PRECIS
Internet-Draft
Obsoletes: 3454 (if approved)
Intended status: Standards Track
Expires: June 8, 2014

P. Saint-Andre
Cisco Systems, Inc.
M. Blanchet
Viagenie

December 5, 2013

PRECIS Framework: Preparation and Comparison of Internationalized Strings in Application Protocols draft-ietf-precis-framework-13

Abstract

Application protocols using Unicode characters in protocol strings need to properly prepare such strings in order to perform valid comparison operations (e.g., for purposes of authentication or authorization). This document defines a framework enabling application protocols to perform the preparation and comparison of internationalized strings ("PRECIS") in a way that depends on the properties of Unicode characters and thus is agile with respect to versions of Unicode. As a result, this framework provides a more sustainable approach to the handling of internationalized strings than the previous framework, known as Stringprep (RFC 3454). This document obsoletes RFC 3454.

Status of this Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at http://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on June 8, 2014.

Copyright Notice

Table of Contents

1. Introduction

As described in the problem statement for the preparation and comparison of internationalized strings ("PRECIS") [RFC6885], many IETF protocols have used the Stringprep framework [RFC3454] as the basis for preparing and comparing protocol strings that contain Unicode characters [UNICODE] outside the ASCII range [RFC20]. The Stringprep framework was developed during work on the original technology for internationalized domain names (IDNs), here called "IDNA2003" [RFC3490], and Nameprep [RFC3491] was the Stringprep profile for IDNs. At the time, Stringprep was designed as a general framework so that other application protocols could define their own Stringprep profiles for the preparation and comparison of strings and identifiers. Indeed, a number of application protocols defined such profiles.

After the publication of [RFC3454] in 2002, several significant issues arose with the use of Stringprep in the IDN case, as documented in the IAB's recommendations regarding IDNs [RFC4690] (most significantly, Stringprep was tied to Unicode version 3.2). Therefore, the newer IDNA specifications, here called "IDNA2008" ([RFC5890], [RFC5891], [RFC5892], [RFC5893], [RFC5894]), no longer use Stringprep and Nameprep. This migration away from Stringprep for IDNs has prompted other "customers" of Stringprep to consider new approaches to the preparation and comparison of internationalized strings, as described in [RFC6885].

This document defines a framework for a post-Stringprep approach to the preparation and comparison of internationalized strings in application protocols, based on several principles:

1. Define a small set of string classes that specify the Unicode characters (i.e., specific "code points") appropriate for common application protocol constructs.
2. Define each PRECIS string class in terms of Unicode code points and their properties so that an algorithm can be used to determine whether each code point or character category is (a) valid, (b) allowed in certain contexts, (c) disallowed, or (d) unassigned.
3. Use an "inclusion model" such that a string class consists only of code points that are explicitly allowed, with the result that any code point not explicitly allowed is forbidden.
4. Enable application protocols to define profiles of the PRECIS string classes, addressing matters such as width mapping, case folding and other forms of character mapping, Unicode normalization, directionality, and further excluded code points or character categories.

Whereas the string classes define the "baseline" code points for a range of applications, profiling enables application protocols to further restrict the allowable code points beyond those specified for the relevant string class (e.g., characters with special or reserved meaning, such as "@" and "/" when used as separators within identifiers) and to apply the string classes in ways that are appropriate for constructs such as usernames and passwords [I-D.ietf-precis-saslprepbis], nicknames [I-D.ietf-precis-nickname], the localparts of instant messaging addresses [I-D.ietf-xmpp-6122bis], and free-form strings [I-D.ietf-xmpp-6122bis]. Profiles are responsible for defining the handling of right-to-left characters as well as various mapping operations of the kind also discussed for IDNs in [RFC5895], such as case preservation or lowercasing, Unicode normalization, mapping of certain characters to other characters or to nothing, and mapping of full-width and half-width characters.

It is expected that this framework will yield the following benefits:

- Application protocols will be agile with regard to Unicode versions.
- Implementers will be able to share code point tables and software code across application protocols, most likely by means of software libraries.
- End users will be able to acquire more accurate expectations about the characters that are acceptable in various contexts. Given this more uniform set of string classes, it is also expected that copy/paste operations between software implementing different application protocols will be more predictable and coherent.

Although this framework is similar to IDNA2008 and borrows some of the character categories defined in [RFC5892], it defines additional character categories to meet the needs of common application protocols.

The character categories and calculation rules defined under Section 7 and Section 8 are normative and apply to all Unicode code points. The code point table provided under Appendix A is non-normative and merely shows, for illustrative purposes, the consequences of the character categories and calculation rules, as well as the resulting property values.

2. Terminology

Many important terms used in this document are defined in [RFC5890], [RFC6365], [RFC6885], and [UNICODE]. The terms "left-to-right" (LTR) and "right-to-left" (RTL) are defined in Unicode Standard Annex #9

[UAX9].

As of the date of writing, the version of Unicode published by the Unicode Consortium is 6.3; however, PRECIS is not tied to a specific version of Unicode.

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

3. String Classes

3.1. Overview

IDNA2008 essentially defines a string class of internationalized domain name (IDN), although it does not use the term "string class". (This document does not define a string class for domain names, and application protocols are strongly encouraged to use IDNA2008 as the appropriate method to prepare domain names and hostnames.) Because the IDN string class is designed to meet the particular requirements of the Domain Name System (DNS), additional string classes are needed for non-DNS applications.

Starting in 2010, various "customers" of Stringprep began to discuss the need to define a post-Stringprep approach to the preparation and comparison of internationalized strings other than IDNs. This community analyzed the existing Stringprep profiles and also weighed the costs and benefits of defining a relatively small set of Unicode characters that would minimize the potential for user confusion caused by visually similar characters (and thus be relatively "safe") vs. defining a much larger set of Unicode characters that would maximize the potential for user creativity (and thus be relatively "expressive"). As a result, the community concluded that most existing uses could be addressed by two string classes:

IdentifierClass: a sequence of letters, numbers, and some symbols that is used to identify or address a network entity such as a user account, a venue (e.g., a chatroom), an information source (e.g., a data feed), or a collection of data (e.g., a file); the intent is that this class will minimize user confusion in a wide variety of application protocols, with the result that safety has been prioritized over expressiveness for this class.

