Network Working Group
Request for Comments: 3454
Category: Standards Track

P. Hoffman
IMC & VPNC
M. Blanchet
Viagenie

December 2002

# Preparation of Internationalized Strings ("stringprep")

## Status of this Memo



## Copyright Notice



## Abstract

This document describes a framework for preparing Unicode text strings in order to increase the likelihood that string input and string comparison work in ways that make sense for typical users throughout the world. The stringprep protocol is useful for protocol identifier values, company and personal names, internationalized domain names, and other text strings.

This document does not specify how protocols should prepare text strings. Protocols must create profiles of stringprep in order to fully specify the processing options.

## Table of Contents

## 1. Introduction

Application programs can display text in many different ways. Similarly, a user can enter text into an application program in a myriad of fashions. Internationalized text (that is, text that is not restricted to the narrow set of US-ASCII characters) has many input and display behaviors that make it difficult to compare text in a consistent fashion.

This document specifies a framework of processing rules for Unicode text. Other protocols can create profiles of these rules; these profiles will allow users to enter internationalized text strings in applications and have the highest chance of getting the content of the strings correct. In this case, "correct" means that if two different people enter what they think is the same string into two different input mechanisms, the strings should match on a character-by-character basis.

This framework does not describe how data is transcoded from other character sets into Unicode. In systems that uses non-Unicode character sets, the transcoding algorithm is a critical part of enabling secure and "correct" operation of internationalized text strings.

In addition to helping string matching, profiles of stringprep can also exclude characters that should not normally appear in text that is used in the protocol. The profile can prevent such characters by changing the characters to be excluded to other characters, by removing those characters, or by causing an error if the characters would appear in the output. For example, because the backspace character can cause unpredictable display results, a profile can specify that a string containing a backspace character would cause an error.

A profile of stringprep converts a single string of input characters to a string of output characters, or returns an error if the output string would contain a prohibited character. Stringprep profiles cannot both emit a string and return an error.

Stringprep profiles cannot account for all of the variations that might occur or that a user might expect. In particular, a profile will not be able to account for choice of spellings in all languages for all scripts because the number of alternative spellings of words and phrases is immense. Users would probably expect all spelling equivalents to be made equivalent, or none of them to be. Examples of spelling equivalents include "theater" vs. "theatre", and "hemoglobin" vs. "h<U+00E6>moglobin" in American vs. British English. Other examples are simplified Chinese spellings of names (for

example,"<U+7EDF><U+4E00><U+7801>") vs. the equivalent traditional Chinese spelling (for example, "<U+7D71><U+4E00><U+78BC>"). Language-specific equivalences such as "Aepfel" vs. "<U+00C4>pfel", which are sometimes considered equivalent in German, may not be considered equivalent in other languages.

## 1.1 Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14, RFC 2119 [RFC2119].

Note: A glossary of terms used in Unicode and ISO/IEC 10646 can be found in [Glossary]. Information on the 10646/Unicode character encoding model can be found in [CharModel].

Character names in this document use the notation for code points and names from the Unicode Standard [Unicode3.2] and ISO/IEC 10646 [ISO10646]. For example, the letter "a" may be represented as either "U+0061" or "LATIN SMALL LETTER A". In the lists of mappings and the prohibited characters, the "U+" is left off to make the lists easier to read. The comments for character ranges are shown in square brackets (such as "[CONTROL CHARACTERS]") and do not come from the standards.

## 1.2 Using stringprep in protocols

The stringprep protocol does not stand on its own; it has to be used by other protocols at precisely-defined places in those other protocols. For example, a protocol that has strings that come from the entire ISO/IEC 10646 [ISO10646] character repertoire might specify that only strings that have been processed with a particular profile of stringprep are legal. Another example would be a protocol that does string comparison as a step in the protocol; that protocol might specify that such comparison is done only after processing the strings with a specific profile of stringprep.

When two protocols that use different profiles of stringprep interoperate, there may be conflict about what characters are and are not allowed in the final string. Thus, protocol developers should strongly consider re-using existing profiles of stringprep.

When developers wish to allow users as wide of a range of characters as possible in input text strings, they should, where possible, cause stringprep to convert characters from the input string to a canonical form instead of prohibiting them.

Although it would be easy to use the stringprep process to "correct" perceived mis-features or bugs in the current character standards, stringprep profiles SHOULD NOT do so.

A profile of stringprep can create tables different from those in the appendixes of this document, but it will be an exception when they do. The intention of stringprep is to define the tables and have the profiles of stringprep select among those defined tables.

A profile of stringprep MUST include all of the following:

- The intended applicability of the profile
- The character repertoire that is the input and output to stringprep (which is Unicode 3.2 for this version of stringprep)
- The mapping tables from this document used (as described in section 3)
- Any additional mapping tables specific to the profile
- The Unicode normalization used, if any (as described in section 4)
- The tables from this document of characters that are prohibited as output (as described in section 5)
- The bidirectional string testing used, if any (as described in section 6)
- Any additional characters that are prohibited as output specific to the profile

Each profile MUST state the character repertoire on which the profile will operate. Appendix A lists the Unicode repertoires that can be selected. No repertoire is ever complete, and it is expected that characters will be added to the Unicode repertoire for the foreseeable future. Section 7 of this document describes how to handle characters that are assigned in later versions of the Unicode repertories. Subsections of appendix A also list unassigned code points for each repertoire.

This document is for Unicode version 3.2, and should not be considered to automatically apply to later Unicode versions. The IETF, through an explicit standards action, may update this document as appropriate to handle later Unicode versions.

This document lists the unassigned code points in the range 0 to 10FFFF for Unicode 3.2 in appendix A. The list in appendix A MUST be used by implementations of this specification. If there are any discrepancies between the list in appendix A and the Unicode 3.2 specification, the list in appendix A always takes precedence.

Each profile of stringprep MUST be registered with IANA. The registration procedure is described in the IANA Considerations appendix; basically, the IESG must review each profile of stringprep. Protocol developers are strongly encouraged to look through the IANA profile registry when creating new profiles for stringprep, and to re-use logic from earlier profiles where possible in new profiles. In some cases, an existing profile can be reused by a different protocol.

## 2. Preparation Overview

The steps for preparing strings are:

1) Map -- For each character in the input, check if it has a mapping and, if so, replace it with its mapping. This is described in section 3.

2) Normalize -- Possibly normalize the result of step 1 using Unicode normalization. This is described in section 4.

3) Prohibit -- Check for any characters that are not allowed in the output. If any are found, return an error. This is described in section 5.

4) Check bidi -- Possibly check for right-to-left characters, and if any are found, make sure that the whole string satisfies the requirements for bidirectional strings. If the string does not satisfy the requirements for bidirectional strings, return an error. This is described in section 6.

The above steps MUST be performed in the order given to comply with this specification.

The mappings described in section 3, and the optional Unicode normalization described in section 4, can be one-to-none, one-to-one, one-to-many, many-to-one, or many-to-many. That is, some characters might be eliminated or replaced by more than one character, and the output of this step might be shorter or longer than the input. Because of this, the system using stringprep MUST be prepared to receive a longer or shorter string than the one input in the stringprep algorithm.

## 3. Mapping

Each character in the input stream MUST be checked against a mapping table. The mapping table SHOULD come from this document, although the mapping table MAY be added to or altered by the profile. The mapping tables are subsections of appendix B.

The lists in appendix B MUST be used by implementations of this specification. If there are any discrepancies between the lists in appendix B and subsections below, the lists in appendix B always takes precedence.

For any individual character, the mapping table MAY specify that a character be mapped to nothing, or mapped to one other character, or mapped to a string of other characters.

Mapped characters are not re-scanned during the mapping step. That is, if character A at position X is mapped to character B, character B which is now at position X is not checked against the mapping table.

### 3.1 Commonly mapped to nothing

The following characters are simply deleted from the input (that is, they are mapped to nothing) because their presence or absence in protocol identifiers should not make two strings different. They are listed in Table B.1.

Some characters are only useful in line-based text, and are otherwise invisible and ignored.

00AD; SOFT HYPHEN
1806; MONGOLIAN TODO SOFT HYPHEN
200B; ZERO WIDTH SPACE
2060; WORD JOINER
FEFF; ZERO WIDTH NO-BREAK SPACE

Some characters affect glyph choice and glyph placement, but do not bear semantics.

034F; COMBINING GRAPHEME JOINER
180B; MONGOLIAN FREE VARIATION SELECTOR ONE
180C; MONGOLIAN FREE VARIATION SELECTOR TWO
180D; MONGOLIAN FREE VARIATION SELECTOR THREE
200C; ZERO WIDTH NON-JOINER
200D; ZERO WIDTH JOINER
FE00; VARIATION SELECTOR-1
FE01; VARIATION SELECTOR-2

FE02; VARIATION SELECTOR-3
FE03; VARIATION SELECTOR-4
FE04; VARIATION SELECTOR-5
FE05; VARIATION SELECTOR-6
FE06; VARIATION SELECTOR-7
FE07; VARIATION SELECTOR-8
FE08; VARIATION SELECTOR-9
FE09; VARIATION SELECTOR-10
FE0A; VARIATION SELECTOR-11
FE0B; VARIATION SELECTOR-12
FE0C; VARIATION SELECTOR-13
FE0D; VARIATION SELECTOR-14
FE0E; VARIATION SELECTOR-15
FE0F; VARIATION SELECTOR-16

## 3.2 Case folding

If a profile is going to map characters for case-insensitive comparison, that profile SHOULD map using either appendix B.2 or appendix B.3. appendix B.2 is for profiles that also use Unicode normalization form KC, while appendix B.3 is for profiles that do not use Unicode normalization. These tables map from uppercase to lowercase characters. Note that this could have been "change all lowercase characters into uppercase characters". However, the upper-to-lower folding was chosen because there is a tradition of using lowercase in current Internet applications and protocols.

If a profile creates its own mapping tables for case folding, they SHOULD be based on [UTR21], and SHOULD map from uppercase characters to lowercase. The "CaseFolding.txt" file from the Unicode database SHOULD be used to prepare the mapping table. The profile SHOULD do full case mapping (that is, using statuses C, F, and I).

If the profile is using Unicode normalization form KC (as described in section 4 of this document), it is important to note that there are some characters that do not have mappings in [UTR21] but still need processing. These characters include a few Greek characters and many symbols that contain Latin characters. The list of characters to add to the mapping table can determined by the following algorithm:

```
b = NormalizeWithKC(Fold(a));
c = NormalizeWithKC(Fold(b));
if c is not the same as b, add a mapping for "a to c".
```

Because NormalizeWithKC(Fold(c)) always equals c, the table is stable from that point on.

Appendix B.3 is derived from the CaseFolding-3.txt file associated with Unicode 3.2; appendix B.2 is based on appendix B.3 with the additional characters added from the algorithm above.

Authors of profiles of this document need to consider the effects of changing the mapping of any currently-assigned character when updating their profiles. Adding a new mapping for a currently-assigned character, or changing an existing mapping, could cause a variance between the behavior of systems that have been updated and systems that have not been updated.

## 4. Normalization

The output of the mapping step is optionally normalized using one of the Unicode normalization forms, as described in [UAX15]. A profile can specify one of two options for Unicode normalization:

- no normalization

- Unicode normalization with form KC

A profile MAY choose to do no normalization. However, such a profile can easily yield results that will be surprising to typical users, depending on the input mechanism they use. For example, some input mechanisms enter compatibility characters that look exactly like the underlying characters, but have different code points. Another example of where Unicode normalization helps create predictable results is with characters that have multiple combining diacritics: normalization orders those diacritics in a predictable fashion.

On the other hand, Unicode normalization requires fairly large tables and somewhat complicated character reordering logic. The size and complexity should not be considered daunting except in the most restricted of environments, and needs to be weighed against the problems of user surprise from comparing unnormalized strings. Note that the tables used for normalization are not given in this document, but instead must be derived from the Unicode database, as described in [UAX15].

There is a third form of normalization, Unicode normalization with form C. If a profile is going to use a Unicode normalization, it MUST use Unicode normalization form KC. Form KC maps many "compatibility characters" to their equivalents. Some user interface systems make it possible to enter compatibility characters instead of the base equivalents. Thus, using form KC instead of form C will cause more strings that users would expect to match to actually match.

A profile that specifies Unicode normalization MUST use the normalization in [UAX15] that is associated with the version of the Unicode character set specified for the profile.

The composition process described in [UAX15] requires a fixed composition version of Unicode to ensure that strings normalized under one version of Unicode remain normalized under all future versions of Unicode.

The IETF is relying on Unicode not to change the normalization of currently-assigned characters in future versions of normalization. If a future version of the normalization tables changes the normalized value of an existing character, authors of profiles of this document have to look at the changes very carefully before they update their normalization tables. Such a change could cause a variance between the behavior of systems that have been updated and systems that have not been updated.

## 5. Prohibited Output

Before the text can be emitted, it MUST be checked for prohibited code points. There are a variety of prohibited code points, as described in this section. A profile of this document MAY use all or some of the tables in appendix C.

The stringprep process never emits both an error and a string. If an error is detected during the checking for prohibited code points, only an error is returned.

Note that the subsections below describe how the tables in appendix C were formed. They are here for people who want to understand more, but they should be ignored by implementors. Implementations that use tables MUST map based on the tables themselves, not based on the descriptions in this section of how the tables were created.

The lists in appendix C MUST be used by implementations of this specification. If there are any discrepancies between the lists in appendix C and subsections below, the lists in appendix C always take precedence.

Some code points listed in one section may also appear in other sections.

It is important to note that a profile of this document MAY prohibit additional characters.

Each subsection of this section has a matching subsection in appendix C. For example, the characters listed in section 5.1 are listed in appendix C.1.

## 5.1 Space characters

Space characters can make accurate visual transcription of strings nearly impossible and could lead to user entry errors in many ways. Note that the list below is split into two tables in appendix C: Table C.1.1 contains the ASCII code points, while Table C.1.2 contains the non-ASCII code points. Most profiles of this document that want to prohibit space characters will want to include both tables.

```
0020; SPACE
00A0; NO-BREAK SPACE
1680; OGHAM SPACE MARK
2000; EN QUAD
2001; EM QUAD
2002; EN SPACE
2003; EM SPACE
2004; THREE-PER-EM SPACE
2005; FOUR-PER-EM SPACE
2006; SIX-PER-EM SPACE
2007; FIGURE SPACE
2008; PUNCTUATION SPACE
2009; THIN SPACE
200A; HAIR SPACE
200B; ZERO WIDTH SPACE
202F; NARROW NO-BREAK SPACE
205F; MEDIUM MATHEMATICAL SPACE
3000; IDEOGRAPHIC SPACE
```

## 5.2 Control characters

Control characters (or characters with control function) cannot be seen and can cause unpredictable results when displayed. Note that the list below is split into two tables in appendix C: Table C.2.1 contains the ASCII code points, while Table C.2.2 contains the non-ASCII code points. Most profiles of this document that want to prohibit control characters will want to include both tables.

```
0000-001F; [CONTROL CHARACTERS]
007F; DELETE
0080-009F; [CONTROL CHARACTERS]
06DD; ARABIC END OF AYAH
070F; SYRIAC ABBREVIATION MARK
180E; MONGOLIAN VOWEL SEPARATOR
```

200C; ZERO WIDTH NON-JOINER
200D; ZERO WIDTH JOINER
2028; LINE SEPARATOR
2029; PARAGRAPH SEPARATOR
2060; WORD JOINER
2061; FUNCTION APPLICATION
2062; INVISIBLE TIMES
2063; INVISIBLE SEPARATOR
206A-206F; [CONTROL CHARACTERS]
FEFF; ZERO WIDTH NO-BREAK SPACE
FFF9-FFFC; [CONTROL CHARACTERS]
1D173-1D17A; [MUSICAL CONTROL CHARACTERS]

## 5.3 Private use

Because private-use characters do not have defined meanings, they are likely to be prohibited. The private-use characters are:

E000-F8FF; [PRIVATE USE, PLANE 0]
F0000-FFFFD; [PRIVATE USE, PLANE 15]
100000-10FFFD; [PRIVATE USE, PLANE 16]

## 5.4 Non-character code points

Non-character code points are code points that have been allocated in ISO/IEC 10646 but are not characters. Because they are already assigned, they are guaranteed not to later change into characters.

