL LIGATURE FF
5725	<fi>	<UFB01>	LATIN SMALL LIGATURE FI
5726	<fl>	<UFB02>	LATIN SMALL LIGATURE FL
5727	<ffi>	<UFB03>	LATIN SMALL LIGATURE FFI
5728	<ffl>	<UFB04>	LATIN SMALL LIGATURE FFL
5729	<St>	<UFB05>	LATIN SMALL LIGATURE LONG S T
5730	<st>	<UFB06>	LATIN SMALL LIGATURE ST
5731	<3+>	<UFE7D>	ARABIC SHADDA MEDIAL FORM
5732	<aM.>	<UFE82>	ARABIC LETTER ALEF WITH MADDA ABOVE FINAL FORM
5733	<aH.>	<UFE84>	ARABIC LETTER ALEF WITH HAMZA ABOVE FINAL FORM
5734	<aH.>	<UFE88>	ARABIC LETTER ALEF WITH HAMZA BELOW FINAL FORM
5735	<a+>	<UFE8D>	ARABIC LETTER ALEF ISOLATED FORM
5736	<a+.>	<UFE8E>	ARABIC LETTER ALEF FINAL FORM
5737	<b+>	<UFE8F>	ARABIC LETTER BEH ISOLATED FORM
5738	<b+.>	<UFE90>	ARABIC LETTER BEH FINAL FORM
5739	<b+,>	<UFE91>	ARABIC LETTER BEH INITIAL FORM
5740	<b+;>	<UFE92>	ARABIC LETTER BEH MEDIAL FORM
5741	<tm->	<UFE93>	ARABIC LETTER TEH MARBUTA ISOLATED FORM
5742	<tm.>	<UFE94>	ARABIC LETTER TEH MARBUTA FINAL FORM
5743	<t+>	<UFE95>	ARABIC LETTER TEH ISOLATED FORM
5744	<t+.>	<UFE96>	ARABIC LETTER TEH FINAL FORM
5745	<t+,>	<UFE97>	ARABIC LETTER TEH INITIAL FORM
5746	<t+;>	<UFE98>	ARABIC LETTER TEH MEDIAL FORM
5747	<tk->	<UFE99>	ARABIC LETTER THEH ISOLATED FORM
5748	<tk.>	<UFE9A>	ARABIC LETTER THEH FINAL FORM
5749	<tk,>	<UFE9B>	ARABIC LETTER THEH INITIAL FORM
5750	<tk;>	<UFE9C>	ARABIC LETTER THEH MEDIAL FORM
5751	<g+>	<UFE9D>	ARABIC LETTER JEEM ISOLATED FORM
5752	<g+.>	<UFE9E>	ARABIC LETTER JEEM FINAL FORM
5753	<g+,>	<UFE9F>	ARABIC LETTER JEEM INITIAL FORM
5754	<g+;>	<UFEA0>	ARABIC LETTER JEEM MEDIAL FORM
5755	<hk->	<UFEA1>	ARABIC LETTER HAH ISOLATED FORM
5756	<hk.>	<UFEA2>	ARABIC LETTER HAH FINAL FORM
5757	<hk,>	<UFEA3>	ARABIC LETTER HAH INITIAL FORM
5758	<hk;>	<UFEA4>	ARABIC LETTER HAH MEDIAL FORM
5759	<x+>	<UFEA5>	ARABIC LETTER KHAH ISOLATED FORM
5760	<x+.>	<UFEA6>	ARABIC LETTER KHAH FINAL FORM
5761	<x+,>	<UFEA7>	ARABIC LETTER KHAH INITIAL FORM
5762	<x+;>	<UFEA8>	ARABIC LETTER KHAH MEDIAL FORM
5763	<d+>	<UFEA9>	ARABIC LETTER DAL ISOLATED FORM
5764	<d+.>	<UFEAA>	ARABIC LETTER DAL FINAL FORM
5765	<dk->	<UFEAB>	ARABIC LETTER THAL ISOLATED FORM
5766	<dk.>	<UFEAC>	ARABIC LETTER THAL FINAL FORM
5767	<r+>	<UFEAD>	ARABIC LETTER REH ISOLATED FORM
5768	<r+.>	<UFEAE>	ARABIC LETTER REH FINAL FORM
5769	<z+>	<UFEAF>	ARABIC LETTER ZAIN ISOLATED FORM
5770	<z+.>	<UFEB0>	ARABIC LETTER ZAIN FINAL FORM
5771	<s+>	<UFEB1>	ARABIC LETTER SEEN ISOLATED FORM
5772	<s+.>	<UFEB2>	ARABIC LETTER SEEN FINAL FORM
5773	<s+,>	<UFEB3>	ARABIC LETTER SEEN INITIAL FORM
5774	<s+;>	<UFEB4>	ARABIC LETTER SEEN MEDIAL FORM
5775	<sn->	<UFEB5>	ARABIC LETTER SHEEN ISOLATED FORM
5776	<sn.>	<UFEB6>	ARABIC LETTER SHEEN FINAL FORM
5777	<sn,>	<UFEB7>	ARABIC LETTER SHEEN INITIAL FORM
5778	<sn;>	<UFEB8>	ARABIC LETTER SHEEN MEDIAL FORM
5779	<c+>	<UFEB9>	ARABIC LETTER SAD ISOLATED FORM
5780	<c+.>	<UFEBA>	ARABIC LETTER SAD FINAL FORM
5781	<c+,>	<UFEBB>	ARABIC LETTER SAD INITIAL FORM
5782	<c+;>	<UFEBC>	ARABIC LETTER SAD MEDIAL FORM
5783	<dd->	<UFEBD>	ARABIC LETTER DAD ISOLATED FORM
5784	<dd.>	<UFEBE>	ARABIC LETTER DAD FINAL FORM
5785	<dd,>	<UFEBF>	ARABIC LETTER DAD INITIAL FORM
5786	<dd;>	<UFEC0>	ARABIC LETTER DAD MEDIAL FORM
5787	<tj->	<UFEC1>	ARABIC LETTER TAH ISOLATED FORM
5788	<tj.>	<UFEC2>	ARABIC LETTER TAH FINAL FORM
5789	<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+;.>	<UFEC9>	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
<i+;.>	<UFED0>	ARABIC LETTER GHAIN MEDIAL FORM
<f+>	<UFED1>	ARABIC LETTER FEH ISOLATED FORM
<f+.>	<UFED2>	ARABIC LETTER FEH FINAL FORM
<f+,>	<UFED3>	ARABIC LETTER FEH INITIAL FORM
<f+;.>	<UFED4>	ARABIC LETTER FEH MEDIAL FORM
<q+>	<UFED5>	ARABIC LETTER QAF ISOLATED FORM
<q+.>	<UFED6>	ARABIC LETTER QAF FINAL FORM
<q+,>	<UFED7>	ARABIC LETTER QAF INITIAL FORM
<q+;.>	<UFED8>	ARABIC LETTER QAF MEDIAL FORM
<k+>	<UFED9>	ARABIC LETTER KAF ISOLATED FORM
<k+.>	<UFEDA>	ARABIC LETTER KAF FINAL FORM
<k+,>	<UFEDB>	ARABIC LETTER KAF INITIAL FORM
<k+;.>	<UFEDC>	ARABIC LETTER KAF MEDIAL FORM
<l+>	<UFEDD>	ARABIC LETTER LAM ISOLATED FORM
<l+.>	<UFEDE>	ARABIC LETTER LAM FINAL FORM
<l+,>	<UFEDF>	ARABIC LETTER LAM INITIAL FORM
<l+;.>	<UFEE0>	ARABIC LETTER LAM MEDIAL FORM
<m+>	<UFEE1>	ARABIC LETTER MEEM ISOLATED FORM
<m+.>	<UFEE2>	ARABIC LETTER MEEM FINAL FORM
<m+,>	<UFEE3>	ARABIC LETTER MEEM INITIAL FORM
<m+;.>	<UFEE4>	ARABIC LETTER MEEM MEDIAL FORM
<n+>	<UFEE5>	ARABIC LETTER NOON ISOLATED FORM
<n+.>	<UFEE6>	ARABIC LETTER NOON FINAL FORM
<n+,>	<UFEE7>	ARABIC LETTER NOON INITIAL FORM
<n+;.>	<UFEE8>	ARABIC LETTER NOON MEDIAL FORM
<h+>	<UFEE9>	ARABIC LETTER HEH ISOLATED FORM
<h+.>	<UFEEA>	ARABIC LETTER HEH FINAL FORM
<h+,>	<UFEEB>	ARABIC LETTER HEH INITIAL FORM
<h+;.>	<UFEEC>	ARABIC LETTER HEH MEDIAL FORM
<w+>	<UFEEED>	ARABIC LETTER WAW ISOLATED FORM
<w+.>	<UFEEEE>	ARABIC LETTER WAW FINAL FORM
<j+>	<UFEEF>	ARABIC LETTER ALEF MAKSURA ISOLATED FORM
<j+.>	<UFEEF0>	ARABIC LETTER ALEF MAKSURA FINAL FORM
<y+>	<UFEEF1>	ARABIC LETTER YEH ISOLATED FORM
<y+.>	<UFEEF2>	ARABIC LETTER YEH FINAL FORM
<y+,>	<UFEEF3>	ARABIC LETTER YEH INITIAL FORM
<y+;.>	<UFEEF4>	ARABIC LETTER YEH MEDIAL FORM
<lM->	<UFEEF5>	ARABIC LIGATURE LAM WITH ALEF WITH MADDA ABOVE ISOLATED FORM
<lM.>	<UFEEF6>	ARABIC LIGATURE LAM WITH ALEF WITH MADDA ABOVE FINAL FORM
<lH->	<UFEEF7>	ARABIC LIGATURE LAM WITH ALEF WITH HAMZA ABOVE ISOLATED FORM
<lH.>	<UFEEF8>	ARABIC LIGATURE LAM WITH ALEF WITH HAMZA ABOVE FINAL FORM
<lH->	<UFEEF9>	ARABIC LIGATURE LAM WITH ALEF WITH HAMZA BELOW ISOLATED FORM
<lH.>	<UFEEFA>	ARABIC LIGATURE LAM WITH ALEF WITH HAMZA BELOW FINAL FORM
<la->	<UFEEFB>	ARABIC LIGATURE LAM WITH ALEF ISOLATED FORM
<la.>	<UFEEFC>	ARABIC LIGATURE LAM WITH ALEF FINAL FORM
<H->	<U0023>	NUMBER SIGN
<!S>	<U0024>	DOLLAR SIGN
<@>	<U0040>	COMMERCIAL AT
<Oa>	<U0040>	COMMERCIAL AT
<!C>	<U00A2>	CENT SIGN
<L->	<U00A3>	POUND SIGN
<Xo>	<U00A4>	CURRENCY SIGN
<Y->	<U00A5>	YEN SIGN
<!B>	<U00A6>	BROKEN BAR
<So>	<U00A7>	SECTION SIGN
<7!>	<U00AC>	NOT SIGN
<9I>	<U00B6>	PILCROW SIGN
<_>	<U2500>	BOX DRAWINGS LIGHT HORIZONTAL
<_>	<U2501>	BOX DRAWINGS HEAVY HORIZONTAL
<_!>	<U2502>	BOX DRAWINGS LIGHT VERTICAL
<_V/>>	<U250C>	BOX DRAWINGS LIGHT DOWN AND RIGHT
<_V<w>	<U2510>	BOX DRAWINGS LIGHT DOWN AND LEFT
<_A/>>	<U2514>	BOX DRAWINGS LIGHT UP AND RIGHT
<_A<	<U2518>	BOX DRAWINGS LIGHT UP AND LEFT
<_!/>>	<U251C>	BOX DRAWINGS LIGHT VERTICAL AND RIGHT
<_!<	<U2524>	BOX DRAWINGS LIGHT VERTICAL AND LEFT
<_V->	<U252C>	BOX DRAWINGS LIGHT DOWN AND HORIZONTAL
<_A->	<U2534>	BOX DRAWINGS LIGHT UP AND HORIZONTAL
<_!->	<U253C>	BOX DRAWINGS LIGHT VERTICAL AND HORIZONTAL
<_//>>	<U2571>	BOX DRAWINGS LIGHT DIAGONAL UPPER RIGHT TO LOWER LEFT
<_<	<U2572>	BOX DRAWINGS LIGHT DIAGONAL UPPER LEFT TO LOWER RIGHT
<_./>>	<U25E2>	BLACK LOWER RIGHT TRIANGLE
<_<	<U25E3>	BLACK LOWER LEFT TRIANGLE
<_d!>	<U266A>	EIGHTH NOTE