FreeformClass: a sequence of letters, numbers, symbols, spaces, and other characters that is used for free-form strings, including passwords as well as display elements such as human-friendly nicknames in chatrooms; the intent is that this class will allow nearly any Unicode character, with the result that expressiveness has been prioritized over safety for this class (e.g., protocol designers, application developers, service providers, and end users might not understand or be able to enter all of the characters that can be included in the FreeformClass).

Future specifications might define additional PRECIS string classes, such as a class that falls somewhere between the IdentifierClass and the FreeformClass. At this time, it is not clear how useful such a class would be. In any case, because application developers are able to define profiles of PRECIS string classes, a protocol needing a construct between the IdentiferClass and the FreeformClass could define a restricted profile of the FreeformClass if needed.

The following subsections discuss the IdentifierClass and FreeformClass in more detail, with reference to the dimensions described in Section 3 of [RFC6885]. Each string class is defined by the following behavioral rules:

Valid: Defines which code points and character categories are treated as valid input to the string.

Contextual Rule Required: Defines which code points and character categories are treated as allowed only if the requirements of a contextual rule are met (i.e., either CONTEXTJ or CONTEXTO).

Disallowed: Defines which code points and character categories need to be excluded from the string.

Unassigned: Defines application behavior in the presence of code points that are unknown (i.e., not yet designated) for the version of Unicode used by the application.

This document defines the valid, contextual rule required, disallowed, and unassigned rules for the IdentifierClass and FreeformClass. As described under Section 4, profiles of these string classes are responsible for defining the width mapping, additional mapping, case mapping, normalization, directionality, and exclusion rules.

3.2. IdentifierClass

Most application technologies need strings that can be used to refer to, include, or communicate protocol strings like usernames, file

names, data feed identifiers, and chatroom names. We group such strings into a class called "IdentifierClass" having the following features.

3.2.1. Valid

- Code points traditionally used as letters and numbers in writing systems, i.e., the LetterDigits ("A") category first defined in [RFC5892] and listed here under Section 7.1.
- Code points in the range U+0021 through U+007E, i.e., the (printable) ASCII7 ("K") rule defined under Section 7.11. These code points are "grandfathered" into PRECIS and thus are valid even if they would otherwise be disallowed according to the property-based rules specified in the next section.

Note: Although the PRECIS IdentifierClass re-uses the LetterDigits category from IDNA2008, the range of characters allowed in the IdentifierClass is wider than the range of characters allowed in IDNA2008. The main reason is that IDNA2008 applies the Unstable category before the LetterDigits category, thus disallowing uppercase characters, whereas the IdentifierClass does not apply the Unstable category.

3.2.2. Contextual Rule Required

- A number of characters from the Exceptions ("F") category defined under Section 7.6 (see Section 7.6 for a full list).
- Joining characters, i.e., the JoinControl ("H") category defined under Section 7.8.

3.2.3. Disallowed

- Old Hangul Jamo characters, i.e., the OldHangulJamo ("I") category defined under Section 7.9.
- Control characters, i.e., the Controls ("L") category defined under Section 7.12.
- Ignorable characters, i.e., the PrecisIgnorableProperties ("M") category defined under Section 7.13.
- Space characters, i.e., the Spaces ("N") category defined under Section 7.14.
- Symbol characters, i.e., the Symbols ("O") category defined under Section 7.15.
- Punctuation characters, i.e., the Punctuation ("P") category defined under Section 7.16.
- Any character that has a compatibility equivalent, i.e., the HasCompat ("Q") category defined under Section 7.17. These code points are disallowed even if they would otherwise be valid according to the property-based rules specified in the previous

section.
- Letters and digits other than the "traditional" letters and digits allowed in IDNs, i.e., the OtherLetterDigits ("R") category defined under Section 7.18.

3.2.4. Unassigned

Any code points that are not yet designated in the Unicode character set SHALL be considered Unassigned for purposes of the IdentifierClass, and a string containing such code points SHALL be rejected.

3.3. FreeformClass

Some application technologies need strings that can be used in a free-form way, e.g., as a password in an authentication exchange (see [I-D.ietf-precis-saslprepbis] or a nickname in a chatroom (see [I-D.ietf-precis-nickname]). We group such things into a class called "FreeformClass" having the following features.

Note: Consult Section 10.6 for relevant security considerations when strings conforming to the FreeformClass, or a profile thereof, are used as passwords.

3.3.1. Valid

- Traditional letters and numbers, i.e., the LetterDigits ("A") category first defined in [RFC5892] and listed here under Section 7.1.
- Letters and digits other than the "traditional" letters and digits allowed in IDNs, i.e., the OtherLetterDigits ("R") category defined under Section 7.18.
- Code points in the range U+0021 through U+007E, i.e., the (printable) ASCII7 ("K") rule defined under Section 7.11.
- Any character that has a compatibility equivalent, i.e., the HasCompat ("Q") category defined under Section 7.17.
- Space characters, i.e., the Spaces ("N") category defined under Section 7.14.
- Symbol characters, i.e., the Symbols ("O") category defined under Section 7.15.
- Punctuation characters, i.e., the Punctuation ("P") category defined under Section 7.16.

3.3.2. Contextual Rule Required

- A number of characters from the Exceptions ("F") category defined under Section 7.6 (see Section 7.6 for a full list).

- Joining characters, i.e., the JoinControl ("H") category defined under Section 7.8.

3.3.3. Disallowed

- Old Hangul Jamo characters, i.e., the OldHangulJamo ("I") category defined under Section 7.9.
- Control characters, i.e., the Controls ("L") category defined under Section 7.12.
- Ignorable characters, i.e., the PrecisIgnorableProperties ("M") category defined under Section 7.13.

3.3.4. Unassigned

Any code points that are not yet designated in the Unicode character set SHALL be considered Unassigned for purposes of the FreeformClass, and a string containing such code points SHALL be rejected.