FDD0-FDEF; [NONCHARACTER CODE POINTS]
FFFE-FFFF; [NONCHARACTER CODE POINTS]
1FFFE-1FFFF; [NONCHARACTER CODE POINTS]
2FFFE-2FFFF; [NONCHARACTER CODE POINTS]
3FFFE-3FFFF; [NONCHARACTER CODE POINTS]
4FFFE-4FFFF; [NONCHARACTER CODE POINTS]
5FFFE-5FFFF; [NONCHARACTER CODE POINTS]
6FFFE-6FFFF; [NONCHARACTER CODE POINTS]
7FFFE-7FFFF; [NONCHARACTER CODE POINTS]
8FFFE-8FFFF; [NONCHARACTER CODE POINTS]
9FFFE-9FFFF; [NONCHARACTER CODE POINTS]
AFFFE-AFFFF; [NONCHARACTER CODE POINTS]
BFFFE-BFFFF; [NONCHARACTER CODE POINTS]
CFFFE-CFFFF; [NONCHARACTER CODE POINTS]
DFFFE-DFFFF; [NONCHARACTER CODE POINTS]
EFFFE-EFFFF; [NONCHARACTER CODE POINTS]
FFFFE-FFFFF; [NONCHARACTER CODE POINTS]
10FFFE-10FFFF; [NONCHARACTER CODE POINTS]

The non-character code points are listed in the PropList.txt file from the Unicode database.

## 5.5 Surrogate codes

The following code points are permanently reserved for use as surrogate code values in the UTF-16 encoding, will never be assigned to characters in the Unicode repertoire, and are therefore prohibited:

D800-DFFF; [SURROGATE CODES]

## 5.6 Inappropriate for plain text

The following characters do not appear in regular text.

FFF9; INTERLINEAR ANNOTATION ANCHOR
FFFA; INTERLINEAR ANNOTATION SEPARATOR
FFFB; INTERLINEAR ANNOTATION TERMINATOR
FFFC; OBJECT REPLACEMENT CHARACTER

Although the replacement character (U+FFFD) might be used when a string is displayed, it doesn't make sense for it to be part of the string itself. It is often displayed by renderers to indicate "there would be some character here, but it cannot be rendered". For example, on a computer with no Asian fonts, a string with three ideographs might be rendered with three replacement characters.

FFFD; REPLACEMENT CHARACTER

## 5.7 Inappropriate for canonical representation

The ideographic description characters allow different sequences of characters to be rendered the same way, which makes them inappropriate for strings that have to have a single canonical representation.

2FF0-2FFB; [IDEOGRAPHIC DESCRIPTION CHARACTERS]

## 5.8 Change display properties or are deprecated

The following characters can cause changes in display or the order in which characters appear when rendered, or are deprecated in Unicode.

0340; COMBINING GRAVE TONE MARK
0341; COMBINING ACUTE TONE MARK
200E; LEFT-TO-RIGHT MARK
200F; RIGHT-TO-LEFT MARK

```
202A; LEFT-TO-RIGHT EMBEDDING
202B; RIGHT-TO-LEFT EMBEDDING
202C; POP DIRECTIONAL FORMATTING
202D; LEFT-TO-RIGHT OVERRIDE
202E; RIGHT-TO-LEFT OVERRIDE
206A; INHIBIT SYMMETRIC SWAPPING
206B; ACTIVATE SYMMETRIC SWAPPING
206C; INHIBIT ARABIC FORM SHAPING
206D; ACTIVATE ARABIC FORM SHAPING
206E; NATIONAL DIGIT SHAPES
206F; NOMINAL DIGIT SHAPES
```

### 5.9 Tagging characters

The following characters are used for tagging text and are invisible.

```
E0001; LANGUAGE TAG
E0020-E007F; [TAGGING CHARACTERS]
```

## 6. Bidirectional Characters

Most characters are displayed from left to right, but some are displayed from right to left. This feature of Unicode is called "bidirectional text", or "bidi" for short. The Unicode standard has an extensive discussion of how to reorder glyphs for display when dealing with bidirectional text such as Arabic or Hebrew. See [UAX9] for more information. In particular, all Unicode text is stored in logical order.

A profile MAY choose to ignore bidirectional text. However, ignoring bidirectional text can cause display ambiguities. For example, it is quite easy to create two different strings with the same characters (but in different order) that are correctly displayed identically. Therefore, in order to avoid most problems with ambiguous bidirectional text display, profile creators should strongly consider including the bidirectional character handling described in this section in their profile.

The stringprep process never emits both an error and a string. If an error is detected during the checking of bidirectional strings, only an error is returned.

[Unicode3.2] defines several bidirectional categories; each character has one bidirectional category assigned to it. For the purposes of the requirements below, an "RandALCat character" is a character that has Unicode bidirectional categories "R" or "AL"; an "LCat character" is a character that has Unicode bidirectional category "L". Note

that there are many characters which fall in neither of the above definitions; Latin digits (<U+0030> through <U+0039>) are examples of this because they have bidirectional category "EN".

In any profile that specifies bidirectional character handling, all three of the following requirements MUST be met:

1) The characters in section 5.8 MUST be prohibited.

2) If a string contains any RandALCat character, the string MUST NOT contain any LCat character.

3) If a string contains any RandALCat character, a RandALCat character MUST be the first character of the string, and a RandALCat character MUST be the last character of the string.

Note that requirement 3 prohibits strings such as <U+0627><U+0031> ("aleph 1") but allows strings such as <U+0627><U+0031><U+0628> ("aleph 1 beh"). [UAX9] goes into great detail about the display order of strings that contain particular categories of characters in particular sequences.

Table D.1 lists the characters that belong to Unicode bidirectional categories "R" and "AL". Table D.2 lists all the characters that belong to Unicode bidirectonal category "L". These tables are derived from [Unicode3.2].

## 7. Unassigned Code Points in Stringprep Profiles

This section describes two different types of strings in typical protocols where internationalized strings are used: "stored strings" and "queries". Of course, different Internet protocols use strings very differently, so these terms cannot be used exactly in every protocol that needs to use stringprep. In general, "stored strings" are strings that are used in protocol identifiers and named entities, such as names in digital certificates and DNS domain name parts. "Queries" are strings that are used to match against strings that are stored identifiers, such as user-entered names for digital certificate authorities and DNS lookups.

All code points not assigned in the character repertoire named in a stringprep profile are called "unassigned code points". Stored strings using the profile MUST NOT contain any unassigned code points. Queries for matching strings MAY contain unassigned code points. Note that this is the only part of this document where the requirements for queries differs from the requirements for stored strings.

Using two different policies for where unassigned code points can appear removes the need for versioning in protocols that use stringprep profiles. This is very useful since it makes the overall processing simpler and does not impose a "protocol" to handle versioning. It is expected that the ISO/IEC 10646 and Unicode repertoires will be updated fairly frequently; at the time that this document is being written, it has happened approximately once a year. Each time a new version of a repertoire appears, a new version of a profile MAY be created. Some end users will want to use the new code points as soon as they are defined.

The list of unassigned code points MUST be given in a profile, and that list MUST be used by implementations of the profile.

The goal of the requirements in this section is to prevent comparisons between two strings that were both permitted to contain unassigned code points. When two strings X and Y are compared and string Y was prepared in a way that permits unassigned code points, a negative result to the comparison is not definitive; it's possible that the strings don't match even though they would match if a more recent version of the profile were used for Y. However, if both X and Y were prepared in a way that permits unassigned code points, something worse can happen: even a positive result for the comparison is not definitive. It is possible that the strings do match even though they would not match if a more recent version of the profile were used (one that prohibits a code point appearing in both X and Y).

Due to the way that versioning is handled in this section, stored strings that are embedded in structures that cannot be changed (such as the signed parts of digital certificates) MUST NOT contain any unassigned code points.

## 7.1 Categories of code points

Each code point in a repertoire named by a profile of stringprep can be categorized by how it acts in the process described in earlier sections of this document:

| | |
|---|---|
| AO | Code points that can be in the output |
| MN | Code points that cannot be in the output because they never appear as output from mapping or normalization |
| D | Code points that cannot be in the output because they are disallowed in the prohibition step |
| U | Unassigned code points |

A subsequent version of a profile that references a newer version of a repertoire with new code points will inherently have some code points move from category U to either D, MN, or AO. For backwards compatibility, a subsequent version of a profile MUST NOT move code points from any other category. That is, current AO, MN, or D code points MUST NOT ever change to a different category.

Stored strings MUST NOT contain any code points outside of AO for the latest version of a profile. That is, they are forbidden to contain code points from the MN, D, or U categories.

Applications creating queries MUST treat U code points as if they were AO when preparing the query to be entered in the process described by a profile of stringprep. Those applications MAY optionally have a preprocessor that provide stricter checks: treating unassigned code points in the input as errors, or warning the user about the fact that the code point is unassigned in the version of a profile that the software is based on; such a choice is a local matter for the software.

## 7.2 Reasons for the difference between stored strings and queries

Different software using different versions of a stringprep profile need to interoperate with maximal compatibility. The scheme described in this section (stored strings MUST NOT contain unassigned code points, queries MAY include unassigned code points) allows that compatibility without introducing any known security or interoperability issues.

The list below shows what happens if a query contains a code point from category U that is allowed in a newer version of a profile. The query either matches the string that was intended, or matches no string at all. In this list, the query comes from an application using version "oldVersion" of a profile, the stored string was created using version "newVersion" of the same profile, and the code point X was in category U in oldVersion, and has changed category to AO, MN, or D. There are 3 possible scenarios:

1. X is assigned to AO -- In newVersion, X is in category AO. Because the application passed X through, it gets back a positive match with the stored string. There is one exceptional case, where X is a combining mark.

   The order of combining marks is normalized, so if another combining mark Y has a lower combining class than X then XY will be put in the canonical order YX. (Unassigned code points are never reordered, so this doesn't happen in oldVersion). If the query contains YX, the query will get positive match with the

stored string. However, no string can be stored with XY, so a query with XY will get a negative answer to the test for matching.

2. X is assigned to MN -- In newVersion, X is normalized to code point "nX" and therefore X is now put in category MN. This cannot exist in any stored string, so any query containing X will get a negative answer to the test for matching. Note, however, if the query had contained the letter nX, it would have positively matched.

3. X is assigned to D -- In newVersion, X is in category D. This cannot exist in any stored string, so any query containing X will get a negative answer to the test for matching.

In none of the cases does the query get data for a stored string other than the one it actually tried to match against.

Profiles are stable between versions in the following sense: If a string S has been prepared using newVersion, then it will not change if it is subsequently prepared using oldVersion.

### 7.3 Versions of applications and stored strings

Another way to see that this versioning system works is to compare what happens when an application uses a newer or older version of a profile.

Newer query application -- Suppose that a querying application is using version newVersion and the stored string was created using version oldVersion. This case is simple: there will be no characters in the stored string that cannot be queried by the application because the new profile uses a superset of the code points used for making the stored string.

Newer stored string -- Suppose that a querying application is using oldVersion and the stored string was created using a profile that uses newVersion. Because the querying application let unassigned code points pass through, the user can query on stored strings that use code points in newVersion. No stored strings can have code points that are unassigned in newVersion, since that is illegal. In order to get a match, the querying application has to enter the unassigned code points in the proper order, and has to use unassigned code points that would make it through both the mapping and the normalization steps.

## 8. References

### 8.1 Normative references

[UAX15] Mark Davis and Martin Duerst. Unicode Standard Annex #15: Unicode Normalization Forms, Version 3.2.0. <http://www.unicode.org/unicode/reports/tr15/tr15-22.html>.

[Unicode3.2] The Unicode Consortium. The Unicode Standard, Version 3.2.0 is defined by The Unicode Standard, Version 3.0 (Reading, MA, Addison-Wesley, 2000. ISBN 0-201-61633-5), as amended by the Unicode Standard Annex #27: Unicode 3.1 (http://www.unicode.org/reports/tr27/) and by the Unicode Standard Annex #28: Unicode 3.2 (http://www.unicode.org/reports/tr28/).

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

### 8.2 Informative references

[CharModel] Unicode Technical Report;17, Character Encoding Model. <http://www.unicode.org/unicode/reports/tr17/>.

[Glossary] Unicode Glossary, <http://www.unicode.org/glossary/>.

[ISO10646] ISO/IEC, "Information Technology - Universal Multiple-Octet Coded Character Set (UCS) - Part 1: Architecture and Basic Multilingual Plane", ISO/IEC 10646-1:2000, October 2000.

[RFC2434] Narten, T. and H. Alvestrand, "Guidelines for IANA Considerations", BCP 26, RFC 2434, October 1998.

[UAX9] The Unicode Consortium. Unicode Standard Annex #9, The Bidirectional Algorithm, <http://www.unicode.org/unicode/reports/tr9/>.

[UTR21] Mark Davis. Case Mappings. Unicode Technical Report 21. <http://www.unicode.org/unicode/reports/tr21/>.

## 9. Security Considerations

Stringprep is used with Unicode characters. There are security considerations that are specific to stringprep, and others that are generic to using Unicode.

### 9.1 Stringprep-specific security considerations

The Unicode and ISO/IEC 10646 repertoires have many characters that look similar. In many cases, users of security protocols might do visual matching, such as when comparing the names of trusted third parties. Because it is impossible to map similar-looking characters without a great deal of context such as knowing the fonts used, stringprep does nothing to map similar-looking characters together nor to prohibit some characters because they look like others. User applications can help disambiguate some similar-looking characters by showing the user when a string changes between scripts.

Most profiles of stringprep can cause changes in strings that are input to stringprep. Because of this, protocols that have sets of non-allowed characters or sequences MUST check for the non-allowed characters or sequences after the stringprep processing.

This document does not mandate the checking of bidirectional characters in section 6. If the requirements in section 6 are not used in a profile of stringprep, it is easy to create many strings whose characters are in different order but are displayed identically. This can cause security-related user confusion similar to look-alike characters, as described above.

Stringprep does not do anything to assure that any algorithms translating characters from non-Unicode into Unicode produce the same output in all implementations.

Some Unicode codepoints are invisible. Protocols that allow these characters (that is, do not map them out or prohibit them in stringprep) can cause users confusion when two identical-looking strings do not match.

### 9.2 Generic Unicode security considerations

Using Unicode characters explicitly forces applications to use multi-octet characters. Converting an application from one that uses single-octet characters to one that uses multi-octet characters must be done very carefully, particularly in an application that checks for values of characters or sorts characters.

Protocols that use stringprep usually also use encodings of Unicode, such as UTF-8 or UTF-16. Some applications using those encodings have been known to not check for illegal or ill-formed sequences in the encodings, and thereby have not detected sequences of octets that would have been detected if they used just ASCII. For example, in

UTF-8 the octet sequence "0xC0 0xAB" is an illegal formation of U+002B (plus sign). All programs should reject any string that is an illegal or ill-formed octet sequence for the encoding being used.

Both Unicode normalization and conversion between Unicode encodings can cause strings to grow or shrink. Programs that used fixed-size buffers, or that make assumptions that buffers will always be greater than or less than particular sizes, are likely to fail in insecure fashions when using Unicode normalization or encoding conversions.

Covering an extensive list of security threats and considerations on the use of current and future versions of Unicode is outside of the scope of this document.

## 10. IANA Considerations

Stringprep profiles MUST have IETF consensus as described in [RFC2434]. Each profile MUST be reviewed by the IESG before it is registered. The IESG MAY change a profile before registration.

IANA has set up a registry of stringprep profiles. This registry is a single text file that lists the known profiles. Each entry in the registry has three fields:

- Profile name

- RFC in which the profile is defined

- Indicator whether or not this is the newest version of the profile

Each version of a profile will remain listed in the registry forever. That is, if a new version of a profile supersedes an earlier version, both versions will continue to be listed in the registry, but the current version indicator will be turned off for the earlier version and turned on for the newer version.

It is probably harmful if a large number of profiles of stringprep proliferate. Therefore, the IESG may reject proposals for new profiles and instead suggest that protocols reuse existing profiles.