5877  
5878  
5879  
5880  
5881  
5882  
5883  
5884  
5885  
5886  
5887  
5888  
5889  
5890  
5891  
5892  
5893  
5894  
5895  
5896  
5897  
5898  
5899

## **7 CONFORMANCE**

### **7.1 FDCC-set**

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

### **7.2 FDCC-set category**

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

### **7.3 Charmap**

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

### **7.4 Repertoiremap**

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

5900  
5901  
5902  
5903  
5904  
5905  
5906  
5907  
5908  
5909  
5910  
5911  
5912  
5913  
5914  
5915  
5916  
5917  
5918  
5919  
5920  
5921  
5922  
5923  
5924  
5925  
5926  
5927  
5928  
5929  
5930  
5931  
5932  
5933  
5934  
5935  
5936  
5937  
5938  
5939  
5940  
5941  
5942  
5943  
5944  
5945  
5946  
5947  
5948  
5949  
5950  
5951

## 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.

- 5952 11. The "print" CTYPE class includes automatically all "graph" characters.  
5953
- 5954 12. The <Uxxxx> and <Uxxxxxxxx> notations have been introduced as predefined  
5955 symbolic character names, together with a number of symbolic character names derived  
5956 from POSIX and the Internet.  
5957
- 5958 13. New categories LC\_IDENTIFICATION, LC\_XLITERATE, LC\_NAME,  
5959 LC\_ADDRESS, and LC\_TELEPHONE, have been introduced.  
5960
- 5961 14. The LC\_CTYPE has got support for new classes, via the new keywords class and  
5962 map, which corresponds to the C standard library functions iswctype() and towctrans()  
5963 respectively.  
5964
- 5965 15. The "digit" keyword now supports digits for multiple scripts.  
5966
- 5967 16. The LC\_MONETARY category provides support for multiple currencies, such as the  
5968 native currency and the Euro in some European countries.  
5969
- 5970 17. The LC\_TIME has got a number of enhancements to cater for alternate calendars, and  
5971 timezone information may be given.  
5972
- 5973 18. The charmap specification has been enhanced to support ISO 2022.

5974  
5975  
5976  
5977  
5978  
5979  
5980  
5981  
5982  
5983  
5984  
5985  
5986  
5987  
5988  
5989  
5990  
5991  
5992  
5993  
5994  
5995  
5996  
5997  
5998  
5999  
6000  
6001  
6002  
6003  
6004  
6005  
6006  
6007  
6008  
6009  
6010  
6011  
6012  
6013  
6014  
6015  
6016  
6017  
6018  
6019  
6020  
6021  
6022  
6023  
6024  
6025

## **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

6026 Technical Report does not mandate that the encoding used in the FDCC-set is the same as  
 6027 the one used by the application, because an application may decide that it is advantageous  
 6028 to define a FDCC-set in a system-wide encoding rather than having multiple, logically  
 6029 identical FDCC-sets in different encodings, and to convert from the application encoding  
 6030 to the system-wide encoding on usage. Other applications could require encoding-depend-  
 6031 ent FDCC-sets. In either case, the LC\_CTYPE attributes that are directly dependent on  
 6032 the encoding, such as "mb\_cur\_max" and the display width of characters, are not user-  
 6033 specifiable in a locale source, and are consequently not defined as keywords.

6034  
 6035 As the LC\_CTYPE character classes are based on the C Standard character-class  
 6036 definition, the category does not support multicharacter elements. For instance, the  
 6037 German character <sharp-s> is traditionally classified as a lowercase letter. There is no  
 6038 corresponding uppercase letter; in proper capitalization of German text the <sharp-s> will  
 6039 be replaced by SS; i.e., by two characters. This kind of conversion is outside the scope of  
 6040 the "toupper" and "tolower" keywords.

6041  
 6042 The character classes "digit", "xdigit", "lower", "upper", and "space" have a set of  
 6043 automatically included characters. These only need to be specified if the character values  
 6044 (i.e. encoding) differs from the application default values. The definition of character class  
 6045 "digit" allows alternate digits (e.g., Hindi) to be specified here. The definition of character  
 6046 class "xdigit" requires that the characters included in character class "digit" are included  
 6047 here also, and allows for different symbols for the hexadecimal digits 10 through 15.

6048  
 6049 The "combining" and "combining-level3" classes are an IT-enablement of ISO/IEC 10646  
 6050 definitions of combining characters. These can be used to check identifiers for consistence  
 6051 with the guidelines given in TR 10176 annex A.

### 6052 6053 **B.1.3 LC\_COLLATE Rationale.**

6054  
 6055 The LC\_COLLATE category governs the collation order in the FDCC-set, and may thus  
 6056 be useful for the processing of the ISO/IEC 14651 string ordering and comparison  
 6057 standard, the C Standard strxfrm() and strcoll() functions, as well as a number of ISO/IEC  
 6058 9945-2:1993 POSIX utilities.

6059  
 6060  
 6061 The rules governing collation depends to some extent on the use. At least five different  
 6062 levels of increasingly complex collation rules can be distinguished:

- 6063  
 6064 (1) Byte/machine code order. This is the historical collation order in the UNIX  
 6065 system and many proprietary operating systems. Collation is here done  
 6066 character by character, without any regard to context. The primary virtue is that  
 6067 it usually is quite fast, and also completely deterministic; it works well when  
 6068 the native machine collation sequence matches the user expectations.
- 6069 (2) Character order. On this level, collation is also done character by character,  
 6070 without regard to context. The order between characters is, however, not deter-  
 6071 mined by the code values, but on the user's expectations of the correct order  
 6072 between characters. In addition, such a (simple) collation order can specify that  
 6073 certain characters collate equal (e.g., upper and lowercase letters).
- 6074 (3) String ordering. On this level, entire strings are compared based on relatively  
 6075 straightforward rules. At this level, several "passes" may be required to deter-  
 6076 mine the order between two strings. Characters may be ignored in some passes,  
 6077 but not in others; the strings may be compared in different directions; and

- 6078 simple string substitutions may be made before strings are compared. This level  
 6079 is best described as "dictionary" ordering; it is based on the spelling, not the  
 6080 pronunciation, or meaning, of the words.
- 6081 (4) Text search ordering. This is a further refinement of the previous level, best de-  
 6082 scribed as "telephone book ordering"; some common homonyms (words spelled  
 6083 differently but with same pronunciation) are collated together; numbers are  
 6084 collated as if spelled with words, and so on.
- 6085 (5) Semantic level ordering. Words and strings are collated based on their meaning;  
 6086 entire words (such as "the") are eliminated, the ordering is not deterministic.  
 6087 This may requires special software, and is highly dependent on the intended  
 6088 use.

6089  
 6090 While the historical collation order formally is at level 1, for the English language it  
 6091 corresponds roughly to elements at level 2. The user expects to see the output from the  
 6092 "ls" utility sorted very much as it would be in a dictionary. While telephone book ordering  
 6093 would be an optimal goal for standard collation, this was ruled out as the order would be  
 6094 language dependent. Furthermore, a requirement was that the order must be determined  
 6095 solely from the text string and the collation rules; no external information (e.g., "pronu-  
 6096 nciation dictionaries") could be required.

6097  
 6098 As a result, the goal for the collation support is at level 3. This also matches the re-  
 6099 quirements for the Canadian collation order standard, as well as other, known collation  
 6100 requirements for alphabetic scripts. It specifically rules out collation based on pronun-  
 6101 ciation rules, or based on semantic analysis of the text. The syntax for the LC\_COLLATE  
 6102 category source is the result of a cooperative effort between representatives for many  
 6103 countries and organizations working with international issues, such as UniForum, The  
 6104 Open Group, The Unicode Consortium Inc. and ISO, and it meets the requirements for  
 6105 level 3, and has been verified to produce the correct result with examples based on  
 6106 Canadian and Danish collation order.