4. Profiles

4.1. Principles

This framework document defines the valid, contextual-rule-required, disallowed, and unassigned rules for the IdentifierClass and the FreeformClass. A profile of a PRECIS string class MUST define the width mapping, additional mapping (if any), case mapping, normalization, directionality, and exclusion rules. A profile MAY also restrict the allowable characters above and beyond the definition of the relevant PRECIS string class (but MUST NOT add as valid any code points or character categories that are disallowed by the relevant PRECIS string class). These matters are discussed in the following subsections.

Profiles of the PRECIS string classes MUST register with the IANA as described under Section 9.3. It is RECOMMENDED for profile names to be of the form "ProfilenameBaseClass", where the "Profilename" string is a differentiator and "BaseClass" is the name of the PRECIS string class being profiled; for example, the profile of the IdentifierClass used for localparts of Jabber IDs in the Extensible Messaging and Presence Protocol (XMPP) is named "JIDlocalIdentifierClass" [I-D.ietf-xmpp-6122bis].

4.1.1. Width Mapping

The width mapping rule of a profile specifies whether width mapping is performed on fullwidth and halfwidth characters, and how the mapping is done. Typically such mapping consists of mapping

fullwidth and halfwidth characters, i.e., code points with a Decomposition Type of Wide or Narrow, to their decomposition mappings; as an example, FULLWIDTH DIGIT ZERO (U+FF10) would be mapped to DIGIT ZERO (U+0030).

The normalization form specified by a profile (see below) has an impact on the need for width mapping. Because width mapping is performed as a part of compatibility decomposition, a profile employing either normalization form KD (NFKD) or normalization form KC (NFKC) does not need to specify width mapping. However, if Unicode normalization form C (NFC) is used then the profile needs to specify whether to apply width mapping; in this case, width mapping is in general RECOMMENDED because allowing fullwidth and halfwidth characters to remain unmapped to their compatibility variants would violate the principle of least user surprise. For more information about the concept of width in East Asian scripts within Unicode, see Unicode Standard Annex #11 [UAX11].

4.1.2. Additional Mappings

The additional mappings rule of a profile specifies whether additional mappings are to be applied, such as mapping of delimiter characters, mapping of special characters (e.g., non-ASCII space characters to ASCII space or certain characters to nothing), and case mapping based on locale or on locale and context (see [I-D.ietf-precis-mappings]).

4.1.3. Case Mapping

The case mapping rule of a profile specifies whether case mapping is performed (instead of case preservation) on uppercase and titlecase characters, and how the mapping is done (e.g., mapping uppercase and titlecase characters to their lowercase equivalents).

If case preservation is not desired, it is RECOMMENDED to use Unicode Default Case Folding as defined in Chapter 3 of the Unicode Standard [UNICODE].

In order to maximize entropy and minimize the potential for false positives, it is NOT RECOMMENDED for application protocols to map uppercase and titlecase code points to their lowercase equivalents when strings conforming to the FreeformClass, or a profile thereof, are used in passwords; instead, it is RECOMMENDED to preserve the case of all code points contained in such strings and then perform case-sensitive comparison. See also the related discussion in [I-D.ietf-precis-saslprepbis].

4.1.4. Normalization

The normalization rule of a profile specifies which Unicode normalization form (D, KD, C, or KC) is to be applied (see Unicode Standard Annex #15 [UAX15] for background information).

In accordance with [RFC5198], normalization form C (NFC) is RECOMMENDED.

4.1.5. Directionality

The directionality rule of a profile specifies which strings are to be considered left-to-right (LTR) and right-to-left (RTL), and the allowable sequences of characters in LTR and RTL strings (see Unicode Standard Annex #9 [UAX9]); note that mixed-direction strings are not supported, since there is currently no widely accepted and implemented solution for the processing and display of mixed-direction strings. Possible rules include, but are not limited to, (a) considering any string that contains a right-to-left code point to be a right-to-left string, or (b) applying the "Bidi Rule" from [RFC5893].

4.1.6. Exclusions

The exclusions rule of a profile specifies whether the profile excludes additional code points or character categories above and beyond those excluded by the string class being profiled. That is, a profile MAY do either of the following:

1. Exclude specific code points that are allowed by the relevant string class.
2. Exclude characters matching certain Unicode properties (e.g., math symbols) that are included in the relevant PRECIS string class.

As a result of such exclusions, code points that are defined as valid for the PRECIS string class being profiled will be defined as disallowed for the profile.

4.2. Building Application-Layer Constructs

Sometimes, an application-layer construct does not map in a straightforward manner to one of the PRECIS string classes or a profile thereof. Consider, for example, the "simple user name" construct in the Simple Authentication and Security Layer (SASL) [RFC4422]. Depending on the deployment, a simple user name might take the form of a user's full name (e.g., the user's personal name followed by a space and then the user's family name). Such a simple

user name cannot be defined as an instance of the IdentifierClass or a profile thereof, since space characters are not allowed in the IdentifierClass; however, it could be defined using a space-separated sequence of IdentifierClass instances, as in the following pseudo-ABNF [RFC5234]:

```
fullname = namepart [1*(1*SP namepart)]
namepart = 1*(idpoint)
           ;
           ; an "idpoint" is a UTF-8 encoded Unicode code point
           ; that conforms to the PRECIS IdentifierClass
```

Similar techniques could be used to define many application-layer constructs, say of the form "user@domain" or "/path/to/file".

4.3. A Note about Spaces

With regard to the IdentiferClass, the consensus of the PRECIS Working Group was that spaces are problematic for many reasons, including:

- Many Unicode characters are confusable with ASCII space.
- Even if non-ASCII space characters are mapped to ASCII space (U+0020), space characters are often not rendered in user interfaces, leading to the possibility that a human user might consider a string containing spaces to be equivalent to the same string without spaces.
- In some locales, some devices are known to generate a character other than ASCII space (such as ZERO WIDTH JOINER, U+200D) when a user performs an action like hit the space bar on a keyboard.

One consequence of disallowing space characters in the IdentifierClass might be to effectively discourage the use of ASCII space (or, even more problematically, non-ASCII space characters) within identifiers created in newer application protocols; given the challenges involved in properly handling space characters in identifiers and other protocol strings, the Working Group considered this to be a feature, not a bug.