## 11. Acknowledgements

Many people from the IETF IDN Working Group and the Unicode Technical Committee contributed ideas that went into the first document of this document. Mark Davis and Patrik Faltstrom were particularly helpful in some of the ideas, such as the versioning description.

The IDN nameprep design team made many useful changes to the first document. That team and its advisors include:

Asmus Freytag
Cathy Wissink
Francois Yergeau
James Seng
Marc Blanchet
Mark Davis
Martin Duerst
Patrik Faltstrom
Paul Hoffman

Additional significant improvements were proposed by:

Jonathan Rosenne
Kent Karlsson
Scott Hollenbeck
Dave Crocker
Erik Nordmark
Matitiahu Allouche

## A. Unicode repertoires

The following is the only repertoire covered in this document:

Unicode 3.2, as defined in [Unicode3.2].

### A.1 Unassigned code points in Unicode 3.2

```
----- Start Table A.1 -----
0221
0234-024F
02AE-02AF
02EF-02FF
0350-035F
0370-0373
0376-0379
037B-037D
037F-0383
038B
038D
03A2
03CF
03F7-03FF
0487
04CF
04F6-04F7
04FA-04FF
0510-0530
0557-0558
0560
0588
058B-0590
05A2
05BA
05C5-05CF
05EB-05EF
05F5-060B
060D-061A
061C-061E
0620
063B-063F
0656-065F
06EE-06EF
06FF
070E
072D-072F
074B-077F
07B2-0900
```

```
0904
093A-093B
094E-094F
0955-0957
0971-0980
0984
098D-098E
0991-0992
09A9
09B1
09B3-09B5
09BA-09BB
09BD
09C5-09C6
09C9-09CA
09CE-09D6
09D8-09DB
09DE
09E4-09E5
09FB-0A01
0A03-0A04
0A0B-0A0E
0A11-0A12
0A29
0A31
0A34
0A37
0A3A-0A3B
0A3D
0A43-0A46
0A49-0A4A
0A4E-0A58
0A5D
0A5F-0A65
0A75-0A80
0A84
0A8C
0A8E
0A92
0AA9
0AB1
0AB4
0ABA-0ABB
0AC6
0ACA
0ACE-0ACF
0AD1-0ADF
0AE1-0AE5
```

0AF0-0B00
0B04
0B0D-0B0E
0B11-0B12
0B29
0B31
0B34-0B35
0B3A-0B3B
0B44-0B46
0B49-0B4A
0B4E-0B55
0B58-0B5B
0B5E
0B62-0B65
0B71-0B81
0B84
0B8B-0B8D
0B91
0B96-0B98
0B9B
0B9D
0BA0-0BA2
0BA5-0BA7
0BAB-0BAD
0BB6
0BBA-0BBD
0BC3-0BC5
0BC9
0BCE-0BD6
0BD8-0BE6
0BF3-0C00
0C04
0C0D
0C11
0C29
0C34
0C3A-0C3D
0C45
0C49
0C4E-0C54
0C57-0C5F
0C62-0C65
0C70-0C81
0C84
0C8D
0C91
0CA9
0CB4

```
0CBA-0CBD
0CC5
0CC9
0CCE-0CD4
0CD7-0CDD
0CDF
0CE2-0CE5
0CF0-0D01
0D04
0D0D
0D11
0D29
0D3A-0D3D
0D44-0D45
0D49
0D4E-0D56
0D58-0D5F
0D62-0D65
0D70-0D81
0D84
0D97-0D99
0DB2
0DBC
0DBE-0DBF
0DC7-0DC9
0DCB-0DCE
0DD5
0DD7
0DE0-0DF1
0DF5-0E00
0E3B-0E3E
0E5C-0E80
0E83
0E85-0E86
0E89
0E8B-0E8C
0E8E-0E93
0E98
0EA0
0EA4
0EA6
0EA8-0EA9
0EAC
0EBA
0EBE-0EBF
0EC5
0EC7
0ECE-0ECF
```

0EDA-0EDB
0EDE-0EFF
0F48
0F6B-0F70
0F8C-0F8F
0F98
0FBD
0FCD-0FCE
0FD0-0FFF
1022
1028
102B
1033-1035
103A-103F
105A-109F
10C6-10CF
10F9-10FA
10FC-10FF
115A-115E
11A3-11A7
11FA-11FF
1207
1247
1249
124E-124F
1257
1259
125E-125F
1287
1289
128E-128F
12AF
12B1
12B6-12B7
12BF
12C1
12C6-12C7
12CF
12D7
12EF
130F
1311
1316-1317
131F
1347
135B-1360
137D-139F
13F5-1400

1677-167F
169D-169F
16F1-16FF
170D
1715-171F
1737-173F
1754-175F
176D
1771
1774-177F
17DD-17DF
17EA-17FF
180F
181A-181F
1878-187F
18AA-1DFF
1E9C-1E9F
1EFA-1EFF
1F16-1F17
1F1E-1F1F
1F46-1F47
1F4E-1F4F
1F58
1F5A
1F5C
1F5E
1F7E-1F7F
1FB5
1FC5
1FD4-1FD5
1FDC
1FF0-1FF1
1FF5
1FFF
2053-2056
2058-205E
2064-2069
2072-2073
208F-209F
20B2-20CF
20EB-20FF
213B-213C
214C-2152
2184-218F
23CF-23FF
2427-243F
244B-245F
24FF

2614-2615
2618
267E-267F
268A-2700
2705
270A-270B
2728
274C
274E
2753-2755
2757
275F-2760
2795-2797
27B0
27BF-27CF
27EC-27EF
2B00-2E7F
2E9A
2EF4-2EFF
2FD6-2FEF
2FFC-2FFF
3040
3097-3098
3100-3104
312D-3130
318F
31B8-31EF
321D-321F
3244-3250
327C-327E
32CC-32CF
32FF
3377-337A
33DE-33DF
33FF
4DB6-4DFF
9FA6-9FFF
A48D-A48F
A4C7-ABFF
D7A4-D7FF
FA2E-FA2F
FA6B-FAFF
FB07-FB12
FB18-FB1C
FB37
FB3D
FB3F
FB42

FB45
FBB2-FBD2
FD40-FD4F
FD90-FD91
FDC8-FDCF
FDFD-FDFF
FE10-FE1F
FE24-FE2F
FE47-FE48
FE53
FE67
FE6C-FE6F
FE75
FEFD-FEFE
FF00
FFBF-FFC1
FFC8-FFC9
FFD0-FFD1
FFD8-FFD9
FFDD-FFDF
FFE7
FFEF-FFF8
10000-102FF
1031F
10324-1032F
1034B-103FF
10426-10427
1044E-1CFFF
1D0F6-1D0FF
1D127-1D129
1D1DE-1D3FF
1D455
1D49D
1D4A0-1D4A1
1D4A3-1D4A4
1D4A7-1D4A8
1D4AD
1D4BA
1D4BC
1D4C1
1D4C4
1D506
1D50B-1D50C
1D515
1D51D
1D53A
1D53F
1D545

```
1D547-1D549
1D551
1D6A4-1D6A7
1D7CA-1D7CD
1D800-1FFFD
2A6D7-2F7FF
2FA1E-2FFFD
30000-3FFFD
40000-4FFFD
50000-5FFFD
60000-6FFFD
70000-7FFFD
80000-8FFFD
90000-9FFFD
A0000-AFFFD
B0000-BFFFD
C0000-CFFFD
D0000-DFFFD
E0000
E0002-E001F
E0080-EFFFD
----- End Table A.1 -----
```

## B. Mapping Tables

The following is the mapping table from section 3. The table has three columns:

- the code point that is mapped from
- the zero or more code points that it is mapped to
- the reason for the mapping

The columns are separated by semicolons. Note that the second column may be empty, or it may have one code point, or it may have more than one code point, with each code point separated by a space.

### B.1 Commonly mapped to nothing

```
----- Start Table B.1 -----
00AD; ; Map to nothing
034F; ; Map to nothing
1806; ; Map to nothing
180B; ; Map to nothing
180C; ; Map to nothing
180D; ; Map to nothing
200B; ; Map to nothing
200C; ; Map to nothing
200D; ; Map to nothing
```

```
2060; ; Map to nothing
FE00; ; Map to nothing
FE01; ; Map to nothing
FE02; ; Map to nothing
FE03; ; Map to nothing
FE04; ; Map to nothing
FE05; ; Map to nothing
FE06; ; Map to nothing
FE07; ; Map to nothing
FE08; ; Map to nothing
FE09; ; Map to nothing
FE0A; ; Map to nothing
FE0B; ; Map to nothing
FE0C; ; Map to nothing
FE0D; ; Map to nothing
FE0E; ; Map to nothing
FE0F; ; Map to nothing
FEFF; ; Map to nothing
----- End Table B.1 -----
```