6107  
 6108 The directives that can be specified in an operand to the order\_start keyword are based on  
 6109 the requirements specified in several proposed standards and in customary use. The  
 6110 following is a rephrasing of rules defined for "lexical ordering in English and French" by  
 6111 the Canadian Standards Association (text in brackets is rephrased):

- 6112  
 6113 (1) Once special characters (punctuation) have been removed from original strings,  
 6114 the ordering is determined by scanning forward (left to right) [disregarding case  
 6115 and diacriticals].
- 6116 (2) In case of equivalence, special characters are once again removed from original  
 6117 strings and the ordering is determined scanning backward (starting from the  
 6118 rightmost character of the string and back), character by character, (disregarding  
 6119 case but considering diacriticals).
- 6120 (3) In case of repeated equivalence, special characters are removed again from  
 6121 original strings and the ordering is determined scanning forward, character by  
 6122 character, (considering both case and diacriticals).
- 6123 (4) If there is still an ordering equivalence after rules (1) through (3) have been  
 6124 applied, then only special characters and the position they occupy in the string  
 6125 are considered to determine ordering. The string that has a special character in  
 6126 the lowest position comes first. If two strings have a special character in the  
 6127 same position, the character [with the lowest collation value] comes first. In  
 6128 case of equality, the other special characters are considered until there is a  
 6129 difference or all special characters have been exhausted.



6130 It is estimated that the Technical Report covers the mechanisms to specify data to cover  
6131 the requirements for all European languages, and Cyrillic and Middle Eastern scripts.  
6132

6133 The Far East (particularly Japanese/Chinese) collations are often based on contextual  
6134 information. In Japan, collations of strings containing CJK characters (ideograms) are  
6135 often done considering some related information such as pronunciation, which needs a  
6136 bulk dictionary (and some common sense). Such collation, in general, falls outside the  
6137 desired goal of this Technical Report, and this Technical Report can support only a  
6138 restricted of collations used in Japan. There are, however, several other collation rules  
6139 (stroke/radical, or "most common pronunciation") which can be supported with the  
6140 mechanism described here. Previous drafts contained a substitute statement, which  
6141 performed a regular expression style replacement before string compares. It has been  
6142 withdrawn based on balloter objections that it was not required for the types of ordering  
6143 this Technical Report is aimed at.  
6144

6145 The character (and collating element) order is defined by the order in which characters and  
6146 elements are specified between the `order_start` and `order_end` keywords. This character  
6147 order is used in range expressions in regular expressions. Weights assigned to the charac-  
6148 ters and elements define the collation sequence; in the absence of weights, the character  
6149 order is also the collation sequence.  
6150

6151 The position keyword was introduced to provide the capability to consider, in a compare,  
6152 the relative position of non-IGNORED characters. As an example, consider the two strings  
6153 "o-ring" and "or-ing". Assuming the hyphen is IGNORED on the first pass, the two strings  
6154 will compare equal, and the position of the hyphen is immaterial. On second pass, all  
6155 characters except the hyphen are IGNORED, and in the normal case the two strings would  
6156 again compare equal. By taking position into account, the first collates before the second.  
6157

6158 This Technical Report adds a number of facilities over the ISO/IEC 9945:1993 POSIX  
6159 standard, especially in the support for the ISO/IEC 10646 UCS character set. These  
6160 extended facilities are in alignment with the ISO/IEC 14651 sorting standard. In addition  
6161 to the facilities provided in ISO/IEC 14651, this specification contains mechanisms to put  
6162 data into a FDCC-set environment, and has added facilities to sort sections differently, has  
6163 facilities to reuse FDCC-sets in different notations via the "equivalence-symbol" keyword  
6164 and tables.  
6165

### 6166 **B.1.3.1 "reorder-after" rationale** 6167

6168 Much work has been done on FDCC-sets, making them quite general. The ISO/IEC 9945-  
6169 2:1993 POSIX standard introduced a "copy" command for all categories of the POSIX  
6170 locale. This is useful for many purposes and it ensures that two FDCC-sets are equivalent  
6171 for this category. A further step in building on previous FDCC-set work is defined in this  
6172 Technical Report.  
6173

6174 Collating sequences often vary a bit from country to country, and from language to  
6175 language, but generally much of the collating sequence is the same. For example the  
6176 Danish sequence is for the most part the same as the German or English collation, but for  
6177 about a dozen letters it differs. The same can be said for Swedish or Hungarian: generally  
6178 the Latin collating sequence is the same, but a few characters are different.  
6179

6180 This Technical Report defines a FDCC-set defined on the character repertoire of the  
6181 ISO/IEC 10646 standard, in a character set independent way. The intention is that some of

6182 the information from this FDCC-set will be acceptable in many cultures, and that it can  
 6183 serve as the basis for modifications in other cultures, to obtain a culturally acceptable  
 6184 specification. Using the "reorder-after" construct will also help improve the overview of  
 6185 what the changes really are for implementers and other users.  
 6186

6187 An example of the use of the "reorder-after" construct is the following. A default  
 6188 international ordering for the Latin alphabet may be adequate for Danish, with the  
 6189 exception of the collation rules for the letters Û, ü, Æ, æ, Ä, ä, Ø, ø, Ö, ö, Å and å. By  
 6190 applying the "reorder-after" construct, the Danish specification can be made more easily  
 6191 by copying and reordering the existing international specification, rather than specifying  
 6192 collation parameters for all Latin letters (with or without diacritics). There is no obligation  
 6193 for Denmark to take this approach, but the "reorder-after" construct provides the  
 6194 mechanism for doing so if it is deemed desirable.  
 6195

### 6196 **B.1.3.2 awk script for "reorder-after" construct**

6197  
 6198 A script has been written in the "awk" language defined in the POSIX standard ISO/IEC  
 6199 9945-2 to implement the "reorder-after" construct. It functions as follows: It reads all of  
 6200 the FDCC-set and if in the LC\_COLLATE category, it processes the line, else it just  
 6201 outputs the line. For the LC\_COLLATE category it reads the lines and puts it into a  
 6202 double linked list of strings identified by a line number; at the end of the LC\_COLLATE  
 6203 category all the lines are output. If the line is a "copy" keyword and it reads the file  
 6204 referenced, extracting the LC\_COLLATE section of the file in to the list of strings. If the  
 6205 line is a "reorder-after" keyword, it sets a pointer to be the line number of the symbol to  
 6206 of the "reorder-after" keyword. If the line is part of the "reorder-after" specification, it is  
 6207 entered into the double linked list at this point, and the previous entry in the double linked  
 6208 list for the <collation-element> is removed from the list. A "reorder-end" keyword  
 6209 terminates the reordering.  
 6210

```

6211 BEGIN { comment = "%"; back[0]= follow[0] = 0; }
6212 /LC_COLLATE/ { coll=1 }
6213 /END LC_COLLATE/ { coll=0; for (lnr= 1; lnr; lnr= follow[lnr]) print c-
6214 ont[lnr] }
6215
6216 { if (coll == 0) print $0 ;
6217   else { if ($1 == "copy") {
6218     file = $2
6219     while (getline < file )
6220       if ( $1 == "LC_COLLATE" ) copy_lc = 1
6221       else if ( $1 == "END" && $2 == "LC_COLLATE" ) copy_lc = 0
6222       else if (copy_lc) {
6223         lnr++
6224         follow[lnr-1] = lnr; back [ lnr ] = lnr-1
6225         cont[lnr] = $0; symb[ $1 ] = lnr
6226       }
6227     close (file )
6228   }
6229   else if ($1 == "reorder-after") { ra=1 ; after = symb [ $2 ] }
6230   else if ($1 == "reorder-end") ra = 0
6231   else {
6232     lnr++
6233     if (ra) follow [ lnr ] = follow [ after ]
6234     if (ra) back [ follow [ after ] ] = lnr
6235     follow[after] = lnr; back [ lnr ] = after
6236     cont[lnr] = $0
6237     if ( ra && $1 != comment && $1 != " ) {
6238       old = symb [ $1 ];
6239       follow [ back [ old ] ] = follow [ old ];
6240       back [ follow [ old ] ] = back [ old ];
6241       symb[ $1 ] = lnr;
6242     }
  
```

```

6243         after = lnr
6244     }
6245 }
6246 }
6247 B.1.3.3 Sample FDCC-set specification for Danish
6248
6249 escape_char /
6250 comment_char %
6251 repertoiremap "i18nrep"
6252 charset "ISO_8859-1:1987"
6253 % Distribution and use is free, also
6254 % for commercial purposes.
6255
6256 LC_VERSION
6257 title "Danish language FDCC-set for Denmark"
6258 source "Danish Standards Association"
6259 address "Kollegievej 6, DK-2920 Charlottenlund, Danmark"
6260 contact "Keld Simonsen"
6261 email "Keld.Simonsen@dkuug.dk"
6262 tel "+45 - 3996-6101"
6263 fax "+45 - 3996-6202"
6264 language "da"
6265 territory "DK"
6266 revision "4.2"
6267 date "1997-12-22"
6268
6269 category i18n:2000;LC_IDENTIFICATION
6270 category i18n:2000;LC_CTYPE
6271 category i18n:2000;LC_COLLATE
6272 category i18n:2000;LC_TIME
6273 category posix:1993;LC_NUMERIC
6274 category i18n:2000;LC_MONETARY
6275 category posix:1993;LC_MESSAGES
6276 category i18n:2000;LC_XLITERATE
6277 category i18n:2000;LC_NAME
6278 category i18n:2000;LC_ADDRESS
6279 category i18n:2000;LC_TELEPHONE
6280
6281 END LC_VERSION
6282
6283 LC_CTYPE
6284 copy "i18n"
6285 END LC_CTYPE
6286
6287 LC_COLLATE
6288 % The ordering algorithm is in accordance
6289 % with Danish Standard DS 377 (1980)
6290 % and the Danish Orthography Dictionary
6291 % (Retskrivningsordbogen, 2. udgave, 1996).
6292 % It is also in accordance with
6293 % Greenlandic orthography.
6294
6295 collating-element <A-A> from "<A><A>"
6296 collating-element <A-a> from "<A><a>"
6297 collating-element <a-A> from "<a><A>"
6298 collating-element <a-a> from "<a><a>"
6299 collating-symbol <SPECIAL>
6300 copy i18n
6301 reorder-after <CAPITAL>
6302 <CAPITAL>
6303 <CAPITAL-SMALL>
6304 <SMALL-CAPITAL>
6305 <SMALL>
6306 reorder-after <q8>
6307 <kk> <Q>;<SPECIAL>;<SMALL>;IGNORE
6308 reorder-after <t8>
6309 <TH> "<T><H>";"<TH><TH>";"<CAPITAL><CAPITAL>";IGNORE
6310 <th> "<T><H>";"<TH><TH>";"<SMALL><SMALL>";IGNORE
6311 reorder-after <y8>
6312 % <U:> and <U"> are treated as <Y> in Danish