However, the FreeformClass does allow spaces, which enables application protocols to define profiles of the FreeformClass that are more flexible than any profiles of the IdentifierClass. In addition, as explained in the previous section, application protocols can also define application-layer constructs containing spaces.

5. Order of Operations

To ensure proper comparison, the following order of operations is REQUIRED:

1. Width mapping
2. Optionally, additional mappings such as those as specified in [I-D.ietf-precis-mappings]:
    1. Delimiter mapping
    2. Special mapping
    3. Local case mapping
3. Non-local case mapping
4. Normalization
5. Behavioral rules for determining whether a code point is valid, allowed under a contextual rule, disallowed, or unassigned

As already described, the width mapping, additional mapping, non-local case mapping, and normalization operations are specified for each profile, whereas the behavioral rules are specified for each string class. Some of the logic behind this order is provided under Section 4.1.1 and in [I-D.ietf-precis-mappings].

6. Code Point Properties

In order to implement the string classes described above, this document does the following:

1. Reviews and classifies the collections of code points in the Unicode character set by examining various code point properties.

2. Defines an algorithm for determining a derived property value, which can vary depending on the string class being used by the relevant application protocol.

This document is not intended to specify precisely how derived property values are to be applied in protocol strings. That information is the responsibility of the protocol specification that uses or profiles a PRECIS string class from this document.

The value of the property is to be interpreted as follows.

PROTOCOL VALID Those code points that are allowed to be used in any PRECIS string class (currently, IdentifierClass and FreeformClass). Code points with this property value are permitted for general use in any string class. The abbreviated term "PVALID" is used to refer to this value in the remainder of this document.

SPECIFIC CLASS PROTOCOL VALID Those code points that are allowed to be used in specific string classes. Code points with this property value are permitted for use in specific string classes. In the remainder of this document, the abbreviated term *_PVAL is used, where * = (ID | FREE), i.e., either "FREE_PVAL" or "ID_PVAL".

CONTEXTUAL RULE REQUIRED Some characteristics of the character, such as its being invisible in certain contexts or problematic in others, require that it not be used in labels unless specific other characters or properties are present. As in IDNA2008, there are two subdivisions of CONTEXTUAL RULE REQUIRED, the first for Join_controls (called "CONTEXTJ") and the second for other characters (called "CONTEXTO"). A character with the derived property value CONTEXTJ or CONTEXTO MUST NOT be used unless an appropriate rule has been established and the context of the character is consistent with that rule. The most notable of the CONTEXTUAL RULE REQUIRED characters are the Join Control characters U+200D ZERO WIDTH JOINER and U+200C ZERO WIDTH NON-JOINER, which have a derived property value of CONTEXTJ. See Appendix A of [RFC5892] for more information.

DISALLOWED Those code points that are not permitted in any PRECIS string class.

SPECIFIC CLASS DISALLOWED Those code points that are not to be included in a specific string class. Code points with this property value are not permitted in one of the string classes but might be permitted in others. In the remainder of this document, the abbreviated term *_DIS is used, where * = (ID | FREE), i.e., either "FREE_DIS" or "ID_DIS".

UNASSIGNED Those code points that are not designated (i.e. are unassigned) in the Unicode Standard.

The mechanisms described here allow determination of the value of the property for future versions of Unicode (including characters added after Unicode 5.2 or 6.3 depending on the category, since some categories in this document are reused from IDNA2008 and therefore were defined at the time of Unicode 5.2). Changes in Unicode properties that do not affect the outcome of this process therefore do not affect this framework. For example, a character can have its Unicode General_Category value [UNICODE] change from So to Sm, or from Lo to Ll, without affecting the algorithm results. Moreover, even if such changes were to result, the BackwardCompatible list (Section 7.7) can be adjusted to ensure the stability of the results.

7. Category Definitions Used to Calculate Derived Property

The derived property obtains its value based on a two-step procedure:

1. Characters are placed in one or more character categories either (1) based on core properties defined by the Unicode Standard or (2) by treating the code point as an exception and addressing the code point based on its code point value. These categories are not mutually exclusive.
2. Set operations are used with these categories to determine the values for a property specific to a given string class. These operations are specified under Section 8.

(Note: Unicode property names and property value names might have short abbreviations, such as "gc" for the General_Category property and "Ll" for the Lowercase_Letter property value of the gc property.)

In the following specification of character categories, the operation that returns the value of a particular Unicode character property for a code point is designated by using the formal name of that property (from the Unicode PropertyAliases.txt [1]) followed by '(cp)' for "code point". For example, the value of the General_Category property for a code point is indicated by General_Category(cp).

The first ten categories (A-J) shown below were previously defined for IDNA2008 and are copied directly from [RFC5892]. Some of these categories are reused in PRECIS and some of them are not; however, the lettering of categories is retained to prevent overlap and to ease implementation of both IDNA2008 and PRECIS in a single software application. The next eight categories (K-R) are specific to PRECIS.

7.1. LetterDigits (A)

Note: This category is defined in [RFC5892] and copied here for use in PRECIS.

A: General_Category(cp) is in {Ll, Lu, Lm, Lo, Mn, Mc, Nd}

These rules identify characters commonly used in mnemonics and often informally described as "language characters".

For more information, see Chapter 4 of the Unicode Standard [UNICODE].

The categories used in this rule are:
o Ll - Lowercase_Letter

- Lu - Uppercase_Letter
- Lm - Modifier_Letter
- Lo - Other_Letter
- Mn - Nonspacing_Mark
- Mc - Spacing_Mark
- Nd - Decimal_Number

7.2. Unstable (B)

Note: This category is defined in [RFC5892] but not used in PRECIS.

7.3. IgnorableProperties (C)

Note: This category is defined in [RFC5892] but not used in PRECIS. See the "PrecisIgnorableProperties (M)" category below for a more inclusive category used in PRECIS identifiers.

7.4. IgnorableBlocks (D)

Note: This category is defined in [RFC5892] but not used in PRECIS.