## B.2 Mapping for case-folding used with NFKC

```
----- Start Table B.2 -----
0041; 0061; Case map
0042; 0062; Case map
0043; 0063; Case map
0044; 0064; Case map
0045; 0065; Case map
0046; 0066; Case map
0047; 0067; Case map
0048; 0068; Case map
0049; 0069; Case map
004A; 006A; Case map
004B; 006B; Case map
004C; 006C; Case map
004D; 006D; Case map
004E; 006E; Case map
004F; 006F; Case map
0050; 0070; Case map
0051; 0071; Case map
0052; 0072; Case map
0053; 0073; Case map
0054; 0074; Case map
0055; 0075; Case map
0056; 0076; Case map
0057; 0077; Case map
0058; 0078; Case map
0059; 0079; Case map
```

```
005A; 007A; Case map
00B5; 03BC; Case map
00C0; 00E0; Case map
00C1; 00E1; Case map
00C2; 00E2; Case map
00C3; 00E3; Case map
00C4; 00E4; Case map
00C5; 00E5; Case map
00C6; 00E6; Case map
00C7; 00E7; Case map
00C8; 00E8; Case map
00C9; 00E9; Case map
00CA; 00EA; Case map
00CB; 00EB; Case map
00CC; 00EC; Case map
00CD; 00ED; Case map
00CE; 00EE; Case map
00CF; 00EF; Case map
00D0; 00F0; Case map
00D1; 00F1; Case map
00D2; 00F2; Case map
00D3; 00F3; Case map
00D4; 00F4; Case map
00D5; 00F5; Case map
00D6; 00F6; Case map
00D8; 00F8; Case map
00D9; 00F9; Case map
00DA; 00FA; Case map
00DB; 00FB; Case map
00DC; 00FC; Case map
00DD; 00FD; Case map
00DE; 00FE; Case map
00DF; 0073 0073; Case map
0100; 0101; Case map
0102; 0103; Case map
0104; 0105; Case map
0106; 0107; Case map
0108; 0109; Case map
010A; 010B; Case map
010C; 010D; Case map
010E; 010F; Case map
0110; 0111; Case map
0112; 0113; Case map
0114; 0115; Case map
0116; 0117; Case map
0118; 0119; Case map
011A; 011B; Case map
011C; 011D; Case map
```

```
011E; 011F; Case map
0120; 0121; Case map
0122; 0123; Case map
0124; 0125; Case map
0126; 0127; Case map
0128; 0129; Case map
012A; 012B; Case map
012C; 012D; Case map
012E; 012F; Case map
0130; 0069 0307; Case map
0132; 0133; Case map
0134; 0135; Case map
0136; 0137; Case map
0139; 013A; Case map
013B; 013C; Case map
013D; 013E; Case map
013F; 0140; Case map
0141; 0142; Case map
0143; 0144; Case map
0145; 0146; Case map
0147; 0148; Case map
0149; 02BC 006E; Case map
014A; 014B; Case map
014C; 014D; Case map
014E; 014F; Case map
0150; 0151; Case map
0152; 0153; Case map
0154; 0155; Case map
0156; 0157; Case map
0158; 0159; Case map
015A; 015B; Case map
015C; 015D; Case map
015E; 015F; Case map
0160; 0161; Case map
0162; 0163; Case map
0164; 0165; Case map
0166; 0167; Case map
0168; 0169; Case map
016A; 016B; Case map
016C; 016D; Case map
016E; 016F; Case map
0170; 0171; Case map
0172; 0173; Case map
0174; 0175; Case map
0176; 0177; Case map
0178; 00FF; Case map
0179; 017A; Case map
017B; 017C; Case map
```

```
017D; 017E; Case map
017F; 0073; Case map
0181; 0253; Case map
0182; 0183; Case map
0184; 0185; Case map
0186; 0254; Case map
0187; 0188; Case map
0189; 0256; Case map
018A; 0257; Case map
018B; 018C; Case map
018E; 01DD; Case map
018F; 0259; Case map
0190; 025B; Case map
0191; 0192; Case map
0193; 0260; Case map
0194; 0263; Case map
0196; 0269; Case map
0197; 0268; Case map
0198; 0199; Case map
019C; 026F; Case map
019D; 0272; Case map
019F; 0275; Case map
01A0; 01A1; Case map
01A2; 01A3; Case map
01A4; 01A5; Case map
01A6; 0280; Case map
01A7; 01A8; Case map
01A9; 0283; Case map
01AC; 01AD; Case map
01AE; 0288; Case map
01AF; 01B0; Case map
01B1; 028A; Case map
01B2; 028B; Case map
01B3; 01B4; Case map
01B5; 01B6; Case map
01B7; 0292; Case map
01B8; 01B9; Case map
01BC; 01BD; Case map
01C4; 01C6; Case map
01C5; 01C6; Case map
01C7; 01C9; Case map
01C8; 01C9; Case map
01CA; 01CC; Case map
01CB; 01CC; Case map
01CD; 01CE; Case map
01CF; 01D0; Case map
01D1; 01D2; Case map
01D3; 01D4; Case map
```

```
01D5; 01D6; Case map
01D7; 01D8; Case map
01D9; 01DA; Case map
01DB; 01DC; Case map
01DE; 01DF; Case map
01E0; 01E1; Case map
01E2; 01E3; Case map
01E4; 01E5; Case map
01E6; 01E7; Case map
01E8; 01E9; Case map
01EA; 01EB; Case map
01EC; 01ED; Case map
01EE; 01EF; Case map
01F0; 006A 030C; Case map
01F1; 01F3; Case map
01F2; 01F3; Case map
01F4; 01F5; Case map
01F6; 0195; Case map
01F7; 01BF; Case map
01F8; 01F9; Case map
01FA; 01FB; Case map
01FC; 01FD; Case map
01FE; 01FF; Case map
0200; 0201; Case map
0202; 0203; Case map
0204; 0205; Case map
0206; 0207; Case map
0208; 0209; Case map
020A; 020B; Case map
020C; 020D; Case map
020E; 020F; Case map
0210; 0211; Case map
0212; 0213; Case map
0214; 0215; Case map
0216; 0217; Case map
0218; 0219; Case map
021A; 021B; Case map
021C; 021D; Case map
021E; 021F; Case map
0220; 019E; Case map
0222; 0223; Case map
0224; 0225; Case map
0226; 0227; Case map
0228; 0229; Case map
022A; 022B; Case map
022C; 022D; Case map
022E; 022F; Case map
0230; 0231; Case map
```

```
0232; 0233; Case map
0345; 03B9; Case map
037A; 0020 03B9; Additional folding
0386; 03AC; Case map
0388; 03AD; Case map
0389; 03AE; Case map
038A; 03AF; Case map
038C; 03CC; Case map
038E; 03CD; Case map
038F; 03CE; Case map
0390; 03B9 0308 0301; Case map
0391; 03B1; Case map
0392; 03B2; Case map
0393; 03B3; Case map
0394; 03B4; Case map
0395; 03B5; Case map
0396; 03B6; Case map
0397; 03B7; Case map
0398; 03B8; Case map
0399; 03B9; Case map
039A; 03BA; Case map
039B; 03BB; Case map
039C; 03BC; Case map
039D; 03BD; Case map
039E; 03BE; Case map
039F; 03BF; Case map
03A0; 03C0; Case map
03A1; 03C1; Case map
03A3; 03C3; Case map
03A4; 03C4; Case map
03A5; 03C5; Case map
03A6; 03C6; Case map
03A7; 03C7; Case map
03A8; 03C8; Case map
03A9; 03C9; Case map
03AA; 03CA; Case map
03AB; 03CB; Case map
03B0; 03C5 0308 0301; Case map
03C2; 03C3; Case map
03D0; 03B2; Case map
03D1; 03B8; Case map
03D2; 03C5; Additional folding
03D3; 03CD; Additional folding
03D4; 03CB; Additional folding
03D5; 03C6; Case map
03D6; 03C0; Case map
03D8; 03D9; Case map
03DA; 03DB; Case map
```

```
03DC; 03DD; Case map
03DE; 03DF; Case map
03E0; 03E1; Case map
03E2; 03E3; Case map
03E4; 03E5; Case map
03E6; 03E7; Case map
03E8; 03E9; Case map
03EA; 03EB; Case map
03EC; 03ED; Case map
03EE; 03EF; Case map
03F0; 03BA; Case map
03F1; 03C1; Case map
03F2; 03C3; Case map
03F4; 03B8; Case map
03F5; 03B5; Case map
0400; 0450; Case map
0401; 0451; Case map
0402; 0452; Case map
0403; 0453; Case map
0404; 0454; Case map
0405; 0455; Case map
0406; 0456; Case map
0407; 0457; Case map
0408; 0458; Case map
0409; 0459; Case map
040A; 045A; Case map
040B; 045B; Case map
040C; 045C; Case map
040D; 045D; Case map
040E; 045E; Case map
040F; 045F; Case map
0410; 0430; Case map
0411; 0431; Case map
0412; 0432; Case map
0413; 0433; Case map
0414; 0434; Case map
0415; 0435; Case map
0416; 0436; Case map
0417; 0437; Case map
0418; 0438; Case map
0419; 0439; Case map
041A; 043A; Case map
041B; 043B; Case map
041C; 043C; Case map
041D; 043D; Case map
041E; 043E; Case map
041F; 043F; Case map
0420; 0440; Case map
```

```
0421; 0441; Case map
0422; 0442; Case map
0423; 0443; Case map
0424; 0444; Case map
0425; 0445; Case map
0426; 0446; Case map
0427; 0447; Case map
0428; 0448; Case map
0429; 0449; Case map
042A; 044A; Case map
042B; 044B; Case map
042C; 044C; Case map
042D; 044D; Case map
042E; 044E; Case map
042F; 044F; Case map
0460; 0461; Case map
0462; 0463; Case map
0464; 0465; Case map
0466; 0467; Case map
0468; 0469; Case map
046A; 046B; Case map
046C; 046D; Case map
046E; 046F; Case map
0470; 0471; Case map
0472; 0473; Case map
0474; 0475; Case map
0476; 0477; Case map
0478; 0479; Case map
047A; 047B; Case map
047C; 047D; Case map
047E; 047F; Case map
0480; 0481; Case map
048A; 048B; Case map
048C; 048D; Case map
048E; 048F; Case map
0490; 0491; Case map
0492; 0493; Case map
0494; 0495; Case map
0496; 0497; Case map
0498; 0499; Case map
049A; 049B; Case map
049C; 049D; Case map
049E; 049F; Case map
04A0; 04A1; Case map
04A2; 04A3; Case map
04A4; 04A5; Case map
04A6; 04A7; Case map
04A8; 04A9; Case map
```

```
04AA; 04AB; Case map
04AC; 04AD; Case map
04AE; 04AF; Case map
04B0; 04B1; Case map
04B2; 04B3; Case map
04B4; 04B5; Case map
04B6; 04B7; Case map
04B8; 04B9; Case map
04BA; 04BB; Case map
04BC; 04BD; Case map
04BE; 04BF; Case map
04C1; 04C2; Case map
04C3; 04C4; Case map
04C5; 04C6; Case map
04C7; 04C8; Case map
04C9; 04CA; Case map
04CB; 04CC; Case map
04CD; 04CE; Case map
04D0; 04D1; Case map
04D2; 04D3; Case map
04D4; 04D5; Case map
04D6; 04D7; Case map
04D8; 04D9; Case map
04DA; 04DB; Case map
04DC; 04DD; Case map
04DE; 04DF; Case map
04E0; 04E1; Case map
04E2; 04E3; Case map
04E4; 04E5; Case map
04E6; 04E7; Case map
04E8; 04E9; Case map
04EA; 04EB; Case map
04EC; 04ED; Case map
04EE; 04EF; Case map
04F0; 04F1; Case map
04F2; 04F3; Case map
04F4; 04F5; Case map
04F8; 04F9; Case map
0500; 0501; Case map
0502; 0503; Case map
0504; 0505; Case map
0506; 0507; Case map
0508; 0509; Case map
050A; 050B; Case map
050C; 050D; Case map
050E; 050F; Case map
0531; 0561; Case map
0532; 0562; Case map
```

```
0533; 0563; Case map
0534; 0564; Case map
0535; 0565; Case map
0536; 0566; Case map
0537; 0567; Case map
0538; 0568; Case map
0539; 0569; Case map
053A; 056A; Case map
053B; 056B; Case map
053C; 056C; Case map
053D; 056D; Case map
053E; 056E; Case map
053F; 056F; Case map
0540; 0570; Case map
0541; 0571; Case map
0542; 0572; Case map
0543; 0573; Case map
0544; 0574; Case map
0545; 0575; Case map
0546; 0576; Case map
0547; 0577; Case map
0548; 0578; Case map
0549; 0579; Case map
054A; 057A; Case map
054B; 057B; Case map
054C; 057C; Case map
054D; 057D; Case map
054E; 057E; Case map
054F; 057F; Case map
0550; 0580; Case map
0551; 0581; Case map
0552; 0582; Case map
0553; 0583; Case map
0554; 0584; Case map
0555; 0585; Case map
0556; 0586; Case map
0587; 0565 0582; Case map
1E00; 1E01; Case map
1E02; 1E03; Case map
1E04; 1E05; Case map
1E06; 1E07; Case map
1E08; 1E09; Case map
1E0A; 1E0B; Case map
1E0C; 1E0D; Case map
1E0E; 1E0F; Case map
1E10; 1E11; Case map
1E12; 1E13; Case map
1E14; 1E15; Case map
```

```
1E16; 1E17; Case map
1E18; 1E19; Case map
1E1A; 1E1B; Case map
1E1C; 1E1D; Case map
1E1E; 1E1F; Case map
1E20; 1E21; Case map
1E22; 1E23; Case map
1E24; 1E25; Case map
1E26; 1E27; Case map
1E28; 1E29; Case map
1E2A; 1E2B; Case map
1E2C; 1E2D; Case map
1E2E; 1E2F; Case map
1E30; 1E31; Case map
1E32; 1E33; Case map
1E34; 1E35; Case map
1E36; 1E37; Case map
1E38; 1E39; Case map
1E3A; 1E3B; Case map
1E3C; 1E3D; Case map
1E3E; 1E3F; Case map
1E40; 1E41; Case map
1E42; 1E43; Case map
1E44; 1E45; Case map
1E46; 1E47; Case map
1E48; 1E49; Case map
1E4A; 1E4B; Case map
1E4C; 1E4D; Case map
1E4E; 1E4F; Case map
1E50; 1E51; Case map
1E52; 1E53; Case map
1E54; 1E55; Case map
1E56; 1E57; Case map
1E58; 1E59; Case map
1E5A; 1E5B; Case map
1E5C; 1E5D; Case map
1E5E; 1E5F; Case map
1E60; 1E61; Case map
1E62; 1E63; Case map
1E64; 1E65; Case map
1E66; 1E67; Case map
1E68; 1E69; Case map
1E6A; 1E6B; Case map
1E6C; 1E6D; Case map
1E6E; 1E6F; Case map
1E70; 1E71; Case map
1E72; 1E73; Case map
1E74; 1E75; Case map
```

```
1E76; 1E77; Case map
1E78; 1E79; Case map
1E7A; 1E7B; Case map
1E7C; 1E7D; Case map
1E7E; 1E7F; Case map
1E80; 1E81; Case map
1E82; 1E83; Case map
1E84; 1E85; Case map
1E86; 1E87; Case map
1E88; 1E89; Case map
1E8A; 1E8B; Case map
1E8C; 1E8D; Case map
1E8E; 1E8F; Case map
1E90; 1E91; Case map
1E92; 1E93; Case map
1E94; 1E95; Case map
1E96; 0068 0331; Case map
1E97; 0074 0308; Case map
1E98; 0077 030A; Case map
1E99; 0079 030A; Case map
1E9A; 0061 02BE; Case map
1E9B; 1E61; Case map
1EA0; 1EA1; Case map
1EA2; 1EA3; Case map
1EA4; 1EA5; Case map
1EA6; 1EA7; Case map
1EA8; 1EA9; Case map
1EAA; 1EAB; Case map
1EAC; 1EAD; Case map
1EAE; 1EAF; Case map
1EB0; 1EB1; Case map
1EB2; 1EB3; Case map
1EB4; 1EB5; Case map
1EB6; 1EB7; Case map
1EB8; 1EB9; Case map
1EBA; 1EBB; Case map
1EBC; 1EBD; Case map
1EBE; 1EBF; Case map
1EC0; 1EC1; Case map
1EC2; 1EC3; Case map
1EC4; 1EC5; Case map
1EC6; 1EC7; Case map
1EC8; 1EC9; Case map
1ECA; 1ECB; Case map
1ECC; 1ECD; Case map
1ECE; 1ECF; Case map
1ED0; 1ED1; Case map
1ED2; 1ED3; Case map
```

```
1ED4; 1ED5; Case map
1ED6; 1ED7; Case map
1ED8; 1ED9; Case map
1EDA; 1EDB; Case map