```

```

6313 <U:> <Y>;<U:>;<CAPITAL>;IGNORE
6314 <u:> <Y>;<U:>;<SMALL>;IGNORE
6315 <U"> <Y>;<U">;<CAPITAL>;IGNORE
6316 <u"> <Y>;<U">;<SMALL>;IGNORE
6317 reorder-after <z8>
6318 % <AE> is a separate letter in Danish
6319 <AE> <AE>;<NONE>;<CAPITAL>;IGNORE
6320 <ae> <AE>;<NONE>;<SMALL>;IGNORE
6321 <AE'> <AE>;<ACUTE>;<CAPITAL>;IGNORE
6322 <ae'> <AE>;<ACUTE>;<SMALL>;IGNORE
6323 <A3> <AE>;<MACRON>;<CAPITAL>;IGNORE
6324 <a3> <AE>;<MACRON>;<SMALL>;IGNORE
6325 <A:> <AE>;<SPECIAL>;<CAPITAL>;IGNORE
6326 <a:> <AE>;<SPECIAL>;<SMALL>;IGNORE
6327 % <O//> is a separate letter in Danish
6328 <O//> <O//>;<NONE>;<CAPITAL>;IGNORE
6329 <o//> <O//>;<NONE>;<SMALL>;IGNORE
6330 <O//'> <O//>;<ACUTE>;<CAPITAL>;IGNORE
6331 <o//'> <O//>;<ACUTE>;<SMALL>;IGNORE
6332 <O:> <O//>;<DIAERESIS>;<CAPITAL>;IGNORE
6333 <o:> <O//>;<DIAERESIS>;<SMALL>;IGNORE
6334 <O"> <O//>;<DOUBLE-ACUTE>;<CAPITAL>;IGNORE
6335 <o"> <O//>;<DOUBLE-ACUTE>;<SMALL>;IGNORE
6336 % <AA> is a separate letter in Danish
6337 <AA> <AA>;<NONE>;<CAPITAL>;IGNORE
6338 <aa> <AA>;<NONE>;<SMALL>;IGNORE
6339 <A-A> <AA>;<A-A>;<CAPITAL>;IGNORE
6340 <A-a> <AA>;<A-A>;<CAPITAL-SMALL>;IGNORE
6341 <a-A> <AA>;<A-A>;<SMALL-CAPITAL>;IGNORE
6342 <a-a> <AA>;<A-A>;<SMALL>;IGNORE
6343 <AA'> <AA>;<AA'>;<CAPITAL>;IGNORE
6344 <aa'> <AA>;<AA'>;<SMALL>;IGNORE
6345 reorder-end
6346 END LC_COLLATE
6347
6348 LC_MONETARY
6349 int_curr_symbol " <D><K><K><SP>"
6350 currency_symbol " <k><r>"
6351 mon_decimal_point " <,>"
6352 mon_thousands_sep " <.>"
6353 mon_grouping 3;3
6354 positive_sign ""
6355 negative_sign " <->"
6356 int_frac_digits 2
6357 frac_digits 2
6358 p_cs_precedes 1
6359 p_sep_by_space 2
6360 n_cs_precedes 1
6361 n_sep_by_space 2
6362 p_sign_posn 4
6363 n_sign_posn 4
6364 END LC_MONETARY
6365
6366 LC_NUMERIC
6367 decimal_point " <,>"
6368 thousands_sep " <.>"
6369 grouping 3;3
6370 END LC_NUMERIC
6371
6372 LC_TIME
6373 abday " <m><a><n>" ; /
6374 " <t><i><r>" ; " <o><n><s>" ; /
6375 " <t><o><r>" ; " <f><r><e>" ; /
6376 " <l><o//><r>" ; " <s><o//><n>"
6377 day " <m><a><n><d><a><g>" ; /
6378 " <t><i><r><s><d><a><g>" ; /
6379 " <o><n><s><d><a><g>" ; /
6380 " <t><o><r><s><d><a><g>" ; /
6381 " <f><r><e><d><a><g>" ; /
6382 " <l><o//><r><d><a><g>" /

```

```

6383         " <s><o//><n><d><a><g>" ;
6384 week      7;19971201;4
6385 abmon     "<j><a><n>" ; "<f><e><b>" ; /
6386         "<m><a><r>" ; "<a><p><r>" ; /
6387         "<m><a><j>" ; "<j><u><n>" ; /
6388         "<j><u><l>" ; "<a><u><g>" ; /
6389         "<s><e><p>" ; "<o><k><t>" ; /
6390         "<n><o><v>" ; "<d><e><c>"
6391 mon       "<j><a><n><u><a><r>" ; /
6392         "<f><e><b><r><u><a><r>" ; /
6393         "<m><a><r><t><s>" ; /
6394         "<a><p><r><i><l>" ; /
6395         "<m><a><j>" ; /
6396         "<j><u><n><i>" ; /
6397         "<j><u><l><i>" ; /
6398         "<a><u><g><u><s><t>" ; /
6399         "<s><e><p><t><e><m><b><e><r>" ; /
6400         "<o><k><t><o><b><e><r>" ; /
6401         "<n><o><v><e><m><b><e><r>" ; /
6402         "<d><e><c><e><m><b><e><r>"
6403 d_t_fmt    "<%><a><SP><%><F><SP><%><T><SP><%><Z>"
6404 d_fmt      "<%><O><d><.><SP><%><B><SP><%><Y>"
6405 alt_digits "<0><.>;<1><.>;<2><.>;<3><.>;<4><.>; /
6406           <5><.>;<6><.>;<7><.>;<8><.>;<9><.>; /
6407           <1><0><.>;<1><1><.>;<1><2><.>;<1><3><.>;<1><4><.>; /
6408           <1><5><.>;<1><6><.>;<1><7><.>;<1><8><.>;<1><9><.>; /
6409           <2><0><.>;<2><1><.>;<2><2><.>;<2><3><.>;<2><4><.>; /
6410           <2><5><.>;<2><6><.>;<2><7><.>;<2><8><.>;<2><9><.>; /
6411           <3><0><.>;<3><1><.>"
6412 t_fmt      "<%><T>"
6413 am_pm     "" ; ""
6414 t_fmt_ampm ""
6415 timezone  "<C><E><T><-><1><C><E><T><SP><D><S><T><,><M><3><.><5><.><0> /
6416           <,><M><1><0><.><5><.><0>"
6417 END LC_TIME
6418
6419 LC_MESSAGES
6420 yesexpr   "<<(><1><J><j><Y><y><)/>><.><*>"
6421 noexpr    "<<(><0><N><n><)/>><.><*>"
6422 END LC_MESSAGES
6423
6424 LC_NAME
6425 name_fmt  "<%><p><%><t><%><g><%><t><%><m><%><t><%><f>"
6426 name_gen  ""
6427 name_mr   "<h><r>"
6428 name_mrs  "<f><r><u>"
6429 name_miss "<f><r><o/><k><e><n>"
6430 name_ms   "<f><r>"
6431 END LC_NAME
6432
6433 LC_ADDRESS
6434 country_name "<D><a><n><m><a><r><k>"
6435 country_post "<D><K>"
6436 lang_ab     "<d><a>"
6437 lang_term   "<d><a><n>"
6438 postal_fmt  "<%><a><%><N><%><f><%><N><%><d><%><N><%><b><%><N><%> /
6439           <%><s><SP><%><h><SP><%><e><SP><%><r><%><N> /
6440           <%><C><-><%><z><SP><%><T><%><N><%><C><%><N>"
6441 END LC_ADDRESS
6442
6443 LC_TELEPHONE
6444 tel_int_fmt "<+><%><c><SP><%><a><SP><%><l>"
6445 tel_dom_fmt "<%><l>"
6446 int_select "<0><0>"
6447 int_prefix "<4><5>"
6448 END LC_TELEPHONE
6449
6450
6451

```

6452 **B.1.4 LC\_MONETARY Rationale.**

6453

6454 The currency symbol does not appear in LC\_MONETARY because it is not defined in the  
 6455 C Standard's C locale. The C Standard limits the size of decimal points and thousands  
 6456 delimiters to single-byte values. In FDCC-sets based on multibyte coded character sets this  
 6457 cannot be enforced, obviously; this Technical Report does not prohibit such characters, but  
 6458 makes the behaviour unspecified (in the text "In contexts where other standards . . .").

6459

6460 The grouping specification is based on, but not identical to, the C Standard . The "-1"  
 6461 signals that no further grouping is performed, the equivalent of (CHAR\_MAX) in the C  
 6462 Standard ).

6463

6464 The FDCC-set definition is an extension of the C Standard localeconv() specification. In  
 6465 particular, rules on how currency\_symbol is treated are extended to also cover int\_  
 6466 curr\_symbol, and p\_set\_by\_space and n\_sep\_by\_space have been augmented with the  
 6467 value 2, which places a space between the sign and the symbol (if they are adjacent;  
 6468 otherwise it should be treated as a 0). The following table shows the result of various  
 6469 combinations:

6470

6471

6472

6473

6474

6475

6476

6477

6478

6479

6480

6481

6482

6483

6484

6485

6486

6487

6488

6489

6490

6491

6492

6493

6494

6495

6496

6497

6498

6499

6500

6501

6502

6503

		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
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\$+

6487 The following is an example of the interpretation of the mon\_grouping keyword.  
 6488 Assuming that the value to be formatted is 123456789 and the mon\_thousands\_sep is "",  
 6489 then the following table shows the result. The third column shows the equivalent C  
 6490 Standard string that would be used to accommodate this grouping. It is the responsibility  
 6491 of the utility to perform mappings of the formats in this clause to those used by language  
 6492 bindings such as the C Standard .