7.5. LDH (E)

Note: This category is defined in [RFC5892] but not used in PRECIS. See the "ASCII7 (K)" category below for a more inclusive category used in PRECIS identifiers.

7.6. Exceptions (F)

Note: This category is defined in [RFC5892] and used in PRECIS to ensure consistent treatment of the relevant code points.

```
F: cp is in {00B7, 00DF, 0375, 03C2, 05F3, 05F4, 0640, 0660,
             0661, 0662, 0663, 0664, 0665, 0666, 0667, 0668,
             0669, 06F0, 06F1, 06F2, 06F3, 06F4, 06F5, 06F6,
             06F7, 06F8, 06F9, 06FD, 06FE, 07FA, 0F0B, 3007,
             302E, 302F, 3031, 3032, 3033, 3034, 3035, 303B,
             30FB}
```

This category explicitly lists code points for which the category cannot be assigned using only the core property values that exist in the Unicode Standard. The values are according to the table below:

PVALID -- Would otherwise have been DISALLOWED

```
00DF; PVALID     # LATIN SMALL LETTER SHARP S
03C2; PVALID     # GREEK SMALL LETTER FINAL SIGMA
06FD; PVALID     # ARABIC SIGN SINDHI AMPERSAND
```

```
   06FE; PVALID     # ARABIC SIGN SINDHI POSTPOSITION MEN
   0F0B; PVALID     # TIBETAN MARK INTERSYLLABIC TSHEG
   3007; PVALID     # IDEOGRAPHIC NUMBER ZERO

   CONTEXTO -- Would otherwise have been DISALLOWED

   00B7; CONTEXTO   # MIDDLE DOT
   0375; CONTEXTO   # GREEK LOWER NUMERAL SIGN (KERAIA)
   05F3; CONTEXTO   # HEBREW PUNCTUATION GERESH
   05F4; CONTEXTO   # HEBREW PUNCTUATION GERSHAYIM
   30FB; CONTEXTO   # KATAKANA MIDDLE DOT

   CONTEXTO -- Would otherwise have been PVALID

   0660; CONTEXTO   # ARABIC-INDIC DIGIT ZERO
   0661; CONTEXTO   # ARABIC-INDIC DIGIT ONE
   0662; CONTEXTO   # ARABIC-INDIC DIGIT TWO
   0663; CONTEXTO   # ARABIC-INDIC DIGIT THREE
   0664; CONTEXTO   # ARABIC-INDIC DIGIT FOUR
   0665; CONTEXTO   # ARABIC-INDIC DIGIT FIVE
   0666; CONTEXTO   # ARABIC-INDIC DIGIT SIX
   0667; CONTEXTO   # ARABIC-INDIC DIGIT SEVEN
   0668; CONTEXTO   # ARABIC-INDIC DIGIT EIGHT
   0669; CONTEXTO   # ARABIC-INDIC DIGIT NINE
   06F0; CONTEXTO   # EXTENDED ARABIC-INDIC DIGIT ZERO
   06F1; CONTEXTO   # EXTENDED ARABIC-INDIC DIGIT ONE
   06F2; CONTEXTO   # EXTENDED ARABIC-INDIC DIGIT TWO
   06F3; CONTEXTO   # EXTENDED ARABIC-INDIC DIGIT THREE
   06F4; CONTEXTO   # EXTENDED ARABIC-INDIC DIGIT FOUR
   06F5; CONTEXTO   # EXTENDED ARABIC-INDIC DIGIT FIVE
   06F6; CONTEXTO   # EXTENDED ARABIC-INDIC DIGIT SIX
   06F7; CONTEXTO   # EXTENDED ARABIC-INDIC DIGIT SEVEN
   06F8; CONTEXTO   # EXTENDED ARABIC-INDIC DIGIT EIGHT
   06F9; CONTEXTO   # EXTENDED ARABIC-INDIC DIGIT NINE

   DISALLOWED -- Would otherwise have been PVALID

   0640; DISALLOWED # ARABIC TATWEEL
   07FA; DISALLOWED # NKO LAJANYALAN
   302E; DISALLOWED # HANGUL SINGLE DOT TONE MARK
   302F; DISALLOWED # HANGUL DOUBLE DOT TONE MARK
   3031; DISALLOWED # VERTICAL KANA REPEAT MARK
   3032; DISALLOWED # VERTICAL KANA REPEAT WITH VOICED SOUND MARK
   3033; DISALLOWED # VERTICAL KANA REPEAT MARK UPPER HALF
   3034; DISALLOWED # VERTICAL KANA REPEAT WITH VOICED SOUND MARK
                      UPPER HA
   3035; DISALLOWED # VERTICAL KANA REPEAT MARK LOWER HALF
   303B; DISALLOWED # VERTICAL IDEOGRAPHIC ITERATION MARK
```

7.7. BackwardCompatible (G)

Note: This category is defined in [RFC5892] and copied here for use in PRECIS. Because of how the PRECIS string classes are defined, only changes that would result in code points being added to or removed from the LetterDigits ("A") category would result in backward-incompatible modifications to code point assignments. Therefore, management of this category is handled via the processes specified in [RFC5892].

G: cp is in {}

This category includes the code points for which property values in versions of Unicode after 5.2 have changed in such a way that the derived property value would no longer be PVALID or DISALLOWED. If changes are made to future versions of Unicode so that code points might change property value from PVALID or DISALLOWED, then this table can be updated and keep special exception values so that the property values for code points stay stable.

7.8. JoinControl (H)

Note: This category is defined in [RFC5892] and copied here for use in PRECIS.

H: Join_Control(cp) = True

This category consists of Join Control characters (i.e., they are not in LetterDigits (Section 7.1) but are still required in strings under some circumstances).

7.9. OldHangulJamo (I)

Note: This category is defined in [RFC5892] and copied here for use in PRECIS.

I: Hangul_Syllable_Type(cp) is in {L, V, T}

This category consists of all conjoining Hangul Jamo (Leading Jamo, Vowel Jamo, and Trailing Jamo).

Elimination of conjoining Hangul Jamos from the set of PVALID characters results in restricting the set of Korean PVALID characters just to preformed, modern Hangul syllable characters. Old Hangul syllables, which are spelled with sequences of conjoining Hangul Jamos, are not PVALID for string classes.