1EDC; 1EDD; Case map
1EDE; 1EDF; Case map
1EE0; 1EE1; Case map
1EE2; 1EE3; Case map
1EE4; 1EE5; Case map
1EE6; 1EE7; Case map
1EE8; 1EE9; Case map
1EEA; 1EEB; Case map
1EEC; 1EED; Case map
1EEE; 1EEF; Case map
1EF0; 1EF1; Case map
1EF2; 1EF3; Case map
1EF4; 1EF5; Case map
1EF6; 1EF7; Case map
1EF8; 1EF9; Case map
1F08; 1F00; Case map
1F09; 1F01; Case map
1F0A; 1F02; Case map
1F0B; 1F03; Case map
1F0C; 1F04; Case map
1F0D; 1F05; Case map
1F0E; 1F06; Case map
1F0F; 1F07; Case map
1F18; 1F10; Case map
1F19; 1F11; Case map
1F1A; 1F12; Case map
1F1B; 1F13; Case map
1F1C; 1F14; Case map
1F1D; 1F15; Case map
1F28; 1F20; Case map
1F29; 1F21; Case map
1F2A; 1F22; Case map
1F2B; 1F23; Case map
1F2C; 1F24; Case map
1F2D; 1F25; Case map
1F2E; 1F26; Case map
1F2F; 1F27; Case map
1F38; 1F30; Case map
1F39; 1F31; Case map
1F3A; 1F32; Case map
1F3B; 1F33; Case map
1F3C; 1F34; Case map
1F3D; 1F35; Case map
1F3E; 1F36; Case map
```

```
1F3F; 1F37; Case map
1F48; 1F40; Case map
1F49; 1F41; Case map
1F4A; 1F42; Case map
1F4B; 1F43; Case map
1F4C; 1F44; Case map
1F4D; 1F45; Case map
1F50; 03C5 0313; Case map
1F52; 03C5 0313 0300; Case map
1F54; 03C5 0313 0301; Case map
1F56; 03C5 0313 0342; Case map
1F59; 1F51; Case map
1F5B; 1F53; Case map
1F5D; 1F55; Case map
1F5F; 1F57; Case map
1F68; 1F60; Case map
1F69; 1F61; Case map
1F6A; 1F62; Case map
1F6B; 1F63; Case map
1F6C; 1F64; Case map
1F6D; 1F65; Case map
1F6E; 1F66; Case map
1F6F; 1F67; Case map
1F80; 1F00 03B9; Case map
1F81; 1F01 03B9; Case map
1F82; 1F02 03B9; Case map
1F83; 1F03 03B9; Case map
1F84; 1F04 03B9; Case map
1F85; 1F05 03B9; Case map
1F86; 1F06 03B9; Case map
1F87; 1F07 03B9; Case map
1F88; 1F00 03B9; Case map
1F89; 1F01 03B9; Case map
1F8A; 1F02 03B9; Case map
1F8B; 1F03 03B9; Case map
1F8C; 1F04 03B9; Case map
1F8D; 1F05 03B9; Case map
1F8E; 1F06 03B9; Case map
1F8F; 1F07 03B9; Case map
1F90; 1F20 03B9; Case map
1F91; 1F21 03B9; Case map
1F92; 1F22 03B9; Case map
1F93; 1F23 03B9; Case map
1F94; 1F24 03B9; Case map
1F95; 1F25 03B9; Case map
1F96; 1F26 03B9; Case map
1F97; 1F27 03B9; Case map
1F98; 1F20 03B9; Case map
```

```
1F99; 1F21 03B9; Case map
1F9A; 1F22 03B9; Case map
1F9B; 1F23 03B9; Case map
1F9C; 1F24 03B9; Case map
1F9D; 1F25 03B9; Case map
1F9E; 1F26 03B9; Case map
1F9F; 1F27 03B9; Case map
1FA0; 1F60 03B9; Case map
1FA1; 1F61 03B9; Case map
1FA2; 1F62 03B9; Case map
1FA3; 1F63 03B9; Case map
1FA4; 1F64 03B9; Case map
1FA5; 1F65 03B9; Case map
1FA6; 1F66 03B9; Case map
1FA7; 1F67 03B9; Case map
1FA8; 1F60 03B9; Case map
1FA9; 1F61 03B9; Case map
1FAA; 1F62 03B9; Case map
1FAB; 1F63 03B9; Case map
1FAC; 1F64 03B9; Case map
1FAD; 1F65 03B9; Case map
1FAE; 1F66 03B9; Case map
1FAF; 1F67 03B9; Case map
1FB2; 1F70 03B9; Case map
1FB3; 03B1 03B9; Case map
1FB4; 03AC 03B9; Case map
1FB6; 03B1 0342; Case map
1FB7; 03B1 0342 03B9; Case map
1FB8; 1FB0; Case map
1FB9; 1FB1; Case map
1FBA; 1F70; Case map
1FBB; 1F71; Case map
1FBC; 03B1 03B9; Case map
1FBE; 03B9; Case map
1FC2; 1F74 03B9; Case map
1FC3; 03B7 03B9; Case map
1FC4; 03AE 03B9; Case map
1FC6; 03B7 0342; Case map
1FC7; 03B7 0342 03B9; Case map
1FC8; 1F72; Case map
1FC9; 1F73; Case map
1FCA; 1F74; Case map
1FCB; 1F75; Case map
1FCC; 03B7 03B9; Case map
1FD2; 03B9 0308 0300; Case map
1FD3; 03B9 0308 0301; Case map
1FD6; 03B9 0342; Case map
1FD7; 03B9 0308 0342; Case map
```

```
1FD8; 1FD0; Case map
1FD9; 1FD1; Case map
1FDA; 1F76; Case map
1FDB; 1F77; Case map
1FE2; 03C5 0308 0300; Case map
1FE3; 03C5 0308 0301; Case map
1FE4; 03C1 0313; Case map
1FE6; 03C5 0342; Case map
1FE7; 03C5 0308 0342; Case map
1FE8; 1FE0; Case map
1FE9; 1FE1; Case map
1FEA; 1F7A; Case map
1FEB; 1F7B; Case map
1FEC; 1FE5; Case map
1FF2; 1F7C 03B9; Case map
1FF3; 03C9 03B9; Case map
1FF4; 03CE 03B9; Case map
1FF6; 03C9 0342; Case map
1FF7; 03C9 0342 03B9; Case map
1FF8; 1F78; Case map
1FF9; 1F79; Case map
1FFA; 1F7C; Case map
1FFB; 1F7D; Case map
1FFC; 03C9 03B9; Case map
20A8; 0072 0073; Additional folding
2102; 0063; Additional folding
2103; 00B0 0063; Additional folding
2107; 025B; Additional folding
2109; 00B0 0066; Additional folding
210B; 0068; Additional folding
210C; 0068; Additional folding
210D; 0068; Additional folding
2110; 0069; Additional folding
2111; 0069; Additional folding
2112; 006C; Additional folding
2115; 006E; Additional folding
2116; 006E 006F; Additional folding
2119; 0070; Additional folding
211A; 0071; Additional folding
211B; 0072; Additional folding
211C; 0072; Additional folding
211D; 0072; Additional folding
2120; 0073 006D; Additional folding
2121; 0074 0065 006C; Additional folding
2122; 0074 006D; Additional folding
2124; 007A; Additional folding
2126; 03C9; Case map
2128; 007A; Additional folding
```

```
212A; 006B; Case map
212B; 00E5; Case map
212C; 0062; Additional folding
212D; 0063; Additional folding
2130; 0065; Additional folding
2131; 0066; Additional folding
2133; 006D; Additional folding
213E; 03B3; Additional folding
213F; 03C0; Additional folding
2145; 0064; Additional folding
2160; 2170; Case map
2161; 2171; Case map
2162; 2172; Case map
2163; 2173; Case map
2164; 2174; Case map
2165; 2175; Case map
2166; 2176; Case map
2167; 2177; Case map
2168; 2178; Case map
2169; 2179; Case map
216A; 217A; Case map
216B; 217B; Case map
216C; 217C; Case map
216D; 217D; Case map
216E; 217E; Case map
216F; 217F; Case map
24B6; 24D0; Case map
24B7; 24D1; Case map
24B8; 24D2; Case map
24B9; 24D3; Case map
24BA; 24D4; Case map
24BB; 24D5; Case map
24BC; 24D6; Case map
24BD; 24D7; Case map
24BE; 24D8; Case map
24BF; 24D9; Case map
24C0; 24DA; Case map
24C1; 24DB; Case map
24C2; 24DC; Case map
24C3; 24DD; Case map
24C4; 24DE; Case map
24C5; 24DF; Case map
24C6; 24E0; Case map
24C7; 24E1; Case map
24C8; 24E2; Case map
24C9; 24E3; Case map
24CA; 24E4; Case map
24CB; 24E5; Case map
```

```
24CC; 24E6; Case map
24CD; 24E7; Case map
24CE; 24E8; Case map
24CF; 24E9; Case map
3371; 0068 0070 0061; Additional folding
3373; 0061 0075; Additional folding
3375; 006F 0076; Additional folding
3380; 0070 0061; Additional folding
3381; 006E 0061; Additional folding
3382; 03BC 0061; Additional folding
3383; 006D 0061; Additional folding
3384; 006B 0061; Additional folding
3385; 006B 0062; Additional folding
3386; 006D 0062; Additional folding
3387; 0067 0062; Additional folding
338A; 0070 0066; Additional folding
338B; 006E 0066; Additional folding
338C; 03BC 0066; Additional folding
3390; 0068 007A; Additional folding
3391; 006B 0068 007A; Additional folding
3392; 006D 0068 007A; Additional folding
3393; 0067 0068 007A; Additional folding
3394; 0074 0068 007A; Additional folding
33A9; 0070 0061; Additional folding
33AA; 006B 0070 0061; Additional folding
33AB; 006D 0070 0061; Additional folding
33AC; 0067 0070 0061; Additional folding
33B4; 0070 0076; Additional folding
33B5; 006E 0076; Additional folding
33B6; 03BC 0076; Additional folding
33B7; 006D 0076; Additional folding
33B8; 006B 0076; Additional folding
33B9; 006D 0076; Additional folding
33BA; 0070 0077; Additional folding
33BB; 006E 0077; Additional folding
33BC; 03BC 0077; Additional folding
33BD; 006D 0077; Additional folding
33BE; 006B 0077; Additional folding
33BF; 006D 0077; Additional folding
33C0; 006B 03C9; Additional folding
33C1; 006D 03C9; Additional folding
33C3; 0062 0071; Additional folding
33C6; 0063 2215 006B 0067; Additional folding
33C7; 0063 006F 002E; Additional folding
33C8; 0064 0062; Additional folding
33C9; 0067 0079; Additional folding
33CB; 0068 0070; Additional folding
33CD; 006B 006B; Additional folding
```

```
33CE; 006B 006D; Additional folding
33D7; 0070 0068; Additional folding
33D9; 0070 0070 006D; Additional folding
33DA; 0070 0072; Additional folding
33DC; 0073 0076; Additional folding
33DD; 0077 0062; Additional folding
FB00; 0066 0066; Case map
FB01; 0066 0069; Case map
FB02; 0066 006C; Case map
FB03; 0066 0066 0069; Case map
FB04; 0066 0066 006C; Case map
FB05; 0073 0074; Case map
FB06; 0073 0074; Case map
FB13; 0574 0576; Case map
FB14; 0574 0565; Case map
FB15; 0574 056B; Case map
FB16; 057E 0576; Case map
FB17; 0574 056D; Case map
FF21; FF41; Case map
FF22; FF42; Case map
FF23; FF43; Case map
FF24; FF44; Case map
FF25; FF45; Case map
FF26; FF46; Case map
FF27; FF47; Case map
FF28; FF48; Case map
FF29; FF49; Case map
FF2A; FF4A; Case map
FF2B; FF4B; Case map
FF2C; FF4C; Case map
FF2D; FF4D; Case map
FF2E; FF4E; Case map
FF2F; FF4F; Case map
FF30; FF50; Case map
FF31; FF51; Case map
FF32; FF52; Case map
FF33; FF53; Case map
FF34; FF54; Case map
FF35; FF55; Case map
FF36; FF56; Case map
FF37; FF57; Case map
FF38; FF58; Case map
FF39; FF59; Case map
FF3A; FF5A; Case map
10400; 10428; Case map
10401; 10429; Case map
10402; 1042A; Case map
10403; 1042B; Case map
```

```
10404; 1042C; Case map
10405; 1042D; Case map
10406; 1042E; Case map
10407; 1042F; Case map
10408; 10430; Case map
10409; 10431; Case map
1040A; 10432; Case map
1040B; 10433; Case map
1040C; 10434; Case map
1040D; 10435; Case map
1040E; 10436; Case map
1040F; 10437; Case map
10410; 10438; Case map
10411; 10439; Case map
10412; 1043A; Case map
10413; 1043B; Case map
10414; 1043C; Case map
10415; 1043D; Case map
10416; 1043E; Case map
10417; 1043F; Case map
10418; 10440; Case map
10419; 10441; Case map
1041A; 10442; Case map
1041B; 10443; Case map
1041C; 10444; Case map
1041D; 10445; Case map
1041E; 10446; Case map
1041F; 10447; Case map
10420; 10448; Case map
10421; 10449; Case map
10422; 1044A; Case map
10423; 1044B; Case map
10424; 1044C; Case map
10425; 1044D; Case map
1D400; 0061; Additional folding
1D401; 0062; Additional folding
1D402; 0063; Additional folding
1D403; 0064; Additional folding
1D404; 0065; Additional folding
1D405; 0066; Additional folding
1D406; 0067; Additional folding
1D407; 0068; Additional folding
1D408; 0069; Additional folding
1D409; 006A; Additional folding
1D40A; 006B; Additional folding
1D40B; 006C; Additional folding
1D40C; 006D; Additional folding
1D40D; 006E; Additional folding
```

```
1D40E; 006F; Additional folding
1D40F; 0070; Additional folding
1D410; 0071; Additional folding
1D411; 0072; Additional folding
1D412; 0073; Additional folding
1D413; 0074; Additional folding
1D414; 0075; Additional folding
1D415; 0076; Additional folding
1D416; 0077; Additional folding
1D417; 0078; Additional folding
1D418; 0079; Additional folding
1D419; 007A; Additional folding
1D434; 0061; Additional folding
1D435; 0062; Additional folding
1D436; 0063; Additional folding
1D437; 0064; Additional folding
1D438; 0065; Additional folding
1D439; 0066; Additional folding
1D43A; 0067; Additional folding
1D43B; 0068; Additional folding
1D43C; 0069; Additional folding
1D43D; 006A; Additional folding
1D43E; 006B; Additional folding
1D43F; 006C; Additional folding
1D440; 006D; Additional folding
1D441; 006E; Additional folding
1D442; 006F; Additional folding
1D443; 0070; Additional folding
1D444; 0071; Additional folding
1D445; 0072; Additional folding
1D446; 0073; Additional folding
1D447; 0074; Additional folding
1D448; 0075; Additional folding
1D449; 0076; Additional folding
1D44A; 0077; Additional folding
1D44B; 0078; Additional folding
1D44C; 0079; Additional folding
1D44D; 007A; Additional folding
1D468; 0061; Additional folding
1D469; 0062; Additional folding
1D46A; 0063; Additional folding
1D46B; 0064; Additional folding
1D46C; 0065; Additional folding
1D46D; 0066; Additional folding
1D46E; 0067; Additional folding
1D46F; 0068; Additional folding
1D470; 0069; Additional folding
1D471; 006A; Additional folding
```

```
1D472; 006B; Additional folding
1D473; 006C; Additional folding
1D474; 006D; Additional folding
1D475; 006E; Additional folding
1D476; 006F; Additional folding
1D477; 0070; Additional folding
1D478; 0071; Additional folding
1D479; 0072; Additional folding
1D47A; 0073; Additional folding
1D47B; 0074; Additional folding
1D47C; 0075; Additional folding
1D47D; 0076; Additional folding
1D47E; 0077; Additional folding
1D47F; 0078; Additional folding
1D480; 0079; Additional folding