6493

6494

6495

6496

6497

6498

6499

6500

6501

6502

6503

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.

6504 The multiple currency support is specified such that a FDCC-set can be used without  
 6505 change during the transition period in a static environment. For example in the case of the  
 6506 Euro currency as being employed in a number of European countries, there is no need to  
 6507 change the FDCC-set when shifting from one currency to two concurrent currencies; and  
 6508 there is no need to change FDCC-set, when changing to the Euro as the only currency.  
 6509 Also the same application call can be made to be valid for countries with a single  
 6510 currency and countries with dual currencies. The specifications can also be used without  
 6511 change of the FDCC-set on an installation, when converting from one national currency to  
 6512 another, for example when removing some zeroes to form a new currency.

6513  
 6514 The following example illustrates the support for multiple currencies; the example is for  
 6515 the Euro in Germany:  
 6516

```

6517     LC_MONETARY
6518     valid_from           " ";           "19990101"
6519     valid_to            "20020630";     ""
6520     conversion_rate      1;             195/100
6521     int_curr_symbol      "<D><E><M><SP>";  "<E><U><R><SP>"
6522     currency_symbol     "<D><M>";      "<E><U><R>"
6523     mon_decimal_point    "<,>"
6524     mon_thousands_sep   "<.>"
6525     mon_grouping         3;3
6526     positive_sign       ""
6527     negative_sign        "<->"
6528     int_frac_digits      2;             2
6529     frac_digits          2;             2
6530     p_cs_precedes        1;             1
6531     p_sep_by_space       2;             2
6532     n_cs_precedes        1;             1
6533     n_sep_by_space       2;             2
6534     p_sign_posn          4;             4
6535     n_sign_posn          4;             4
6536
6537     END LC_MONETARY
  
```

### 6538 6539 **B.1.5 LC\_NUMERIC Rationale.**

6540  
 6541 See the rationale for LC\_MONETARY (B.1.3) for a description of the behaviour of  
 6542 grouping.

### 6543 6544 **B.1.6 LC\_TIME Rationale.**

6545  
 6546 The LC\_TIME descriptions of abday, day, and abmon imply a Gregorian style calendar  
 6547 (7-day weeks, 12-month years, leap years, etc.). Other calendars can be supported, for  
 6548 example calendars with a fixed week length.

6549  
 6550 In some FDCC-sets the field descriptors for weekday and month names will be given with  
 6551 an initial small letter. Programs using these fields may need to adjust the capitalization if  
 6552 the output is going to be used at the beginning of a sentence.

6553  
 6554 The field descriptors corresponding to the optional keywords consist of a modifier  
 6555 followed by a traditional field descriptor (for instance %Ex). If the optional keywords are  
 6556 not supported by the application or are unspecified for the current FDCC-set, these field  
 6557 descriptors are treated as the traditional field descriptor. For instance, assume the  
 6558 following keywords:

```

6559     alt_digits "0th";"1st";"2nd";"3rd";"4th";"5th";"6th";"7th";"8th";"9th";"10th"
6560     d_fmt      "The %Od day of %B in %Y"
6561
  
```

6562 On 1776-07-04, the %x field descriptor would result in "The 4th day of July in 1776,"  
6563 while 1789-07-14 would come out as "The 14 day of July in 1789." It can be noted that  
6564 the above example is for illustrative purposes only; the %o modifier is primarily intended  
6565 to provide for Kanji or Hindi digits in date formats. While it is clear that an alternate year  
6566 format is required, there is no consensus on the format or the requirements. As a result,  
6567 while these keywords are reserved, the details are left unspecified. It is expected that  
6568 National Standards Bodies will provide specifications.

#### 6570 **B.1.7 LC\_MESSAGES Rationale.**

6571  
6572 The LC\_MESSAGES category is described in clause 4 as affecting the language used by  
6573 utilities for their output. The mechanism used by the application to accomplish this, other  
6574 than the responses shown here in the FDCC-set definition, is not specified by this version  
6575 of this Technical Report. The ISO internationalization working group is developing an  
6576 interface that would allow applications (and, presumably some of the standard utilities) to  
6577 access messages from various message catalogs, tailored to a user's LC\_MESSAGES  
6578 value.

#### 6580 **B.1.8 LC\_XLITERATE Rationale.**

6581  
6582 Transliteration is often language dependent, transliterating one specific language to another  
6583 specific language. For example transliteration from Russian to English, and from Serbian  
6584 to German would normally be quite different, although the same repertoire of characters  
6585 would be transliterated. Even transliteration of two languages using the same script into  
6586 one language (for example from Russian to Danish and from Serbian to Danish), or  
6587 transliteration of the same language (for example Russian into English or German) may be  
6588 different. The language to be transliterated to is identified with the FDCC-set, which may  
6589 also be used to identify a specific language to be transliterated from. Transliteration may  
6590 also be to a specific repertoire of characters, determined for example by limitations of  
6591 displaying equipment, or what the user can intelligibly read. The capabilities here allows  
6592 for multiple fallback, so that the specification can be valid for all target character  
6593 repertoires, eliminating the need for specific data for each target repertoire.

#### 6595 **B.1.9 LC\_NAME Rationale.**

6596  
6597 The LC\_NAME category gives information to prepare a text for addressing a person, for  
6598 example as a part of a postal address on an envelope, or as a saluting line in a letter.  
6599 The information is intended to be given to an API that has the various naming information  
6600 as parameters and yields a formatted string as the return value.

6601  
6602 The "profession" entry is intended for either the general profession of the person in  
6603 question, or the job title, for use in letters or as part of the address on an envelope.

#### 6605 **B.1.10 LC\_ADDRESS Rationale.**

6606  
6607 The LC\_ADDRESS category gives information to prepare a text for writing an address,  
6608 for example as a part of a postal address on an envelope. The information is intended to  
6609 be given to an API that has the various address information as parameters and yields a  
6610 formatted string as the return value.

6611  
6612  
6613



**6614 B.1.11 LC\_TELEPHONE Rationale.**

6615  
6616 The LC\_TELEPHONE category gives information to prepare a text for writing a telephone  
6617 number. The information is intended to be given to an API that has the various  
6618 information on a telephone number as parameters and yields a formatted string as the  
6619 return value. Both an international and a domestic formatting possibility is available.

**6620 B.2 Character Set Rationale.**

6621  
6622 This Technical Report poses no requirement that multiple character sets or code sets be  
6623 supported, leaving this as a marketing differentiation for implementors. Although multiple  
6624 charmaps are supported, it is the responsibility of the application to provide the file(s); if  
6625 only one is provided, only that one will be accessible.

6626  
6627 The character set description text provides the capability to describe character set attributes  
6628 (such as collation order or character classes) independent of character set encoding, and  
6629 using only the characters in the portable character set. This makes it possible to create  
6630 "generic" FDCC-set source texts for all code sets that share the portable character set  
6631 (such as the ISO/IEC 8859 family or IBM Extended ASCII).

6632  
6633 Applications are free to describe more than one code set in a character set description text.  
6634 For example, if an application defines ISO/IEC 8859-1 as the primary code set, and  
6635 ISO/IEC 8859-2 as an alternate set, with each character from the alternate code set  
6636 preceded in data by a shift code, a character set description text could contain a complete  
6637 description of the primary set and those characters from the secondary that are not  
6638 identical, the encoding of the latter including the shift code.

6639  
6640 Applications are free to choose their own symbolic names, as long as the names identified  
6641 by this Technical Report are also defined; this provides support for already existing  
6642 "character names".

6643  
6644 The charmap was introduced to resolve problems with the portability of, especially,  
6645 FDCC-set sources. While the portable character set (in Table 1) is a constant across all  
6646 FDCC-sets for a particular application, this is not true for the extended character set.  
6647 However, the particular coded character set used for an application does not necessarily  
6648 imply different characteristics or collation: on the contrary, these attributes should in many  
6649 cases be identical, regardless of codeset. The charmap provides the capability to define a  
6650 common FDCC-set definition for multiple codesets (the same FDCC-set source can be  
6651 used for codesets with different extended characters; the ability in the charmap to define  
6652 "empty" names allows for characters missing in certain codesets).

6653  
6654 In addition, some implementors have expressed an interest in using the charmap to define  
6655 certain other characteristics of codesets, such as the <mb\_cur\_max> value for the  
6656 particular codeset. (Note that <mb\_cur\_max> has to be equal to or lower than the C  
6657 Standard {MB\_LEN\_MAX}, which is the application limit). Such extensions are not  
6658 described here; but may be added in a later revision of this Technical Report.