7.10. Unassigned (J)

Note: This category is defined in [RFC5892] and copied here for use in PRECIS.

J: General_Category(cp) is in {Cn} and
Noncharacter_Code_Point(cp) = False

This category consists of code points in the Unicode character set that are not (yet) designated. Implementers might want to keep in mind that the Unicode Standard distinguishes between 'unassigned code points' and 'unassigned characters'. The unassigned code points are all but (Cn - Noncharacters), whereas the unassigned characters are all but (Cn + Cs).

7.11. ASCII7 (K)

This PRECIS-specific category consists of all printable, non-space characters from the 7-bit ASCII range. By applying this category, the algorithm specified under Section 8 exempts these characters from other rules that might be applied during PRECIS processing, on the assumption that these code points are in such wide use that disallowing them would be counter-productive.

K: cp is in {0021..007E}

7.12. Controls (L)

L: Control(cp) = True

7.13. PrecisIgnorableProperties (M)

This PRECIS-specific category is used to group code points that are discouraged from use in PRECIS string classes.

M: Default_Ignorable_Code_Point(cp) = True or
Noncharacter_Code_Point(cp) = True

The definition for Default_Ignorable_Code_Point can be found in the DerivedCoreProperties.txt [2] file, and at the time of Unicode 6.3 is as follows:

```
  Other_Default_Ignorable_Code_Point
+ Cf (Format characters)
+ Variation_Selector
- White_Space
- FFF9..FFFB (Annotation Characters)
- 0600..0604, 06DD, 070F, 110BD (exceptional Cf characters
```

that should be visible)

7.14. Spaces (N)

This PRECIS-specific category is used to group code points that are space characters.

N: General_Category(cp) is in {Zs}

7.15. Symbols (O)

This PRECIS-specific category is used to group code points that are symbols.

O: General_Category(cp) is in {Sm, Sc, Sk, So}

7.16. Punctuation (P)

This PRECIS-specific category is used to group code points that are punctuation characters.

P: General_Category(cp) is in {Pc, Pd, Ps, Pe, Pi, Pf, Po}

7.17. HasCompat (Q)

This PRECIS-specific category is used to group code points that have compatibility equivalents as explained in Chapter 2 and Chapter 3 of the Unicode Standard [UNICODE].

Q: toNFKC(cp) != cp

The toNFKC() operation returns the code point in normalization form KC. For more information, see Section 5 of Unicode Standard Annex #15 [UAX15].

7.18. OtherLetterDigits (R)

This PRECIS-specific category is used to group code points that are letters and digits other than the "traditional" letters and digits grouped under the LetterDigits (A) class (see Section 7.1).

R: General_Category(cp) is in {Lt, Nl, No, Me}

8. Calculation of the Derived Property

Possible values of the derived property are:

- PVALID
- ID_PVAL
- FREE_PVAL
- CONTEXTJ
- CONTEXTO
- DISALLOWED
- ID_DIS
- FREE_DIS
- UNASSIGNED

Note: The value of the derived property calculated can depend on the string class; for example, if an identifier used in an application protocol is defined as profiling the PRECIS IdentifierClass then a space character such as U+0020 would be assigned to ID_DIS, whereas if an identifier is defined as profiling the PRECIS FreeformClass then the character would be assigned to FREE_PVAL. For the sake of brevity, the designation "FREE_PVAL" is used in the code point tables, instead of the longer designation "ID_DIS or FREE_PVAL". In practice, the derived properties ID_PVAL and FREE_DIS are not used in this specification, since every ID_PVAL code point is PVALID and every FREE_DIS code point is DISALLOWED.

The algorithm to calculate the value of the derived property is as follows:

```
If .cp. .in. Exceptions Then Exceptions(cp);
Else If .cp. .in. BackwardCompatible Then BackwardCompatible(cp);
Else If .cp. .in. Unassigned Then UNASSIGNED;
Else If .cp. .in. ASCII7 Then PVALID;
Else If .cp. .in. JoinControl Then CONTEXTJ;
Else If .cp. .in. OldHangulJamo Then DISALLOWED;
Else If .cp. .in. PrecisIgnorableProperties Then DISALLOWED;
Else If .cp. .in. Controls Then DISALLOWED;
Else If .cp. .in. HasCompat Then ID_DIS or FREE_PVAL;
Else If .cp. .in. LetterDigits Then PVALID;
Else If .cp. .in. OtherLetterDigits Then ID_DIS or FREE_PVAL;
Else If .cp. .in. Spaces Then ID_DIS or FREE_PVAL;
Else If .cp. .in. Symbols Then ID_DIS or FREE_PVAL;
Else If .cp. .in. Punctuation Then ID_DIS or FREE_PVAL;
Else DISALLOWED;
```

Note: Use of the name of a rule (such as "Exceptions") implies the set of code points that the rule defines, whereas the same name as a function call (such as "Exceptions(cp)") implies the value that the code point has in the Exceptions table.

9. IANA Considerations

9.1. PRECIS Derived Property Value Registry

IANA is requested to create a PRECIS-specific registry with the Derived Properties for the versions of Unicode that are released after (and including) version 6.3. The derived property value is to be calculated in cooperation with a designated expert [RFC5226] according to the rules specified under Section 7 and Section 8, not by copying the non-normative table found under Appendix A.

The IESG is to be notified if backward-incompatible changes to the table of derived properties are discovered or if other problems arise during the process of creating the table of derived property values or during expert review. Changes to the rules defined under Section 7 and Section 8 require IETF Review.

9.2. PRECIS Base Classes Registry

IANA is requested to create a registry of PRECIS string classes. In accordance with [RFC5226], the registration policy is "RFC Required".

The registration template is as follows:

Base Class: [the name of the PRECIS string class]
Description: [a brief description of the PRECIS string class and its intended use, e.g., "A sequence of letters, numbers, and symbols that is used to identify or address a network entity."]
Specification: [the RFC number]

The initial registrations are as follows:

Base Class: FreeformClass.
Description: A sequence of letters, numbers, symbols, spaces, and other code points that is used for free-form strings.
Specification: RFC XXXX. [Note to RFC Editor: please change XXXX to the number issued for this specification.]