1D481; 007A; Additional folding
1D49C; 0061; Additional folding
1D49E; 0063; Additional folding
1D49F; 0064; Additional folding
1D4A2; 0067; Additional folding
1D4A5; 006A; Additional folding
1D4A6; 006B; Additional folding
1D4A9; 006E; Additional folding
1D4AA; 006F; Additional folding
1D4AB; 0070; Additional folding
1D4AC; 0071; Additional folding
1D4AE; 0073; Additional folding
1D4AF; 0074; Additional folding
1D4B0; 0075; Additional folding
1D4B1; 0076; Additional folding
1D4B2; 0077; Additional folding
1D4B3; 0078; Additional folding
1D4B4; 0079; Additional folding
1D4B5; 007A; Additional folding
1D4D0; 0061; Additional folding
1D4D1; 0062; Additional folding
1D4D2; 0063; Additional folding
1D4D3; 0064; Additional folding
1D4D4; 0065; Additional folding
1D4D5; 0066; Additional folding
1D4D6; 0067; Additional folding
1D4D7; 0068; Additional folding
1D4D8; 0069; Additional folding
1D4D9; 006A; Additional folding
1D4DA; 006B; Additional folding
1D4DB; 006C; Additional folding
1D4DC; 006D; Additional folding
1D4DD; 006E; Additional folding
```

```
1D4DE; 006F; Additional folding
1D4DF; 0070; Additional folding
1D4E0; 0071; Additional folding
1D4E1; 0072; Additional folding
1D4E2; 0073; Additional folding
1D4E3; 0074; Additional folding
1D4E4; 0075; Additional folding
1D4E5; 0076; Additional folding
1D4E6; 0077; Additional folding
1D4E7; 0078; Additional folding
1D4E8; 0079; Additional folding
1D4E9; 007A; Additional folding
1D504; 0061; Additional folding
1D505; 0062; Additional folding
1D507; 0064; Additional folding
1D508; 0065; Additional folding
1D509; 0066; Additional folding
1D50A; 0067; Additional folding
1D50D; 006A; Additional folding
1D50E; 006B; Additional folding
1D50F; 006C; Additional folding
1D510; 006D; Additional folding
1D511; 006E; Additional folding
1D512; 006F; Additional folding
1D513; 0070; Additional folding
1D514; 0071; Additional folding
1D516; 0073; Additional folding
1D517; 0074; Additional folding
1D518; 0075; Additional folding
1D519; 0076; Additional folding
1D51A; 0077; Additional folding
1D51B; 0078; Additional folding
1D51C; 0079; Additional folding
1D538; 0061; Additional folding
1D539; 0062; Additional folding
1D53B; 0064; Additional folding
1D53C; 0065; Additional folding
1D53D; 0066; Additional folding
1D53E; 0067; Additional folding
1D540; 0069; Additional folding
1D541; 006A; Additional folding
1D542; 006B; Additional folding
1D543; 006C; Additional folding
1D544; 006D; Additional folding
1D546; 006F; Additional folding
1D54A; 0073; Additional folding
1D54B; 0074; Additional folding
1D54C; 0075; Additional folding
```

```
1D54D; 0076; Additional folding
1D54E; 0077; Additional folding
1D54F; 0078; Additional folding
1D550; 0079; Additional folding
1D56C; 0061; Additional folding
1D56D; 0062; Additional folding
1D56E; 0063; Additional folding
1D56F; 0064; Additional folding
1D570; 0065; Additional folding
1D571; 0066; Additional folding
1D572; 0067; Additional folding
1D573; 0068; Additional folding
1D574; 0069; Additional folding
1D575; 006A; Additional folding
1D576; 006B; Additional folding
1D577; 006C; Additional folding
1D578; 006D; Additional folding
1D579; 006E; Additional folding
1D57A; 006F; Additional folding
1D57B; 0070; Additional folding
1D57C; 0071; Additional folding
1D57D; 0072; Additional folding
1D57E; 0073; Additional folding
1D57F; 0074; Additional folding
1D580; 0075; Additional folding
1D581; 0076; Additional folding
1D582; 0077; Additional folding
1D583; 0078; Additional folding
1D584; 0079; Additional folding
1D585; 007A; Additional folding
1D5A0; 0061; Additional folding
1D5A1; 0062; Additional folding
1D5A2; 0063; Additional folding
1D5A3; 0064; Additional folding
1D5A4; 0065; Additional folding
1D5A5; 0066; Additional folding
1D5A6; 0067; Additional folding
1D5A7; 0068; Additional folding
1D5A8; 0069; Additional folding
1D5A9; 006A; Additional folding
1D5AA; 006B; Additional folding
1D5AB; 006C; Additional folding
1D5AC; 006D; Additional folding
1D5AD; 006E; Additional folding
1D5AE; 006F; Additional folding
1D5AF; 0070; Additional folding
1D5B0; 0071; Additional folding
1D5B1; 0072; Additional folding
```

```
1D5B2; 0073; Additional folding
1D5B3; 0074; Additional folding
1D5B4; 0075; Additional folding
1D5B5; 0076; Additional folding
1D5B6; 0077; Additional folding
1D5B7; 0078; Additional folding
1D5B8; 0079; Additional folding
1D5B9; 007A; Additional folding
1D5D4; 0061; Additional folding
1D5D5; 0062; Additional folding
1D5D6; 0063; Additional folding
1D5D7; 0064; Additional folding
1D5D8; 0065; Additional folding
1D5D9; 0066; Additional folding
1D5DA; 0067; Additional folding
1D5DB; 0068; Additional folding
1D5DC; 0069; Additional folding
1D5DD; 006A; Additional folding
1D5DE; 006B; Additional folding
1D5DF; 006C; Additional folding
1D5E0; 006D; Additional folding
1D5E1; 006E; Additional folding
1D5E2; 006F; Additional folding
1D5E3; 0070; Additional folding
1D5E4; 0071; Additional folding
1D5E5; 0072; Additional folding
1D5E6; 0073; Additional folding
1D5E7; 0074; Additional folding
1D5E8; 0075; Additional folding
1D5E9; 0076; Additional folding
1D5EA; 0077; Additional folding
1D5EB; 0078; Additional folding
1D5EC; 0079; Additional folding
1D5ED; 007A; Additional folding
1D608; 0061; Additional folding
1D609; 0062; Additional folding
1D60A; 0063; Additional folding
1D60B; 0064; Additional folding
1D60C; 0065; Additional folding
1D60D; 0066; Additional folding
1D60E; 0067; Additional folding
1D60F; 0068; Additional folding
1D610; 0069; Additional folding
1D611; 006A; Additional folding
1D612; 006B; Additional folding
1D613; 006C; Additional folding
1D614; 006D; Additional folding
1D615; 006E; Additional folding
```

```
1D616; 006F; Additional folding
1D617; 0070; Additional folding
1D618; 0071; Additional folding
1D619; 0072; Additional folding
1D61A; 0073; Additional folding
1D61B; 0074; Additional folding
1D61C; 0075; Additional folding
1D61D; 0076; Additional folding
1D61E; 0077; Additional folding
1D61F; 0078; Additional folding
1D620; 0079; Additional folding
1D621; 007A; Additional folding
1D63C; 0061; Additional folding
1D63D; 0062; Additional folding
1D63E; 0063; Additional folding
1D63F; 0064; Additional folding
1D640; 0065; Additional folding
1D641; 0066; Additional folding
1D642; 0067; Additional folding
1D643; 0068; Additional folding
1D644; 0069; Additional folding
1D645; 006A; Additional folding
1D646; 006B; Additional folding
1D647; 006C; Additional folding
1D648; 006D; Additional folding
1D649; 006E; Additional folding
1D64A; 006F; Additional folding
1D64B; 0070; Additional folding
1D64C; 0071; Additional folding
1D64D; 0072; Additional folding
1D64E; 0073; Additional folding
1D64F; 0074; Additional folding
1D650; 0075; Additional folding
1D651; 0076; Additional folding
1D652; 0077; Additional folding
1D653; 0078; Additional folding
1D654; 0079; Additional folding
1D655; 007A; Additional folding
1D670; 0061; Additional folding
1D671; 0062; Additional folding
1D672; 0063; Additional folding
1D673; 0064; Additional folding
1D674; 0065; Additional folding
1D675; 0066; Additional folding
1D676; 0067; Additional folding
1D677; 0068; Additional folding
1D678; 0069; Additional folding
1D679; 006A; Additional folding
```

```
1D67A; 006B; Additional folding
1D67B; 006C; Additional folding
1D67C; 006D; Additional folding
1D67D; 006E; Additional folding
1D67E; 006F; Additional folding
1D67F; 0070; Additional folding
1D680; 0071; Additional folding
1D681; 0072; Additional folding
1D682; 0073; Additional folding
1D683; 0074; Additional folding
1D684; 0075; Additional folding
1D685; 0076; Additional folding
1D686; 0077; Additional folding
1D687; 0078; Additional folding
1D688; 0079; Additional folding
1D689; 007A; Additional folding
1D6A8; 03B1; Additional folding
1D6A9; 03B2; Additional folding
1D6AA; 03B3; Additional folding
1D6AB; 03B4; Additional folding
1D6AC; 03B5; Additional folding
1D6AD; 03B6; Additional folding
1D6AE; 03B7; Additional folding
1D6AF; 03B8; Additional folding
1D6B0; 03B9; Additional folding
1D6B1; 03BA; Additional folding
1D6B2; 03BB; Additional folding
1D6B3; 03BC; Additional folding
1D6B4; 03BD; Additional folding
1D6B5; 03BE; Additional folding
1D6B6; 03BF; Additional folding
1D6B7; 03C0; Additional folding
1D6B8; 03C1; Additional folding
1D6B9; 03B8; Additional folding
1D6BA; 03C3; Additional folding
1D6BB; 03C4; Additional folding
1D6BC; 03C5; Additional folding
1D6BD; 03C6; Additional folding
1D6BE; 03C7; Additional folding
1D6BF; 03C8; Additional folding
1D6C0; 03C9; Additional folding
1D6D3; 03C3; Additional folding
1D6E2; 03B1; Additional folding
1D6E3; 03B2; Additional folding
1D6E4; 03B3; Additional folding
1D6E5; 03B4; Additional folding
1D6E6; 03B5; Additional folding
1D6E7; 03B6; Additional folding
```

```
1D6E8; 03B7; Additional folding
1D6E9; 03B8; Additional folding
1D6EA; 03B9; Additional folding
1D6EB; 03BA; Additional folding
1D6EC; 03BB; Additional folding
1D6ED; 03BC; Additional folding
1D6EE; 03BD; Additional folding
1D6EF; 03BE; Additional folding
1D6F0; 03BF; Additional folding
1D6F1; 03C0; Additional folding
1D6F2; 03C1; Additional folding
1D6F3; 03B8; Additional folding
1D6F4; 03C3; Additional folding
1D6F5; 03C4; Additional folding
1D6F6; 03C5; Additional folding
1D6F7; 03C6; Additional folding
1D6F8; 03C7; Additional folding
1D6F9; 03C8; Additional folding
1D6FA; 03C9; Additional folding
1D70D; 03C3; Additional folding
1D71C; 03B1; Additional folding
1D71D; 03B2; Additional folding
1D71E; 03B3; Additional folding
1D71F; 03B4; Additional folding
1D720; 03B5; Additional folding
1D721; 03B6; Additional folding
1D722; 03B7; Additional folding
1D723; 03B8; Additional folding
1D724; 03B9; Additional folding
1D725; 03BA; Additional folding
1D726; 03BB; Additional folding
1D727; 03BC; Additional folding
1D728; 03BD; Additional folding
1D729; 03BE; Additional folding
1D72A; 03BF; Additional folding
1D72B; 03C0; Additional folding
1D72C; 03C1; Additional folding
1D72D; 03B8; Additional folding
1D72E; 03C3; Additional folding
1D72F; 03C4; Additional folding
1D730; 03C5; Additional folding
1D731; 03C6; Additional folding
1D732; 03C7; Additional folding
1D733; 03C8; Additional folding
1D734; 03C9; Additional folding
1D747; 03C3; Additional folding
1D756; 03B1; Additional folding
1D757; 03B2; Additional folding
```

```
1D758; 03B3; Additional folding
1D759; 03B4; Additional folding
1D75A; 03B5; Additional folding
1D75B; 03B6; Additional folding
1D75C; 03B7; Additional folding
1D75D; 03B8; Additional folding
1D75E; 03B9; Additional folding
1D75F; 03BA; Additional folding
1D760; 03BB; Additional folding
1D761; 03BC; Additional folding
1D762; 03BD; Additional folding
1D763; 03BE; Additional folding
1D764; 03BF; Additional folding
1D765; 03C0; Additional folding
1D766; 03C1; Additional folding
1D767; 03B8; Additional folding
1D768; 03C3; Additional folding
1D769; 03C4; Additional folding
1D76A; 03C5; Additional folding
1D76B; 03C6; Additional folding
1D76C; 03C7; Additional folding
1D76D; 03C8; Additional folding
1D76E; 03C9; Additional folding
1D781; 03C3; Additional folding
1D790; 03B1; Additional folding
1D791; 03B2; Additional folding
1D792; 03B3; Additional folding
1D793; 03B4; Additional folding
1D794; 03B5; Additional folding
1D795; 03B6; Additional folding
1D796; 03B7; Additional folding
1D797; 03B8; Additional folding
1D798; 03B9; Additional folding
1D799; 03BA; Additional folding
1D79A; 03BB; Additional folding
1D79B; 03BC; Additional folding
1D79C; 03BD; Additional folding
1D79D; 03BE; Additional folding
1D79E; 03BF; Additional folding
1D79F; 03C0; Additional folding
1D7A0; 03C1; Additional folding
1D7A1; 03B8; Additional folding
1D7A2; 03C3; Additional folding
1D7A3; 03C4; Additional folding
1D7A4; 03C5; Additional folding
1D7A5; 03C6; Additional folding
1D7A6; 03C7; Additional folding
1D7A7; 03C8; Additional folding
```

```
   1D7A8; 03C9; Additional folding
   1D7BB; 03C3; Additional folding
   ----- End Table B.2 -----
```