6659  
6660 The <escape\_char> declaration was added at the request of the international community to  
6661 ease the creation of portable charmaps on terminals not implementing the default  
6662 backslash escape. (This approach was adopted because this is a new interface invented by  
6663 ISO/IEC 9945-2:1993 POSIX. Historical interfaces, such as the shell command language  
6664 and awk, have not been modified to accommodate this type of terminal.)  
6665

6666 The octal number notation was selected to match those of POSIX "awk" and "tr" utilities  
6667 and is consistent with that used by the POSIX localedef utility.  
6668

6669 The charmap capability implements a facility available at some X/Open compatible  
6670 applications. Its prime virtue is to support "generic" collation sequence source definitions.  
6671 An implementor or an applications developer can produce a template definition that can be  
6672 used to produce several codeset-dependent "compiled" FDCC-set definitions. The facility  
6673 also removes any dependency in many source definitions on characters outside the  
6674 character set defined in this clause.  
6675

6676 The charmap allows specification of more than one encoding of a character. This allows  
6677 for encodings that can encode items in more than one way. For example, an item can be  
6678 encoded once as a fully composed character and again as a base character plus combining  
6679 character. This would allow either representation to be recognized. As only the first  
6680 occurrence of the character may be output, this technique could be used to normalize a  
6681 character stream.  
6682

6683 The ISO 2022 support introduced gives the possibility to refer other definitions via  
6684 charmaps, so the full encoding does not have to be replicated. It supports shifting with G0,  
6685 G1, G2 and G3 sets, and also general shifting of coded character sets via escape  
6686 sequences.  
6687

### 6688 **B.3 Repertoiremap Rationale.** 6689

6690 The repertoiremap was introduced to make FDCC-sets independent of the availability of  
6691 charmaps. With the repertoiremap it is possible to use a FDCC-set encoded with one set  
6692 of symbolic character names, together with charmaps with other symbolic character  
6693 naming schemes, provided there are repertoiremaps available for both naming schemes.  
6694

6695 Repertoiremaps are also useful to describe repertoires of characters, to be used for  
6696 example for transliteration.  
6697

## 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

```

6723 % The following is the overall FDCC-set grammar
6724 FDCC_set_definition = [ global_statement* ] category+ ;
6725 global_statement = 'escape_char' SP char_symbol EOL
6726 | 'comment_char' SP char_symbol EOL
6727 | 'repertoiremap' SP quoted_string EOL
6728 | 'charmap' SP quoted_string EOL ;
6729 category = lc_identification | lc_ctype | lc_collate
6730 | lc_monetary | lc_numeric | lc_time
6731 | lc_messages | lc_xliterate | lc_telephone
6732 | lc_name | lc_address ;
6733
6734 % The following is the LC_IDENTIFICATION category grammar
6735 lc_ident = ident_head ident_keyword* ident_tail
6736 | ident_head copy_FDCC_set ident_tail ;
6737 ident_head = 'LC_IDENTIFICATION' EOL ;
6738 ident_keyword = ident_keyword_string SP quoted_string EOL ;
6739 ident_keyword_string = 'title' | 'source' | 'address' | 'contact'
6740 | 'email' | 'tel' | 'fax' | 'language'
6741 | 'territory' | 'audience' | 'application'
6742 | 'abbreviation' | 'revision' | 'date' ;
6743 ident_tail = 'END' SP 'LC_IDENTIFICATION' EOL ;
6744
6745
6746 % The following is the LC_CTYPE category grammar
6747 lc_ctype = ctype_head ctype_keyword* ctype_tail
6748 | ctype_head copy_FDCC_set ctype_tail ;
6749 ctype_head = 'LC_CTYPE' EOL ;
6750 ctype_keyword = charclass_keyword SP charclass_list EOL
6751 | charconv_keyword SP charconv_list EOL
6752 | 'width' SP width_list EOL ;
6753 charclass_keyword = 'upper' | 'lower' | 'alpha' | 'digit' |
6754 | 'alnum' | 'punct' | 'xdigit' | 'space' |
6755 | 'print' | 'graph' | 'blank' | 'cntrl' |
6756 | 'outdigit'
6757 | 'class' charclass_name semicolon ;
6758 charclass_name = "combining" | "combining_level3"
6759 | "" identifier "" ;

```

```

6760 charclass_list = charclass_list semicolon char_symbol
6761 | charclass_list semicolon ctype_abs_ellipsis
6762 semicolon char_symbol
6763 | charclass_list semicolon charsymbol
6764 ctype_symbolic_ellipses charsymbol
6765 | char_symbol ;
6766 width_list = charclass_list ':' number
6767 | width_list semicolon width_list ;
6768 charconv_keyword = 'toupper' | 'tolower'
6769 | 'map' ''' identifier ''' semicolon ;
6770 charconv_list = charconv_list semicolon charconv_entry
6771 | charconv_entry ;
6772 charconv_entry = '(' char_symbol comma char_symbol ')' ;
6773 ctype_symbolic_ellipses = '...' | '....' | '..(2)..' ;
6774 ctype_abs_ellipses = '...' ;
6775 ctype_tail = 'END' SP 'LC_TYPE' EOL ;
6776
6777 % The following is the LC_COLLATE category grammar
6778 lc_collate = collate_head collate_keywords collate_tail ;
6779 collate_head = 'LC_COLLATE' EOL ;
6780 collate_keywords = [ opt_statement* ] order_statements ;
6781 opt_statement = 'collating-symbol' SP collsymbol_list EOL
6782 | 'collating-element' SP collelement SP 'from'
6783 SP collelem_string EOL
6784 | 'section-symbol' space+ sectionsymbol EOL
6785 | 'copy' SP FDCC_set_name EOL
6786 | 'col_weight_max' SP number EOL
6787 | 'symbol-equivalence' SP collsymbol SP
6788 collsymbol ;
6789 collelem_string = ''' char_symbol+ ''' ;
6790 order_statements = order_start collation_order order_end ;
6791 order_start = 'order_start' SP section [ semicolon
6792 order_opts ] EOL
6793 | 'order_start' SP [ order_opts ] EOL ;
6794 order_opts = order_opt [ semicolon order_opt ] ;
6795 order_opt = order_opt [ comma opt_word ] ;
6796 opt_word = 'forward' | 'backward' | 'position' ;
6797 collation_order = collation_statement* ;
6798 collation_statement = collsymbol EOL
6799 | collating_element [ SP weight_list ] EOL ;
6800 collsymbol_list = collsymbol_element
6801 [ semicolon collsymbol_element ]* ;
6802 collsymbol_element = collsymbol
6803 | collsymbol SP ellipses SP collsymbol ;
6804 collating_element = char_symbol | collelement
6805 | ellipses | 'UNDEFINED' ;
6806 weight_list = weight_symbol [ semicolon weight_symbol ]* ;
6807 weight_symbol = <empty>
6808 | char_symbol
6809 | collsymbol
6810 | ''' elem_list '''
6811 | ''' symb_list ''' | 'IGNORE' ;
6812 ellipses = '...' | '..' | '....' ;
6813 reorder_after = 'reorder-after' SP collsymbol EOL ;
6814 reorder_end = 'reorder-end' EOL ;
6815 reorder_section_after = 'reorder-section-after' SP sectionsymbol SP
6816 sectionsymbol EOL ;
6817 reorder_section_end = 'reorder-section-end' EOL ;
6818 order_end = 'order_end' EOL ;
6819 collate_tail = 'END' SP 'LC_COLLATE' EOL ;
6820
6821 % The following is the LC_MESSAGES category grammar
6822 lc_messages = messages_head messages_keyword* messages_tail
6823 | messages_head copy_FDCC_set messages_tail ;
6824 messages_head = 'LC_MESSAGES' EOL ;
6825 messages_keyword = 'yesexpr' SP ''' extended_reg_expr ''' EOL
6826 | 'yesexpr' SP ''' extended_reg_expr ''' EOL ;
6827 messages_tail = 'END' SP 'LC_MESSAGES' EOL ;
6828
6829

```

```

6830 % The following is the LC_MONETARY category grammar
6831 lc_monetary = monetary_head monetary_keyword* monetary_tail
6832 | monetary_head copy_FDCC_set monetary_tail ;
6833 monetary_head = 'LC_MONETARY' EOL ;
6834 monetary_keyword = mon_keyword_string SP quoted_string EOL
6835 | mon_keyword_strings SP mon_string_list EOL
6836 | mon_keyword_char SP mon_number_list EOL
6837 | mon_keyword_date SP mon_date_list EOL
6838 | 'conversion_rate' SP mon_conv_list EOL
6839 | 'mon_grouping' SP mon_group_list EOL ;
6840 mon_keyword_string = 'mon_decimal_point' | 'mon_thousands_sep'
6841 | 'positive_sign' | 'negative_sign' ;
6842 mon_keyword_strings = 'int_curr_symbol' | 'currency_symbol' ;
6843 mon_keyword_char = 'int_frac_digits' | 'frac_digits'
6844 | 'p_cs_precedes' | 'p_sep_by_space'
6845 | 'n_cs_precedes' | 'n_sep_by_space'
6846 | 'int_p_cs_precedes' | 'int_p_sep_by_space'
6847 | 'int_n_cs_precedes' | 'int_n_sep_by_space'
6848 | 'p_sign_posn' | 'n_sign_posn'
6849 | 'int_p_sign_posn' | 'int_n_sign_posn' ;
6850 mon_keyword_date = 'valid_from' | 'valid_to' ;
6851 mon_date_list = mon_date | mon_date_list semicolon mon_date ;
6852 mon_date = "' 8 * digit '" ;
6853 mon_group_list = number | mon_group_list semicolon number ;
6854 mon_string_list = quoted_string [ semicolon quoted_string]* ;
6855 mon_number_list = mon_number | mon_number_list semicolon
6856 mon_number ;
6857 mon_number = number | -1 ;
6858 mon_conv_list = mon_pair | mon_conv_list semicolon mon_pair ;
6859 mon_pair = number spaces* '/' spaces* number ;
6860 monetary_tail = 'END' SP 'LC_MONETARY' EOL ;
6861
6862 % The following is the LC_NUMERIC category grammar
6863 lc_numeric = numeric_head numeric_keyword* numeric_tail
6864 | numeric_head copy_FDCC_set numeric_tail ;
6865 numeric_head = 'LC_NUMERIC' EOL ;
6866 numeric_keyword = num_keyword_string SP quoted_string EOL
6867 | num_keyword_grouping SP num_group_list EOL ;
6868 num_keyword_string = 'decimal_point' | 'thousands_sep' ;
6869 num_keyword_grouping = 'grouping' ;
6870 num_group_list = number
6871 | num_group_list semicolon number ;
6872 numeric_tail = 'END' SP 'LC_NUMERIC' EOL ;
6873
6874 % The following is the LC_TIME category grammar
6875 lc_time = time_head time_keyword* time_tail
6876 | time_head copy_FDCC_set time_tail ;
6877 time_head = 'LC_TIME' EOL ;
6878 time_keyword = time_keyword_name SP time_list EOL
6879 | time_keyword_fmt SP quoted_string EOL
6880 | time_keyword_opt SP time_list EOL
6881 | 'week' SP number semicolon mon_date semicolon
6882 number EOL
6883 | time_keyword_num SP number EOL
6884 | 'timezone' SP time_list EOL ;
6885 time_keyword_name = 'abday' | 'day' | 'abmon' | 'mon' | 'am_pm' ;
6886 time_keyword_fmt = 'd_t_fmt' | 'd_fmt' | 't_fmt' | 't_fmt_ampm' ;
6887 time_keyword_opt = 'era' | 'era_year' | 'era_d_fmt' | 'alt_digits'
6888 | era_d_t_fmt | era_t_fmt ;
6889 time_keyword_week = 'week' ;
6890 time_keyword_num = 'first_weekday' | 'first_workday'
6891 | 'cal_direction' ;
6892 time_list = time_list semicolon quoted_string
6893 | quoted_string ;
6894 time_tail = 'END' SP 'LC_TIME' EOL ;
6895
6896
6897
6898
6899