Base Class: IdentifierClass.
Description: A sequence of letters, numbers, and symbols that is used to identify or address a network entity.
Specification: RFC XXXX. [Note to RFC Editor: please change XXXX to the number issued for this specification.]

9.3. PRECIS Profiles Registry

IANA is requested to create a registry of profiles that use the PRECIS string classes. In accordance with [RFC5226], the registration policy is "Expert Review". This policy was chosen in order to ease the burden of registration while ensuring that "customers" of PRECIS receive appropriate guidance regarding the sometimes complex and subtle internationalization issues related to profiles of PRECIS string classes.

The registration template is as follows:

Name: [the name of the profile]
Applicability: [the specific protocol elements to which this profile applies, e.g., "Localparts in XMPP addresses."]
Base Class: [which PRECIS string class is being profiled]
Replaces: [the Stringprep profile that this PRECIS profile replaces, if any]
Width Mapping: [the behavioral rule for handling of width, e.g., "Map fullwidth and halfwidth characters to their compatibility variants."]
Additional Mappings: [any additional mappings are required or recommended, e.g., "Map non-ASCII space characters to ASCII space."]
Case Mapping: [the behavioral rule for handling of case, e.g., "Map uppercase and titlecase characters to lowercase."]
Normalization: [which Unicode normalization form is applied, e.g., "NFC"]
Directionality: [the behavioral rule for handling of right-to-left code points, e.g., "The 'Bidi Rule' defined in RFC 5893 applies."]
Exclusions: [a brief description of the specific code points or characters categories are excluded, e.g., "Eight legacy characters in the ASCII range" or "Any character that has a compatibility equivalent, i.e., the HasCompat category"]
Enforcement: [which entities enforce the rules, and when that enforcement occurs during protocol operations]
Specification: [a pointer to relevant documentation, such as an RFC or Internet-Draft]

In order to request a review, the registrant shall send a completed template to the precis@ietf.org list or its designated successor.

Factors to focus on while defining profiles and reviewing profile registrations include the following:

- Is the problem being addressed by this profile well-defined?

- Does the specification define what kinds of applications are involved and the protocol elements to which this profile applies?
- Would an existing PRECIS string class or profile solve the problem?
- Is the profile clearly defined?
- Does the profile reduce the degree to which human users could be surprised by application behavior (the "principle of least user surprise")?
- Is the profile based on an appropriate dividing line between user interface (culture, context, intent, locale, device limitations, etc.) and the use of conformant strings in protocol elements?
- Are the width mapping, case mapping, additional mapping, normalization, exclusion, and directionality rules appropriate for the intended use?
- Does the profile explain which entities enforce the rules, and when such enforcement occurs during protocol operations?
- Does the profile reduce the degree to which human users could be surprised or confused by application behavior (the "principle of least user surprise")?
- Does the profile introduce any new security concerns such as those described under Section 10 of this document (e.g., false positives for authentication or authorization)?

10. Security Considerations

10.1. General Issues

The security of applications that use this framework can depend in part on the proper preparation and comparison of internationalized strings. For example, such strings can be used to make authentication and authorization decisions, and the security of an application could be compromised if an entity providing a given string is connected to the wrong account or online resource based on different interpretations of the string.

Specifications of application protocols that use this framework are encouraged to describe how internationalized strings are used in the protocol, including the security implications of any false positives and false negatives that might result from various comparison operations. For some helpful guidelines, refer to [RFC6943], [RFC5890], [UTR36], and [UTS39].

10.2. Use of the IdentifierClass

Strings that conform to the IdentifierClass and any profile thereof are intended to be relatively safe for use in a broad range of applications, primarily because they include only letters, digits,

and "grandfathered" non-space characters from the ASCII range; thus they exclude spaces, characters with compatibility equivalents, and almost all symbols and punctuation marks. However, because such strings can still include so-called confusable characters (see Section 10.5), protocol designers and implementers are encouraged to pay close attention to the security considerations described elsewhere in this document.

10.3. Use of the FreeformClass

Strings that conform to the FreeformClass and many profiles thereof can include virtually any Unicode character. This makes the FreeformClass quite expressive, but also problematic from the perspective of possible user confusion. Protocol designers are hereby warned that the FreeformClass contains codepoints they might not understand, and are encouraged to profile the IdentifierClass wherever feasible; however, if an application protocol requires more code points than are allowed by the IdentifierClass, protocol designers are encouraged to define a profile of the FreeformClass that restricts the allowable code points as tightly as possible. (The PRECIS Working Group considered the option of allowing superclasses as well as profiles of PRECIS string classes, but decided against allowing superclasses to reduce the likelihood of security and interoperability problems.)

10.4. Local Character Set Issues

When systems use local character sets other than ASCII and Unicode, this specification leaves the problem of converting between the local character set and Unicode up to the application or local system. If different applications (or different versions of one application) implement different rules for conversions among coded character sets, they could interpret the same name differently and contact different application servers or other network entities. This problem is not solved by security protocols, such as Transport Layer Security (TLS) [RFC5246] and the Simple Authentication and Security Layer (SASL) [RFC4422], that do not take local character sets into account.

10.5. Visually Similar Characters

Some characters are visually similar and thus can cause confusion among humans. Such characters are often called "confusable characters" or "confusables".

The problem of confusable characters is not necessarily caused by the use of Unicode code points outside the ASCII range. For example, in some presentations and to some individuals the string "ju1iet" (spelled with DIGIT ONE, U+0031, as the third character) might appear

to be the same as "juliet" (spelled with LATIN SMALL LETTER L, U+006C), especially on casual visual inspection. This phenomenon is sometimes called "typejacking".

However, the problem is made more serious by introducing the full range of Unicode code points into protocol strings. For example, the characters U+13DA U+13A2 U+13B5 U+13AC U+13A2 U+13AC U+13D2 from the Cherokee block look similar to the ASCII characters "STPETER" as they might appear when presented using a "creative" font family.