## B.3 Mapping for case-folding used with no normalization

```
   ----- Start Table B.3 -----
   0041; 0061; Case map
   0042; 0062; Case map
   0043; 0063; Case map
   0044; 0064; Case map
   0045; 0065; Case map
   0046; 0066; Case map
   0047; 0067; Case map
   0048; 0068; Case map
   0049; 0069; Case map
   004A; 006A; Case map
   004B; 006B; Case map
   004C; 006C; Case map
   004D; 006D; Case map
   004E; 006E; Case map
   004F; 006F; Case map
   0050; 0070; Case map
   0051; 0071; Case map
   0052; 0072; Case map
   0053; 0073; Case map
   0054; 0074; Case map
   0055; 0075; Case map
   0056; 0076; Case map
   0057; 0077; Case map
   0058; 0078; Case map
   0059; 0079; Case map
   005A; 007A; Case map
   00B5; 03BC; Case map
   00C0; 00E0; Case map
   00C1; 00E1; Case map
   00C2; 00E2; Case map
   00C3; 00E3; Case map
   00C4; 00E4; Case map
   00C5; 00E5; Case map
   00C6; 00E6; Case map
   00C7; 00E7; Case map
   00C8; 00E8; Case map
   00C9; 00E9; Case map
   00CA; 00EA; Case map
   00CB; 00EB; Case map
   00CC; 00EC; Case map
   00CD; 00ED; Case map
```

```
00CE; 00EE; Case map
00CF; 00EF; Case map
00D0; 00F0; Case map
00D1; 00F1; Case map
00D2; 00F2; Case map
00D3; 00F3; Case map
00D4; 00F4; Case map
00D5; 00F5; Case map
00D6; 00F6; Case map
00D8; 00F8; Case map
00D9; 00F9; Case map
00DA; 00FA; Case map
00DB; 00FB; Case map
00DC; 00FC; Case map
00DD; 00FD; Case map
00DE; 00FE; Case map
00DF; 0073 0073; Case map
0100; 0101; Case map
0102; 0103; Case map
0104; 0105; Case map
0106; 0107; Case map
0108; 0109; Case map
010A; 010B; Case map
010C; 010D; Case map
010E; 010F; Case map
0110; 0111; Case map
0112; 0113; Case map
0114; 0115; Case map
0116; 0117; Case map
0118; 0119; Case map
011A; 011B; Case map
011C; 011D; Case map
011E; 011F; Case map
0120; 0121; Case map
0122; 0123; Case map
0124; 0125; Case map
0126; 0127; Case map
0128; 0129; Case map
012A; 012B; Case map
012C; 012D; Case map
012E; 012F; Case map
0130; 0069 0307; Case map
0132; 0133; Case map
0134; 0135; Case map
0136; 0137; Case map
0139; 013A; Case map
013B; 013C; Case map
013D; 013E; Case map
```

```
013F; 0140; Case map
0141; 0142; Case map
0143; 0144; Case map
0145; 0146; Case map
0147; 0148; Case map
0149; 02BC 006E; Case map
014A; 014B; Case map
014C; 014D; Case map
014E; 014F; Case map
0150; 0151; Case map
0152; 0153; Case map
0154; 0155; Case map
0156; 0157; Case map
0158; 0159; Case map
015A; 015B; Case map
015C; 015D; Case map
015E; 015F; Case map
0160; 0161; Case map
0162; 0163; Case map
0164; 0165; Case map
0166; 0167; Case map
0168; 0169; Case map
016A; 016B; Case map
016C; 016D; Case map
016E; 016F; Case map
0170; 0171; Case map
0172; 0173; Case map
0174; 0175; Case map
0176; 0177; Case map
0178; 00FF; Case map
0179; 017A; Case map
017B; 017C; Case map
017D; 017E; Case map
017F; 0073; Case map
0181; 0253; Case map
0182; 0183; Case map
0184; 0185; Case map
0186; 0254; Case map
0187; 0188; Case map
0189; 0256; Case map
018A; 0257; Case map
018B; 018C; Case map
018E; 01DD; Case map
018F; 0259; Case map
0190; 025B; Case map
0191; 0192; Case map
0193; 0260; Case map
0194; 0263; Case map
```

```
0196; 0269; Case map
0197; 0268; Case map
0198; 0199; Case map
019C; 026F; Case map
019D; 0272; Case map
019F; 0275; Case map
01A0; 01A1; Case map
01A2; 01A3; Case map
01A4; 01A5; Case map
01A6; 0280; Case map
01A7; 01A8; Case map
01A9; 0283; Case map
01AC; 01AD; Case map
01AE; 0288; Case map
01AF; 01B0; Case map
01B1; 028A; Case map
01B2; 028B; Case map
01B3; 01B4; Case map
01B5; 01B6; Case map
01B7; 0292; Case map
01B8; 01B9; Case map
01BC; 01BD; Case map
01C4; 01C6; Case map
01C5; 01C6; Case map
01C7; 01C9; Case map
01C8; 01C9; Case map
01CA; 01CC; Case map
01CB; 01CC; Case map
01CD; 01CE; Case map
01CF; 01D0; Case map
01D1; 01D2; Case map
01D3; 01D4; Case map
01D5; 01D6; Case map
01D7; 01D8; Case map
01D9; 01DA; Case map
01DB; 01DC; Case map
01DE; 01DF; Case map
01E0; 01E1; Case map
01E2; 01E3; Case map
01E4; 01E5; Case map
01E6; 01E7; Case map
01E8; 01E9; Case map
01EA; 01EB; Case map
01EC; 01ED; Case map
01EE; 01EF; Case map
01F0; 006A 030C; Case map
01F1; 01F3; Case map
01F2; 01F3; Case map
```

```
01F4; 01F5; Case map
01F6; 0195; Case map
01F7; 01BF; Case map
01F8; 01F9; Case map
01FA; 01FB; Case map
01FC; 01FD; Case map
01FE; 01FF; Case map
0200; 0201; Case map
0202; 0203; Case map
0204; 0205; Case map
0206; 0207; Case map
0208; 0209; Case map
020A; 020B; Case map
020C; 020D; Case map
020E; 020F; Case map
0210; 0211; Case map
0212; 0213; Case map
0214; 0215; Case map
0216; 0217; Case map
0218; 0219; Case map
021A; 021B; Case map
021C; 021D; Case map
021E; 021F; Case map
0220; 019E; Case map
0222; 0223; Case map
0224; 0225; Case map
0226; 0227; Case map
0228; 0229; Case map
022A; 022B; Case map
022C; 022D; Case map
022E; 022F; Case map
0230; 0231; Case map
0232; 0233; Case map
0345; 03B9; Case map
0386; 03AC; Case map
0388; 03AD; Case map
0389; 03AE; Case map
038A; 03AF; Case map
038C; 03CC; Case map
038E; 03CD; Case map
038F; 03CE; Case map
0390; 03B9 0308 0301; Case map
0391; 03B1; Case map
0392; 03B2; Case map
0393; 03B3; Case map
0394; 03B4; Case map
0395; 03B5; Case map
0396; 03B6; Case map
```

```
0397; 03B7; Case map
0398; 03B8; Case map
0399; 03B9; Case map
039A; 03BA; Case map
039B; 03BB; Case map
039C; 03BC; Case map
039D; 03BD; Case map
039E; 03BE; Case map
039F; 03BF; Case map
03A0; 03C0; Case map
03A1; 03C1; Case map
03A3; 03C3; Case map
03A4; 03C4; Case map
03A5; 03C5; Case map
03A6; 03C6; Case map
03A7; 03C7; Case map
03A8; 03C8; Case map
03A9; 03C9; Case map
03AA; 03CA; Case map
03AB; 03CB; Case map
03B0; 03C5 0308 0301; Case map
03C2; 03C3; Case map
03D0; 03B2; Case map
03D1; 03B8; Case map
03D5; 03C6; Case map
03D6; 03C0; Case map
03D8; 03D9; Case map
03DA; 03DB; Case map
03DC; 03DD; Case map
03DE; 03DF; Case map
03E0; 03E1; Case map
03E2; 03E3; Case map
03E4; 03E5; Case map
03E6; 03E7; Case map
03E8; 03E9; Case map
03EA; 03EB; Case map
03EC; 03ED; Case map
03EE; 03EF; Case map
03F0; 03BA; Case map
03F1; 03C1; Case map
03F2; 03C3; Case map
03F4; 03B8; Case map
03F5; 03B5; Case map
0400; 0450; Case map
0401; 0451; Case map
0402; 0452; Case map
0403; 0453; Case map
0404; 0454; Case map
```

```
0405; 0455; Case map
0406; 0456; Case map
0407; 0457; Case map
0408; 0458; Case map
0409; 0459; Case map
040A; 045A; Case map
040B; 045B; Case map
040C; 045C; Case map
040D; 045D; Case map
040E; 045E; Case map
040F; 045F; Case map
0410; 0430; Case map
0411; 0431; Case map
0412; 0432; Case map
0413; 0433; Case map
0414; 0434; Case map
0415; 0435; Case map
0416; 0436; Case map
0417; 0437; Case map
0418; 0438; Case map
0419; 0439; Case map
041A; 043A; Case map
041B; 043B; Case map
041C; 043C; Case map
041D; 043D; Case map
041E; 043E; Case map
041F; 043F; Case map
0420; 0440; Case map
0421; 0441; Case map
0422; 0442; Case map
0423; 0443; Case map
0424; 0444; Case map
0425; 0445; Case map
0426; 0446; Case map
0427; 0447; Case map
0428; 0448; Case map
0429; 0449; Case map
042A; 044A; Case map
042B; 044B; Case map
042C; 044C; Case map
042D; 044D; Case map
042E; 044E; Case map
042F; 044F; Case map
0460; 0461; Case map
0462; 0463; Case map
0464; 0465; Case map
0466; 0467; Case map
0468; 0469; Case map
```

```
046A; 046B; Case map
046C; 046D; Case map
046E; 046F; Case map
0470; 0471; Case map
0472; 0473; Case map
0474; 0475; Case map
0476; 0477; Case map
0478; 0479; Case map
047A; 047B; Case map
047C; 047D; Case map
047E; 047F; Case map
0480; 0481; Case map
048A; 048B; Case map
048C; 048D; Case map
048E; 048F; Case map
0490; 0491; Case map
0492; 0493; Case map
0494; 0495; Case map
0496; 0497; Case map
0498; 0499; Case map
049A; 049B; Case map
049C; 049D; Case map
049E; 049F; Case map
04A0; 04A1; Case map
04A2; 04A3; Case map
04A4; 04A5; Case map
04A6; 04A7; Case map
04A8; 04A9; Case map
04AA; 04AB; Case map
04AC; 04AD; Case map
04AE; 04AF; Case map
04B0; 04B1; Case map
04B2; 04B3; Case map
04B4; 04B5; Case map
04B6; 04B7; Case map
04B8; 04B9; Case map
04BA; 04BB; Case map
04BC; 04BD; Case map
04BE; 04BF; Case map
04C1; 04C2; Case map
04C3; 04C4; Case map
04C5; 04C6; Case map
04C7; 04C8; Case map
04C9; 04CA; Case map
04CB; 04CC; Case map
04CD; 04CE; Case map
04D0; 04D1; Case map
04D2; 04D3; Case map
```

```
04D4; 04D5; Case map
04D6; 04D7; Case map
04D8; 04D9; Case map
04DA; 04DB; Case map
04DC; 04DD; Case map
04DE; 04DF; Case map
04E0; 04E1; Case map
04E2; 04E3; Case map
04E4; 04E5; Case map
04E6; 04E7; Case map
04E8; 04E9; Case map
04EA; 04EB; Case map
04EC; 04ED; Case map
04EE; 04EF; Case map
04F0; 04F1; Case map
04F2; 04F3; Case map
04F4; 04F5; Case map
04F8; 04F9; Case map
0500; 0501; Case map
0502; 0503; Case map
0504; 0505; Case map
0506; 0507; Case map
0508; 0509; Case map
050A; 050B; Case map
050C; 050D; Case map
050E; 050F; Case map
0531; 0561; Case map
0532; 0562; Case map
0533; 0563; Case map
0534; 0564; Case map
0535; 0565; Case map
0536; 0566; Case map
0537; 0567; Case map
0538; 0568; Case map
0539; 0569; Case map
053A; 056A; Case map
053B; 056B; Case map
053C; 056C; Case map
053D; 056D; Case map
053E; 056E; Case map
053F; 056F; Case map
0540; 0570; Case map
0541; 0571; Case map
0542; 0572; Case map
0543; 0573; Case map
0544; 0574; Case map
0545; 0575; Case map
0546; 0576; Case map
```

```
0547; 0577; Case map
0548; 0578; Case map
0549; 0579; Case map
054A; 057A; Case map
054B; 057B; Case map
054C; 057C; Case map
054D; 057D; Case map
054E; 057E; Case map
054F; 057F; Case map
0550; 0580; Case map
0551; 0581; Case map
0552; 0582; Case map
0553; 0583; Case map
0554; 0584; Case map
0555; 0585; Case map
0556; 0586; Case map
0587; 0565 0582; Case map
1E00; 1E01; Case map
1E02; 1E03; Case map
1E04; 1E05; Case map
1E06; 1E07; Case map
1E08; 1E09; Case map
1E0A; 1E0B; Case map
1E0C; 1E0D; Case map
1E0E; 1E0F; Case map
1E10; 1E11; Case map
1E12; 1E13; Case map
1E14; 1E15; Case map
1E16; 1E17; Case map
1E18; 1E19; Case map
1E1A; 1E1B; Case map
1E1C; 1E1D; Case map
1E1E; 1E1F; Case map
1E20; 1E21; Case map
1E22; 1E23; Case map
1E24; 1E25; Case map
1E26; 1E27; Case map
1E28; 1E29; Case map
1E2A; 1E2B; Case map
1E2C; 1E2D; Case map
1E2E; 1E2F; Case map
1E30; 1E31; Case map
1E32; 1E33; Case map
1E34; 1E35; Case map
1E36; 1E37; Case map
1E38; 1E39; Case map
1E3A; 1E3B; Case map
1E3C; 1E3D; Case map
```

```
1E3E; 1E3F; Case map
1E40; 1E41; Case map
1E42; 1E43; Case map
1E44; 1E45; Case map
1E46; 1E47; Case map
1E48; 1E49; Case map
1E4A; 1E4B; Case map
1E4C; 1E4D; Case map
1E4E; 1E4F; Case map
1E50; 1E51; Case map
1E52; 1E53; Case map
1E54; 1E55; Case map
1E56; 1E57; Case map
1E58; 1E59; Case map
1E5A; 1E5B; Case map
1E5C; 1E5D; Case map
1E5E; 1E5F; Case map
1E60; 1E61; Case map
1E62; 1E63; Case map
1E64; 1E65; Case map
1E66; 1E67; Case map
1E68; 1E69; Case map
1E6A; 1E6B; Case map
1E6C; 1E6D; Case map
1E6E; 1E6F; Case map
1E70; 1E71; Case map
1E72; 1E73; Case map
1E74; 1E75; Case map
1E76; 1E77; Case map
1E78; 1E79; Case map
1E7A; 1E7B; Case map
1E7C; 1E7D; Case map
1E7E; 1E7F; Case map
1E80; 1E81; Case map
1E82; 1E83; Case map
1E84; 1E85; Case map
1E86; 1E87; Case map
1E88; 1E89; Case map
1E8A; 1E8B; Case map
1E8C; 1E8D; Case map
1E8E; 1E8F; Case map
1E90; 1E91; Case map
1E92; 1E93; Case map
1E94; 1E95; Case map
1E96; 0068 0331; Case map
1E97; 0074 0308; Case map
1E98; 0077 030A; Case map
1E99; 0079 030A; Case map
```

```
1E9A; 0061 02BE; Case map
1E9B; 1E61; Case map
1EA0; 1EA1; Case map
1EA2; 1EA3; Case map
1EA4; 1EA5; Case map
1EA6; 1EA7; Case map
1EA8; 1EA9; Case map
1EAA; 1EAB; Case map
1EAC; 1EAD; Case map
1EAE; 1EAF; Case map
1EB0; 1EB1; Case map
1EB2; 1EB3; Case map
1EB4; 1EB5; Case map
1EB6; 1EB7; Case map
1EB8; 1EB9; Case map
1EBA; 1EBB; Case map
1EBC; 1EBD; Case map
1EBE; 1EBF; Case map
1EC0; 1EC1; Case map
1EC2; 1EC3; Case map
1EC4; 1EC5; Case map
1EC6; 1EC7; Case map
1EC8; 1EC9; Case map
1ECA; 1ECB; Case map
1ECC; 1ECD; Case map
1ECE; 1ECF; Case map
1ED0; 1ED1; Case map
1ED2; 1ED3; Case map
1ED4; 1ED5; Case map
1ED6; 1ED7; Case map
1ED8; 1ED9; Case map
1EDA; 1EDB; Case map
1EDC; 1EDD; Case map
1EDE; 1EDF; Case map
1EE0; 1EE1; Case map
1EE2; 1EE3; Case map
1EE4; 1EE5; Case map
1EE6; 1EE7; Case map
1EE8; 1EE9; Case map
1EEA; 1EEB; Case map
1EEC; 1EED; Case map
1EEE; 1EEF; Case map
1EF0; 1EF1; Case map
1EF2; 1EF3; Case map
1EF4; 1EF5; Case map
1EF6; 1EF7; Case map
1EF8; 1EF9; Case map
1F08; 1F00; Case map
```

```
1F09; 1F01; Case map
1F0A; 1F02; Case map
1F0B; 1F03; Case map
1F0C; 1F04; Case map
1F0D; 1F05; Case map
1F0E; 1F06; Case map
1F0F; 1F07; Case map
1F18; 1F10; Case map
1F19; 1F11; Case map
1F1A; 1F12; Case map
1F1B; 1F13; Case map
1F1C; 1F14; Case map
1F1D; 1F15; Case map
1F28; 1F20; Case map
1F29; 1F21; Case map
1F2A; 1F22; Case map
1F2B; 1F23; Case map
1F2C; 1F24; Case map
1F2D; 1F25; Case map
1F2E; 1F26; Case map
1F2F; 1F27; Case map
1F38; 1F30; Case map
1F39; 1F31; Case map
1F3A; 1F32; Case map
1F3B; 1F33; Case map
1F3C; 1F34; Case map
1F3D; 1F35; Case map
1F3E; 1F36; Case map
1F3F; 1F37; Case map
1F48; 1F40; Case map
1F49; 1F41; Case map
1F4A; 1F42; Case map
1F4B; 1F43; Case map
1F4C; 1F44; Case map
1F4D; 1F45; Case map
1F50; 03C5 0313; Case map
1F52; 03C5 0313 0300; Case map
1F54; 03C5 0313 0301; Case map
1F56; 03C5 0313 0342; Case map
1F59; 1F51; Case map
1F5B; 1F53; Case map
1F5D; 1F55; Case map
1F5F; 1F57; Case map
1F68; 1F60; Case map
1F69; 1F61; Case map
1F6A; 1F62; Case map
1F6B; 1F63; Case map
1F6C; 1F64; Case map
```

```
1F6D; 1F65; Case map
1F6E; 1F66; Case map
1F6F; 1F67; Case map
1F80; 1F00 03B9; Case map
1F81; 1F01 03B9; Case map
1F82; 1F02 03B9; Case map
1F83; 1F03 03B9; Case map
1F84; 1F04 03B9; Case map
1F85; 1F05 03B9; Case map
1F86; 1F06 03B9; Case map
1F87; 1F07 03B9; Case map
1F88; 1F00 03B9; Case map
1F89; 1F01 03B9; Case map
1F8A; 1F02 03B9; Case map
1F8B; 1F03 03B9; Case map
1F8C; 1F04 03B9; Case map
1F8D; 1F05 03B9; Case map
1F8E; 1F06 03B9; Case map
1F8F; 1F07 03B9; Case map
1F90; 1F20 03B9; Case map
1F91; 1F21 03B9; Case map
1F92; 1F22 03B9; Case map
1F93; 1F23 03B9; Case map
1F94; 1F24 03B9; Case map
1F95; 1F25 03B9; Case map
1F96; 1F26 03B9; Case map
1F97; 1F27 03B9; Case map
1F98; 1F20 03B9; Case map
1F99; 1F21 03B9; Case map
1F9A; 1F22 03B9; Case map
1F9B; 1F23 03B9; Case map
1F9C; 1F24 03B9; Case map
1F9D; 1F25 03B9; Case map
1F9E; 1F26 03B9; Case map
1F9F; 1F27 03B9; Case map
1FA0; 1F60 03B9; Case map
1FA1; 1F61 03B9; Case map
1FA2; 1F62 03B9; Case map
1FA3; 1F63 03B9; Case map
1FA4; 1F64 03B9; Case map
1FA5; 1F65 03B9; Case map
1FA6; 1F66 03B9; Case map
1FA7; 1F67 03B9; Case map
1FA8; 1F60 03B9; Case map
1FA9; 1F61 03B9; Case map
1FAA; 1F62 03B9; Case map
1FAB; 1F63 03B9; Case map
1FAC; 1F64 03B9; Case map
```

```
1FAD; 1F65 03B9; Case map
1FAE; 1F66 03B9; Case map
1FAF; 1F67 03B9; Case map
1FB2; 1F70 03B9; Case map
1FB3; 03B1 03B9; Case map
1FB4; 03AC 03B9; Case map
1FB6; 03B1 0342; Case map
1FB7; 03B1 0342 03B9; Case map
1FB8; 1FB0; Case map
1FB9; 1FB1; Case map
1FBA; 1F70; Case map
1FBB; 1F71; Case map
1FBC; 03B1 03B9; Case map
1FBE; 03B9; Case map
1FC2; 1F74 03B9; Case map
1FC3; 03B7 03B9; Case map
1FC4; 03AE 03B9; Case map
1FC6; 03B7 0342; Case map
1FC7; 03B7 0342 03B9; Case map
1FC8; 1F72; Case map
1FC9; 1F73; Case map
1FCA; 1F74; Case map
1FCB; 1F75; Case map
1FCC; 03B7 03B9; Case map
1FD2; 03B9 0308 0300; Case map
1FD3; 03B9 0308 0301; Case map
1FD6; 03B9 0342; Case map
1FD7; 03B9 0308 0342; Case map
1FD8; 1FD0; Case map
1FD9; 1FD1; Case map
1FDA; 1F76; Case map
1FDB; 1F77; Case map
1FE2; 03C5 0308 0300; Case map
1FE3; 03C5 0308 0301; Case map
1FE4; 03C1 0313; Case map
1FE6; 03C5 0342; Case map
1FE7; 03C5 0308 0342; Case map
1FE8; 1FE0; Case map
1FE9; 1FE1; Case map
1FEA; 1F7A; Case map
1FEB; 1F7B; Case map
1FEC; 1FE5; Case map
1FF2; 1F7C 03B9; Case map
1FF3; 03C9 03B9; Case map
1FF4; 03CE 03B9; Case map
1FF6; 03C9 0342; Case map
1FF7; 03C9 0342 03B9; Case map
1FF8; 1F78; Case map
```

```
1FF9; 1F79; Case map
1FFA; 1F7C; Case map
1FFB; 1F7D; Case map
1FFC; 03C9 03B9; Case map
2126; 03C9; Case map
212A; 006B; Case map
212B; 00E5; Case map
2160; 2170; Case map
2161; 2171; Case map
2162; 2172; Case map
2163; 2173; Case map
2164; 2174; Case map
2165; 2175; Case map
2166; 2176; Case map
2167; 2177; Case map
2168; 2178; Case map
2169; 2179; Case map
216A; 217A; Case map
216B; 217B; Case map
216C; 217C; Case map
216D; 217D; Case map
216E; 217E; Case map
216F; 217F; Case map
24B6; 24D0; Case map
24B7; 24D1; Case map
24B8; 24D2; Case map
24B9; 24D3; Case map
24BA; 24D4; Case map
24BB; 24D5; Case map
24BC; 24D6; Case map
24BD; 24D7; Case map
24BE; 24D8; Case map
24BF; 24D9; Case map
24C0; 24DA; Case map
24C1; 24DB; Case map
24C2; 24DC; Case map
24C3; 24DD; Case map
24C4; 24DE; Case map
24C5; 24DF; Case map
24C6; 24E0; Case map
24C7; 24E1; Case map
24C8; 24E2; Case map
24C9; 24E3; Case map
24CA; 24E4; Case map
24CB; 24E5; Case map
24CC; 24E6; Case map
24CD; 24E7; Case map
24CE; 24E8; Case map
```

```
24CF; 24E9; Case map
FB00; 0066 0066; Case map
FB01; 0066 0069; Case map
FB02; 0066 006C; Case map
FB03; 0066 0066 0069; Case map
FB04; 0066 0066 006C; Case map
FB05; 0073 0074; Case map
FB06; 0073 0074; Case map
FB13; 0574 0576; Case map
FB14; 0574 0565; Case map
FB15; 0574 056B; Case map
FB16; 057E 0576; Case map
FB17; 0574 056D; Case map
FF21; FF41; Case map
FF22; FF42; Case map
FF23; FF43; Case map
FF24; FF44; Case map
FF25; FF45; Case map
FF26; FF46; Case map
FF27; FF47; Case map
FF28; FF48; Case map
FF29; FF49; Case map
FF2A; FF4A; Case map
FF2B; FF4B; Case map
FF2C; FF4C; Case map
FF2D; FF4D; Case map
FF2E; FF4E; Case map
FF2F; FF4F; Case map
FF30; FF50; Case map
FF31; FF51; Case map
FF32; FF52; Case map
FF33; FF53; Case map
FF34; FF54; Case map
FF35; FF55; Case map
FF36; FF56; Case map
FF37; FF57; Case map
FF38; FF58; Case map
FF39; FF59; Case map
FF3A; FF5A; Case map
10400; 10428; Case map
10401; 10429; Case map
10402; 1042A; Case map
10403; 1042B; Case map
10404; 1042C; Case map
10405; 1042D; Case map
10406; 1042E; Case map
10407; 1042F; Case map
10408; 10430; Case map
```

```
10409; 10431; Case map
1040A; 10432; Case map
1040B; 10433; Case map
1040C; 10434; Case map
1040D; 10435; Case map
1040E; 10436; Case map
1040F; 10437; Case map
10410; 10438; Case map
10411; 10439; Case map
10412; 1043A; Case map
10413; 1043B; Case map
10414; 1043C; Case map
10415; 1043D; Case map
10416; 1043E; Case map
10417; 1043F; Case map
10418; 10440; Case map
10419; 10441; Case map
1041A; 10442; Case map
1041B; 10443; Case map
1041C; 10444; Case map
1041D; 10445; Case map
1041E; 10446; Case map
1041F; 10447; Case map
10420; 10448; Case map
10421; 10449; Case map
10422; 1044A; Case map
10423; 1044B; Case map
10424; 1044C; Case map
10425; 1044D; Case map
----- End Table B.3 -----
```