```

```

6900 % The following is the LC_XLITERATE category grammar
6901 lc_xliterate = translit_head [translit_include]
6902 [default_missing] translit_statement*
6903 translit_tail | translit_head copy_FDCC_set
6904 translit_tail ;
6905 translit_head = 'LC_XLITERATE' EOL ;
6906 translit_include = 'include' SP FDCC_set_name semicolon
6907 quoted_nonempty_string EOL ;
6908 default_missing = 'default_missing' SP quoted_string EOL ;
6909 translit_ignore = 'translit_ignore' SP charclass_list EOL ;
6910 translit_statement = char_or_string SP char_or_string [ semicolon
6911 char_or_string ]* EOL ;
6912 translit_tail = 'END' SP 'LC_XLITERATE' EOL ;
6913
6914 % The following is the LC_NAME category grammar
6915 lc_name = name_head name_keyword* name_tail
6916 | name_head copy_FDCC_set name_tail ;
6917 name_head = 'LC_NAME' EOL ;
6918 name_keyword = name_keyword_string SP quoted_string EOL ;
6919 name_keyword_string = 'name_fmt' | 'name_gen' | 'name_mr'
6920 | 'name_mrs' | 'name_ms' | 'name_miss'
6921 | 'name_ms' ;
6922 name_tail = 'END' SP 'LC_NAME' EOL ;
6923
6924 % The following is the LC_ADDRESS category grammar
6925 lc_address = address_head address_keyword* address_tail
6926 | address_head copy_FDCC_set address_tail ;
6927 address_head = 'LC_ADDRESS' EOL ;
6928 address_keyword = address_keyword_string SP quoted_string EOL ;
6929 address_keyword_string = 'postal_fmt' | 'country_name' |
6930 'country_post' | 'lang_name' | 'lang_ab2' |
6931 'lang_ab3_term' | 'lang_ab3_lib' ;
6932 address_tail = 'END' SP 'LC_ADDRESS' EOL ;
6933
6934 % The following is the LC_TELEPHONE category grammar
6935 lc_tel = tel_head tel_keyword* tel_tail
6936 | tel_head copy_FDCC_set tel_tail ;
6937 tel_head = 'LC TELEPHONE' EOL ;
6938 tel_keyword = tel_keyword_string SP quoted_string EOL ;
6939 tel_keyword_string = 'tel_int_fmt' | 'tel_dom_fmt' | 'int_select'
6940 | 'int_prefix' ;
6941 tel_tail = 'END' SP 'LC TELEPHONE' EOL ;
6942
6943 % The following grammar rules are common to all categories
6944 char = <any character except those that makes an End
6945 Of Line>
6946 graphic_char = <any char except control_chars and space> ;
6947 space = ' ' | <TAB> ;
6948 SP = space+ ;
6949 EOL = end_of_line | comment end_of_line ;
6950 end_of_line = <anything that makes an End Of Line (EOL) in
6951 the operating system employed> ;
6952 comment_char = <defined by the 'comment_char' keyword> ;
6953 escape_char = <defined by the 'escape_char' keyword> ;
6954 charsymbol = simple_symbol | ucs_symbol ;
6955 collsymbol = simple_symbol ;
6956 collelement = simple_symbol ;
6957 sectionsymbol = simple_symbol ;
6958 octdigit = '0' | '1' | '2' | '3' | '4' | '5' | '6' | '7' ;
6959 digit = '0' | '1' | '2' | '3' | '4' | '5' | '6' | '7' | '8' | '9' ;
6960 hex_upper = 'A' | 'B' | 'C' | 'D' | 'E' | 'F' | digit ;
6961 hexdigit = hex_upper | 'a' | 'b' | 'c' | 'd' | 'e' | 'f' ;
6962 letter = 'a' | 'b' | 'c' | 'd' | 'e' | 'f' | 'g' | 'h' | 'i' | 'j' | 'k'
6963 | 'l' | 'm' | 'n' | 'o' | 'p' | 'q' | 'r' | 's' |
6964 | 't' | 'u' | 'v' | 'w' | 'x' | 'y' | 'z' | 'A' | 'B' | 'C' | 'D'
6965 | 'E' | 'F' | 'G' | 'H' | 'I' | 'J' | 'K' | 'L' | 'M' | 'N' | 'O'
6966 | 'P' | 'Q' | 'R' | 'S' | 'T' | 'U' | 'V' | 'W' | 'X' | 'Y' | 'Z' ;
6967 portable_graph_gtr = letter | digit | '!' | '"' | '#' | '$' | '%' | '&'
6968 | "'" | '(' | ')' | '*' | '+' | '-' | '.' | ':' | ';'
6969 | '<' | '=' | '?' | '@' | '[' | '\' | ']' | '^' | '_'

```

```

6970 | ' ' | '{' | '|' | '}' | '~' ;
6971 portable_graph = portable_graph_gtr | '>' ;
6972 portable_char = portable_graph | ' ' | <NUL> | <ALERT>
6973 | <BACKSPACE> | <TAB> | <CARRIAGE_RETURN>
6974 | <NEWLINE> | <VERTICAL_TAB> | <FORM_FEED> ;
6975 octal_char = escape_char octdigit octdigit octdigit* ;
6976 hex_char = escape_char 'x' hexdigit hexdigit hexdigit* ;
6977 decimal_char = escape_char 'd' digit digit digit* ;
6978 number = digit+ ;
6979 id_part = letter | digit | '-' | '_' ;
6980 four_digit_hex_string = hex_upper hex_upper hex_upper hex_upper ;
6981 identifier = letter id_part* ;
6982 simple_symbol = space* '<' portable_graph_gtr+ '>' ;
6983 ucs_symbol = space* '<U' four_digit_hex_string
6984 [ four_digit_hex_string ] '>' ;
6985 quoted_string = '"' char_symbol* '"' ;
6986 quoted_nonempty_string = '"' char_symbol+ '"' ;
6987 char_symbol = char | charsymbol
6988 | octal_char | hex_char | decimal_char ;
6989 elem_list = elem+ ;
6990 elem = char_symbol | collsymbol | collelement ;
6991 symb_list = collsymbol+ ;
6992 FDCC_set_name = FDCC-name | '"' FDCC-name '"' ;
6993 copy_FDCC_set = 'copy' FDCC_set_name EOL ;
6994 FDCC-name = portable_graph+ ;
6995 semicolon = space* ';' space* ;
6996 comma = space* ',' space* ;
6997 comment = comment_char char* ;
6998
6999
7000
7001
7002
7003
7004
7005
7006
7007
7008
7009
7010
7011
7012
7013
7014
7015
7016
7017
7018
7019
7020
7021
7022
7023
7024
7025
7026
7027
7028
7029
7030
7031
7032
7033
7034
7035
7036

```

7037  
7038  
7039  
7040  
7041  
7042  
7043  
7044  
7045  
7046  
7047  
7048  
7049  
7050  
7051  
7052  
7053  
7054  
7055  
7056  
7057  
7058  
7059  
7060  
7061  
7062  
7063  
7064  
7065  
7066  
7067  
7068  
7069  
7070  
7071  
7072  
7073  
7074  
7075  
7076  
7077  
7078  
7079  
7080  
7081  
7082  
7083  
7084  
7085  
7086  
7087  
7088

## **Annex D** (informative)

### **Issues list**

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 is decided to be a technical report in order to give information to public earlier.

The issues includes but are not limited to:

- 1) Whether the features which have their origin in ISO/IEC 9945-2 --POSIX Part 2 -- works well after its separation from ISO/IEC 9945-2 or not.
- 2) Whether it makes sense or not to have a default value, which may be considered as a recommendation, for each cultural convention item.
- 3) Whether each specification form fits for world-wide cultural variations or not.

The preparer of this report, ISO/IEC JTC1/SC22, expects the rapid progress of internationalization in the field of information technology will solve the above mentioned issues and this technical report will be used as a base for a new standard in 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. has repeatedly reviewed this document, and is firmly of the opinion that it should not be published as an international standard. The reasons for that assessment include, but are not limited to:

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", and in fact exacerbates the weaknesses of POSIX 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.



7089 2. In an attempt to extend the POSIX LC\_CTYPE specification to cover the repertoire of  
7090 ISO/IEC 10646-1, this document blunders badly in asserting the cultural contextualization  
7091 of character properties for the UCS. The treatment of LC\_CTYPE as part of locales is an  
7092 artifact of POSIX architecture and results from the need to have a place to put locale  
7093 differences for case mapping. But by cloning other character properties having nothing to  
7094 do with case mapping into LC\_CTYPE, the net effect is to create a second source for  
7095 UCS character properties, with attendant dangers of divergence and errors, and with  
7096 inevitable difficulties of maintenance and versioning. The clear intent to force other ISO  
7097 standards to obtain their character property definitions from this document, instead of by  
7098 reference to the widely implemented UCS property tables published by the Unicode  
7099 Consortium, would lead to confusion and interoperability problems for character properties  
7100 -- just the opposite of what a standard should be doing.

7101 3. Each of the categories in the FDCC-set description has unaddressed problems and  
7102 limitations. Rather than being resolved during the development of this document, many of  
7103 these limitations were simply asserted to be "requirements" or necessary limitations. And  
7104 it appears to us that those are limitations of a particular envisioned implementation, rather  
7105 than following from the legitimate needs for specification of cultural conventions. Because  
7106 of this, implementers attempting to make use of the FDCC-set categories are immediately  
7107 faced with an unexplained host of problems and mismatches to the actual cultural  
7108 adaptability that they are trying to implement to meet customer needs for information  
7109 technology.

7110 4. In particular, the LC\_MONETARY extensions to deal with euro sign dual formatting  
7111 requirements seem to be an unnecessarily complicated scheme rolled out too late -- and do  
7112 not follow the approach already taken by production software to solve the problem.

7113 5. This document contains a completely unnecessary repertoire map definition intended to  
7114 promulgate a particularly bad collection of character mnemonic short strings. The U.S.  
7115 views these "mnemonics" as confusing and irrelevant. The need for short identifiers for  
7116 characters can be met by the standard short identifiers spelled out in ISO/IEC 10646,  
7117 which \*are\* in widespread use.

7118 6. There are numerous errors of detail in this document. While these could, in principle,  
7119 be addressed, many have not been. On that basis alone, it seems inadvisable to make the  
7120 document a standard.

7121 The U.S. does not share the optimistic assessment of the usefulness of this document as a  
7122 "trial" mechanism nor of the ease of addressing the issues mentioned here.

7123  
7124  
7125  
7126  
7127  
7128

		<b>Annex E</b>	
		(informative)	
		<b>Index</b>	
7129			
7130			
7131			
7132			
7133			
7134	abbreviation	4.2	col_weight_max 4.4, 4.4.3
7135	abday	4.7	collating-element 4.4
7136	abmon	4.7	collating statements 4.4.1
7137	absolute ellipses	4.3	collating-symbol 4.4.6
7138	address	4.2	collating element 3.1.13
7139	addresses	4.11	collating sequence 3.1.15
7140	addset	5.1	collating-element 4.4.5
7141	alpha	4.3.1	collating-symbol 4.4
7142	alt_digits	4.7	collation 3.1.12
7143	am_pm	4.7	combining 4.3.1
7144	application	4.2	combining_level3 4.3.1
7145	audience	4.2	comment_char 4.1.4.1, 5.1
7146	blank	4.3.1	conformance 7
7147	block_separator	4.3.1	contact 4.2
7148	byte	3.1.1	continuation line 4.1.2
7149	cal_direction	4.7	control characters 4.3.1
7150	category	4.2	conversion_rate 4.5
7151	category names	4.1	copy 4.1.3, 4.2, 4.3.1, 4.4.2,
7152	category trailer	4.1	4.5, 4.6, 4.7, 4.8, 4.9, 4.10, 4.11, 4.12
7153	category header	4.1	country_ab2 4.11
7154	category body	4.1	country_ab3 4.11
7155	character	3.1.2	country_car 4.11
7156	character, graphic	4.3.1	country_isbn 4.11
7157	character, special	4.3.1	country_name 4.11
7158	character representation	4.1.1	country_num 4.11
7159	character, native digit	4.3.1	country_post 4.11
7160	character, hexadecimal digit	4.3.1	cultural convention 3.1.5
7161	character, multibyte	4.1.1	currency_symbol 4.5
7162	character, decimal constant	4.1.1	d_fmt 4.7
7163	character, hexadecimal constant	4.1.1	d_t_fmt 4.7
7164	character, space	4.3.1	date field descriptors 4.7.1
7165	character, octal constant	4.1.1	date 4.2
7166	character, control	4.3.1	day 4.7
7167	character, blank	4.3.1	decimal_point 4.6
7168	character, digit	4.3.1	default_missing 4.3.2
7169	character, punctuation	4.3.1	definitions 3.1
7170	character, printable	3.1.10	digit 4.3.1
7171	character class	3.1.9	ellipses 4.3, 4.4.1, 5.1
7172	character, coded	3.1.3	ellipses, absolute 4.3, 4.4.1
7173	Character set rationale	B.2	ellipses, symbolic 4.3, 4.4.1, 5.1
7174	charmap text	5.1	email 4.2
7175	charmap 5, 4.1.4.4, 3.1.7		equivalence class 3.1.16
7176	charmap rationale	B.2	era 4.7
7177	class	4.3.1	era_d_fmt 4.7
7178	cntrl	4.3.1	era_year 4.7
7179	code_set_name	5.1	escape_char 4.1.4.2, 5.1.6
7180	coded character	3.1.3	esqseq 5.1

7181	euro	B.1.3	extended regular expression	4.8	LC_XLITERATE	4.9
7182	fax			4.2	LC_XLITERATE rationale	B.1.8
7183	FDCC-set, definition			4.1	LC_X_	4
7184	FDCC-set			4f	line continuation	4.1.4
7185	FDCC-set			3.1.6	lower	4.3.1
7186	FDCC-set rationale			B.1	map	4.3.1
7187	first_weekday			4.7	mb_cur_max	5.1
7188	first_workday			4.7	mb_cur_min	5.1
7189	frac_digits			4.5	messages	4.8
7190	graph			4.3.1	modified date field descriptors	4.7.2
7191	graphic characters			4.3.1	mon	4.7
7192	grouping			4.6	mon_decimal_point	4.5
7193	height			4.9	mon_grouping	4.5
7194	include			4.3.2	mon_thousands_sep	4.5
7195	include			5.1	monetary	4.5
7196	include			4.3.2.2	multicharacter collating element	3.1.14
7197	int_curr_symbol			4.5	n_cs_precedes	4.5
7198	int_frac_digits			4.5	n_sep_by_space	4.5
7199	int_n_cs_precedes			4.5	n_sign_posn	4.5
7200	int_n_sep_by_space			4.5	name formatting	4.10
7201	int_n_sign_posn			4.5	name_fmt	4.10
7202	int_p_cs_precedes			4.5	name_gen	4.10
7203	int_p_sep_by_space			4.5	name_miss	4.10
7204	int_p_sign_posn			4.5	name_mr	4.10
7205	int_prefix			4.12	name_mrs	4.10
7206	int_select			4.12	name_ms	4.10
7207	keywords			4.1	negative_sign	4.5
7208	lang_ab			4.11	noexpr	4.8
7209	lang_lib			4.11	notations	3.2
7210	lang_name			4.11	numeric	4.6
7211	lang_term			4.11	operands	4.1
7212	language			4.2	order_end	4.4.9, 4.4
7213	LC_ADDRESS			4.11	order_start	4.4, 4.4.8
7214	LC_ADDRESS rationale			B.1.10	outdigit	4.3.1
7215	LC_COLLATE			4.4	p_cs_precedes	4.5
7216	LC_COLLATE rationale			B.1.3	p_sep_by_space	4.5
7217	LC_CTYPE			4.3	p_sign_posn	4.5
7218	LC_CTYPE rationale			B.1.2	paper format	4.9
7219	LC_IDENTIFICATION			4.2	portable character set	3.2.4
7220	LC_IDENTIFICATION rationale			B.1.1	positive_sign	4.5
7221	LC_MESSAGES			4.8	POSIX	1
7222	LC_MESSAGES rationale			B.1.7	POSIX differences	A
7223	LC_MONETARY			4.5	POSIX conformance	4.2
7224	LC_MONETARY rationale			B.1.4	postal addresses	4.11
7225	LC_NAME			4.10	postal_fmt	4.11
7226	LC_NAME rationale			B.1.9	pre-category statements	4.1.4
7227	LC_NUMERIC			4.6	print	4.3.1
7228	LC_NUMERIC rationale			B.1.5	printable character	3.1.10
7229	LC_TELEPHONE			4.12	punct	4.3.1
7230	LC_TELEPHONE rationale			B.1.11	punctuation characters	4.3.1
7231	LC_TIME			4.7	redefine	4.3.2
7232	LC_TIME rationale			B.1.6	references	2

7233	reorder-section-end	4.4.13
7234	reorder-section-after	4.4.12
7235	reorder-section-after	4.4
7236	reorder-after	4.4
7237	reorder-end	4.4
7238	reorder-section-end	4.4
7239	reorder-after	4.4.10
7240	reorder-end	4.4.11
7241	reorder-after rationale	B.1.2.1
7242	repertoire rationale	B.3
7243	repertoire	6
7244	repertoiremap	6, 3.1.8, 5.1, 4.1.4.3
7245	revision	4.2
7246	scope	1
7247	section	4.4, 4.4.4
7248	source	4.2
7249	space	4.3.1
7250	special characters	4.3.1
7251	symbol-equivalence	4.4, 4.4.7
7252	symbolic ellipses	4.3, 5.1
7253	symbolic name	4.1.1
7254	syntax format	3.2.1
7255	t_fmt	4.7
7256	t_fmt_ampm	4.7
7257	tel	4.2
7258	tel_dom_fmt	4.12
7259	tel_int_fmt	4.12
7260	telephone numbers	4.12
7261	territory	4.2
7262	text file	3.1.4
7263	thousands_sep	4.6
7264	timezone	4.7
7265	title	4.2
7266	tolower	4.3.1
7267	toupper	4.3.1
7268	translit_ignore	4.9
7269	transliteration	4.9
7270	transliteration statements	4.9.1
7271	upper	4.3.1
7272	valid_from	4.5
7273	valid_to	4.5
7274	visible glyph portable characters	3.2.4
7275	week	4.7
7276	white space	3.1.11
7277	width	4.9
7278	xdigit	4.3.1
7279	yesexpr	4.8
7280		
7281		

**BIBLIOGRAPHY**

- 7282  
7283  
7284 The following specifications are considered relevant to this Technical Report, in addition  
7285 to the normative references.  
7286  
7287 CEPT, CEPT-MAILCODE, *Country code for mail*.  
7288  
7289 ISO 646, *Information technology - ISO 7-bit coded character set for information inter-*  
7290 *change*.  
7291  
7292 ISO/IEC 9899, *Information technology - Programming language C*.  
7293  
7294 ISO/IEC 14977, *Information technology - Syntactic metalanguage - Extended BNF*.  
7295  
7296 The Unicode Consortium: *The Unicode Standard, Version 2.0*, Addison Wesley  
7297 Developers Press, July 1996. ISBN 0-201-48345-9.  
7298  
7299 IBM: *National Language Design Guide Volume 2 - National Language Support Reference*  
7300 *Manual*, IBM SE09-8002-03, August 1994.  
7301  
7302 STRÍ: *Nordic Cultural Requirements on Information Technology (Summary report)*, STRÍ  
7303 TS3, Libris, Reykjavík, Iceland 1992. ISBN 9979-9004-3-1.  
7304  
7305