## C. Prohibition tables

The tables in this appendix consist of lines with one prohibited code point per line. The format of the lines are the value of the code point, a semicolon, and a comment which is the name of the code point.

### C.1 Space characters

#### C.1.1 ASCII space characters

```
----- Start Table C.1.1 -----
0020; SPACE
----- End Table C.1.1 -----
```

### C.1.2 Non-ASCII space characters

```
   ----- Start Table C.1.2 -----
   00A0; NO-BREAK SPACE
   1680; OGHAM SPACE MARK
   2000; EN QUAD
   2001; EM QUAD
   2002; EN SPACE
   2003; EM SPACE
   2004; THREE-PER-EM SPACE
   2005; FOUR-PER-EM SPACE
   2006; SIX-PER-EM SPACE
   2007; FIGURE SPACE
   2008; PUNCTUATION SPACE
   2009; THIN SPACE
   200A; HAIR SPACE
   200B; ZERO WIDTH SPACE
   202F; NARROW NO-BREAK SPACE
   205F; MEDIUM MATHEMATICAL SPACE
   3000; IDEOGRAPHIC SPACE
   ----- End Table C.1.2 -----
```

## C.2 Control characters

### C.2.1 ASCII control characters

```
   ----- Start Table C.2.1 -----
   0000-001F; [CONTROL CHARACTERS]
   007F; DELETE
   ----- End Table C.2.1 -----
```

### C.2.2 Non-ASCII control characters

```
   ----- Start Table C.2.2 -----
   0080-009F; [CONTROL CHARACTERS]
   06DD; ARABIC END OF AYAH
   070F; SYRIAC ABBREVIATION MARK
   180E; MONGOLIAN VOWEL SEPARATOR
   200C; ZERO WIDTH NON-JOINER
   200D; ZERO WIDTH JOINER
   2028; LINE SEPARATOR
   2029; PARAGRAPH SEPARATOR
   2060; WORD JOINER
   2061; FUNCTION APPLICATION
   2062; INVISIBLE TIMES
   2063; INVISIBLE SEPARATOR
   206A-206F; [CONTROL CHARACTERS]
   FEFF; ZERO WIDTH NO-BREAK SPACE
   FFF9-FFFC; [CONTROL CHARACTERS]
```

```
   1D173-1D17A; [MUSICAL CONTROL CHARACTERS]
   ----- End Table C.2.2 -----
```

## C.3 Private use

```
   ----- Start Table C.3 -----
   E000-F8FF; [PRIVATE USE, PLANE 0]
   F0000-FFFFD; [PRIVATE USE, PLANE 15]
   100000-10FFFD; [PRIVATE USE, PLANE 16]
   ----- End Table C.3 -----
```

## C.4 Non-character code points

```
   ----- Start Table C.4 -----
   FDD0-FDEF; [NONCHARACTER CODE POINTS]
   FFFE-FFFF; [NONCHARACTER CODE POINTS]
   1FFFE-1FFFF; [NONCHARACTER CODE POINTS]
   2FFFE-2FFFF; [NONCHARACTER CODE POINTS]
   3FFFE-3FFFF; [NONCHARACTER CODE POINTS]
   4FFFE-4FFFF; [NONCHARACTER CODE POINTS]
   5FFFE-5FFFF; [NONCHARACTER CODE POINTS]
   6FFFE-6FFFF; [NONCHARACTER CODE POINTS]
   7FFFE-7FFFF; [NONCHARACTER CODE POINTS]
   8FFFE-8FFFF; [NONCHARACTER CODE POINTS]
   9FFFE-9FFFF; [NONCHARACTER CODE POINTS]
   AFFFE-AFFFF; [NONCHARACTER CODE POINTS]
   BFFFE-BFFFF; [NONCHARACTER CODE POINTS]
   CFFFE-CFFFF; [NONCHARACTER CODE POINTS]
   DFFFE-DFFFF; [NONCHARACTER CODE POINTS]
   EFFFE-EFFFF; [NONCHARACTER CODE POINTS]
   FFFFE-FFFFF; [NONCHARACTER CODE POINTS]
   10FFFE-10FFFF; [NONCHARACTER CODE POINTS]
   ----- End Table C.4 -----
```

## C.5 Surrogate codes

```
   ----- Start Table C.5 -----
   D800-DFFF; [SURROGATE CODES]
   ----- End Table C.5 -----
```

## C.6 Inappropriate for plain text

```
   ----- Start Table C.6 -----
   FFF9; INTERLINEAR ANNOTATION ANCHOR
   FFFA; INTERLINEAR ANNOTATION SEPARATOR
   FFFB; INTERLINEAR ANNOTATION TERMINATOR
   FFFC; OBJECT REPLACEMENT CHARACTER
   FFFD; REPLACEMENT CHARACTER
```

```
   ----- End Table C.6 -----
```

### C.7 Inappropriate for canonical representation

```
   ----- Start Table C.7 -----
   2FF0-2FFB; [IDEOGRAPHIC DESCRIPTION CHARACTERS]
   ----- End Table C.7 -----
```

### C.8 Change display properties or are deprecated

```
   ----- Start Table C.8 -----
   0340; COMBINING GRAVE TONE MARK
   0341; COMBINING ACUTE TONE MARK
   200E; LEFT-TO-RIGHT MARK
   200F; RIGHT-TO-LEFT MARK
   202A; LEFT-TO-RIGHT EMBEDDING
   202B; RIGHT-TO-LEFT EMBEDDING
   202C; POP DIRECTIONAL FORMATTING
   202D; LEFT-TO-RIGHT OVERRIDE
   202E; RIGHT-TO-LEFT OVERRIDE
   206A; INHIBIT SYMMETRIC SWAPPING
   206B; ACTIVATE SYMMETRIC SWAPPING
   206C; INHIBIT ARABIC FORM SHAPING
   206D; ACTIVATE ARABIC FORM SHAPING
   206E; NATIONAL DIGIT SHAPES
   206F; NOMINAL DIGIT SHAPES
   ----- End Table C.8 -----
```

### C.9 Tagging characters

```
   ----- Start Table C.9 -----
   E0001; LANGUAGE TAG
   E0020-E007F; [TAGGING CHARACTERS]
   ----- End Table C.9 -----
```

## D. Bidirectional tables

### D.1 Characters with bidirectional property "R" or "AL"

```
   ----- Start Table D.1 -----
   05BE
   05C0
   05C3
   05D0-05EA
   05F0-05F4
   061B
   061F
   0621-063A
```

```
0640-064A
066D-066F
0671-06D5
06DD
06E5-06E6
06FA-06FE
0700-070D
0710
0712-072C
0780-07A5
07B1
200F
FB1D
FB1F-FB28
FB2A-FB36
FB38-FB3C
FB3E
FB40-FB41
FB43-FB44
FB46-FBB1
FBD3-FD3D
FD50-FD8F
FD92-FDC7
FDF0-FDFC
FE70-FE74
FE76-FEFC
----- End Table D.1 -----
```

## D.2 Characters with bidirectional property "L"

```
----- Start Table D.2 -----
0041-005A
0061-007A
00AA
00B5
00BA
00C0-00D6
00D8-00F6
00F8-0220
0222-0233
0250-02AD
02B0-02B8
02BB-02C1
02D0-02D1
02E0-02E4
02EE
037A
0386
```

0388-038A
038C
038E-03A1
03A3-03CE
03D0-03F5
0400-0482
048A-04CE
04D0-04F5
04F8-04F9
0500-050F
0531-0556
0559-055F
0561-0587
0589
0903
0905-0939
093D-0940
0949-094C
0950
0958-0961
0964-0970
0982-0983
0985-098C
098F-0990
0993-09A8
09AA-09B0
09B2
09B6-09B9
09BE-09C0
09C7-09C8
09CB-09CC
09D7
09DC-09DD
09DF-09E1
09E6-09F1
09F4-09FA
0A05-0A0A
0A0F-0A10
0A13-0A28
0A2A-0A30
0A32-0A33
0A35-0A36
0A38-0A39
0A3E-0A40
0A59-0A5C
0A5E
0A66-0A6F
0A72-0A74

```
0A83
0A85-0A8B
0A8D
0A8F-0A91
0A93-0AA8
0AAA-0AB0
0AB2-0AB3
0AB5-0AB9
0ABD-0AC0
0AC9
0ACB-0ACC
0AD0
0AE0
0AE6-0AEF
0B02-0B03
0B05-0B0C
0B0F-0B10
0B13-0B28
0B2A-0B30
0B32-0B33
0B36-0B39
0B3D-0B3E
0B40
0B47-0B48
0B4B-0B4C
0B57
0B5C-0B5D
0B5F-0B61
0B66-0B70
0B83
0B85-0B8A
0B8E-0B90
0B92-0B95
0B99-0B9A
0B9C
0B9E-0B9F
0BA3-0BA4
0BA8-0BAA
0BAE-0BB5
0BB7-0BB9
0BBE-0BBF
0BC1-0BC2
0BC6-0BC8
0BCA-0BCC
0BD7
0BE7-0BF2
0C01-0C03
0C05-0C0C
```

0C0E-0C10
0C12-0C28
0C2A-0C33
0C35-0C39
0C41-0C44
0C60-0C61
0C66-0C6F
0C82-0C83
0C85-0C8C
0C8E-0C90
0C92-0CA8
0CAA-0CB3
0CB5-0CB9
0CBE
0CC0-0CC4
0CC7-0CC8
0CCA-0CCB
0CD5-0CD6
0CDE
0CE0-0CE1
0CE6-0CEF
0D02-0D03
0D05-0D0C
0D0E-0D10
0D12-0D28
0D2A-0D39
0D3E-0D40
0D46-0D48
0D4A-0D4C
0D57
0D60-0D61
0D66-0D6F
0D82-0D83
0D85-0D96
0D9A-0DB1
0DB3-0DBB
0DBD
0DC0-0DC6
0DCF-0DD1
0DD8-0DDF
0DF2-0DF4
0E01-0E30
0E32-0E33
0E40-0E46
0E4F-0E5B
0E81-0E82
0E84
0E87-0E88

0E8A
0E8D
0E94-0E97
0E99-0E9F
0EA1-0EA3
0EA5
0EA7
0EAA-0EAB
0EAD-0EB0
0EB2-0EB3
0EBD
0EC0-0EC4
0EC6
0ED0-0ED9
0EDC-0EDD
0F00-0F17
0F1A-0F34
0F36
0F38
0F3E-0F47
0F49-0F6A
0F7F
0F85
0F88-0F8B
0FBE-0FC5
0FC7-0FCC
0FCF
1000-1021
1023-1027
1029-102A
102C
1031
1038
1040-1057
10A0-10C5
10D0-10F8
10FB
1100-1159
115F-11A2
11A8-11F9
1200-1206
1208-1246
1248
124A-124D
1250-1256
1258
125A-125D
1260-1286

1288
128A-128D
1290-12AE
12B0
12B2-12B5
12B8-12BE
12C0
12C2-12C5
12C8-12CE
12D0-12D6
12D8-12EE
12F0-130E
1310
1312-1315
1318-131E
1320-1346
1348-135A
1361-137C
13A0-13F4
1401-1676
1681-169A
16A0-16F0
1700-170C
170E-1711
1720-1731
1735-1736
1740-1751
1760-176C
176E-1770
1780-17B6
17BE-17C5
17C7-17C8
17D4-17DA
17DC
17E0-17E9
1810-1819
1820-1877
1880-18A8
1E00-1E9B
1EA0-1EF9
1F00-1F15
1F18-1F1D
1F20-1F45
1F48-1F4D
1F50-1F57
1F59
1F5B
1F5D

1F5F-1F7D
1F80-1FB4
1FB6-1FBC
1FBE
1FC2-1FC4
1FC6-1FCC
1FD0-1FD3
1FD6-1FDB
1FE0-1FEC
1FF2-1FF4
1FF6-1FFC
200E
2071
207F
2102
2107
210A-2113
2115
2119-211D
2124
2126
2128
212A-212D
212F-2131
2133-2139
213D-213F
2145-2149
2160-2183
2336-237A
2395
249C-24E9
3005-3007
3021-3029
3031-3035
3038-303C
3041-3096
309D-309F
30A1-30FA
30FC-30FF
3105-312C
3131-318E
3190-31B7
31F0-321C
3220-3243
3260-327B
327F-32B0
32C0-32CB
32D0-32FE

3300-3376
337B-33DD
33E0-33FE
3400-4DB5
4E00-9FA5
A000-A48C
AC00-D7A3
D800-FA2D
FA30-FA6A
FB00-FB06
FB13-FB17
FF21-FF3A
FF41-FF5A
FF66-FFBE
FFC2-FFC7
FFCA-FFCF
FFD2-FFD7
FFDA-FFDC
10300-1031E
10320-10323
10330-1034A
10400-10425
10428-1044D
1D000-1D0F5
1D100-1D126
1D12A-1D166
1D16A-1D172
1D183-1D184
1D18C-1D1A9
1D1AE-1D1DD
1D400-1D454
1D456-1D49C
1D49E-1D49F
1D4A2
1D4A5-1D4A6
1D4A9-1D4AC
1D4AE-1D4B9
1D4BB
1D4BD-1D4C0
1D4C2-1D4C3
1D4C5-1D505
1D507-1D50A
1D50D-1D514
1D516-1D51C
1D51E-1D539
1D53B-1D53E
1D540-1D544
1D546

```
   1D54A-1D550
   1D552-1D6A3
   1D6A8-1D7C9
   20000-2A6D6
   2F800-2FA1D
   F0000-FFFFD
   100000-10FFFD
   ----- End Table D.2 -----
```

## Authors' Addresses

Paul Hoffman
Internet Mail Consortium and VPN Consortium
127 Segre Place
Santa Cruz, CA 95060 USA

EMail: paul.hoffman@imc.org and paul.hoffman@vpnc.org

Marc Blanchet
Viagenie inc.
2875 boul. Laurier, bur. 300
Ste-Foy, Quebec, Canada, G1V 2M2

EMail: Marc.Blanchet@viagenie.qc.ca

## Full Copyright Statement



## Acknowledgement

Funding for the RFC Editor function is currently provided by the Internet Society.