In some examples of confusable characters, it is unlikely that the average human could tell the difference between the real string and the fake string. (Indeed, there is no programmatic way to distinguish with full certainty which is the fake string and which is the real string; in some contexts, the string formed of Cherokee characters might be the real string and the string formed of ASCII characters might be the fake string.) Because PRECIS-compliant strings can contain almost any properly-encoded Unicode code point, it can be relatively easy to fake or mimic some strings in systems that use the PRECIS framework. The fact that some strings are easily confused introduces security vulnerabilities of the kind that have also plagued the World Wide Web, specifically the phenomenon known as phishing.

Despite the fact that some specific suggestions about identification and handling of confusable characters appear in the Unicode Security Considerations [UTR36] and the Unicode Security Mechanisms [UTS39], it is also true (as noted in [RFC5890]) that "there are no comprehensive technical solutions to the problems of confusable characters". Because it is impossible to map visually similar characters without a great deal of context (such as knowing the font families used), the PRECIS framework does nothing to map similar-looking characters together, nor does it prohibit some characters because they look like others.

Nevertheless, specifications for application protocols that use this framework MUST describe how confusable characters can be abused to compromise the security of systems that use the protocol in question, along with any protocol-specific suggestions for overcoming those threats. In particular, software implementations and service deployments that use PRECIS-based technologies are strongly encouraged to define and implement consistent policies regarding the registration, storage, and presentation of visually similar characters. The following recommendations are appropriate:

1. An application service SHOULD define a policy that specifies the scripts or blocks of characters that the service will allow to be registered (e.g., in an account name) or stored (e.g., in a file

name). Such a policy SHOULD be informed by the languages and scripts that are used to write registered account names; in particular, to reduce confusion, the service SHOULD forbid registration or storage of strings that contain characters from more than one script and SHOULD restrict registrations to characters drawn from a very small number of scripts (e.g., scripts that are well-understood by the administrators of the service, to improve manageability).

2. User-oriented application software SHOULD define a policy that specifies how internationalized strings will be presented to a human user. Because every human user of such software has a preferred language or a small set of preferred languages, the software SHOULD gather that information either explicitly from the user or implicitly via the operating system of the user's device. Furthermore, because most languages are typically represented by a single script or a small set of scripts, and because most scripts are typically contained in one or more blocks of characters, the software SHOULD warn the user when presenting a string that mixes characters from more than one script or block, or that uses characters outside the normal range of the user's preferred language(s). (Such a recommendation is not intended to discourage communication across different communities of language users; instead, it recognizes the existence of such communities and encourages due caution when presenting unfamiliar scripts or characters to human users.)

The challenges inherent in supporting the full range of Unicode code points have in the past led some to hope for a way to programmatically negotiate more restrictive ranges based on locale, script, or other relevant factors, to tag the locale associated with a particular string, etc. As a general-purpose internationalization technology, the PRECIS framework does not include such mechanisms.

10.6. Security of Passwords

Two goals of passwords are to maximize the amount of entropy and to minimize the potential for false positives. These goals can be achieved in part by allowing a wide range of code points and by ensuring that passwords are handled in such a way that code points are not compared aggressively. Therefore, it is NOT RECOMMENDED for application protocols to profile the FreeformClass for use in passwords in a way that removes entire categories (e.g., by disallowing symbols or punctuation). Furthermore, it is NOT RECOMMENDED for application protocols to map uppercase and titlecase code points to their lowercase equivalents in such strings; instead, it is RECOMMENDED to preserve the case of all code points contained in such strings and to compare them in a case-sensitive manner.

That said, software implementers need to be aware that there exist tradeoffs between entropy and usability. For example, allowing a user to establish a password containing "uncommon" code points might make it difficult for the user to access a service when using an unfamiliar or constrained input device.

Some application protocols use passwords directly, whereas others reuse technologies that themselves process passwords (one example of such a technology is the Simple Authentication and Security Layer [RFC4422]). Moreover, passwords are often carried by a sequence of protocols with backend authentication systems or data storage systems such as RADIUS [RFC2865] and LDAP [RFC4510]. Developers of application protocols are encouraged to look into reusing these profiles instead of defining new ones, so that end-user expectations about passwords are consistent no matter which application protocol is used.

11. Interoperability Considerations

Although strings that are consumed in PRECIS-based application protocols are often encoded using UTF-8 [RFC3629], the exact encoding is a matter for the application protocol that uses PRECIS, not for the PRECIS framework.

It is known that some existing systems are unable to support the full Unicode character set, or even any characters outside the ASCII range. If two (or more) applications need to interoperate when exchanging data (e.g., for the purpose of authenticating a username or password), they will naturally need to have in common at least one coded character set (as defined by [RFC6365]). Establishing such a baseline is a matter for the application protocol that uses PRECIS, not for the PRECIS framework.

The PRECIS framework, which is defined in terms of the latest version of Unicode as of the time of this writing (6.3), treats the character U+19DA NEW TAI LUE THAM as DISALLOWED. Implementers need to be aware that this treatment is different from IDNA2008 (originally defined in terms of Unicode 5.2), which treats U+19DA as PVALID.

12. References

12.1. Normative References

[I-D.ietf-precis-mappings]
Yoneya, Y. and T. NEMOTO, "Mapping characters for PRECIS classes", draft-ietf-precis-mappings-05 (work in

progress), October 2013.

[RFC20] Cerf, V., "ASCII format for network interchange", RFC 20, October 1969.

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

[RFC5198] Klensin, J. and M. Padlipsky, "Unicode Format for Network Interchange", RFC 5198, March 2008.

[UNICODE] The Unicode Consortium, "The Unicode Standard", 2013, <http://www.unicode.org/versions/latest/>.

12.2. Informative References

[I-D.ietf-precis-nickname]
Saint-Andre, P., "Preparation and Comparison of Nicknames", draft-ietf-precis-nickname-08 (work in progress), December 2013.

[I-D.ietf-precis-saslprepbis]
Saint-Andre, P. and A. Melnikov, "Username and Password Preparation Algorithms", draft-ietf-precis-saslprepbis-06 (work in progress), December 2013.

[I-D.ietf-xmpp-6122bis]
Saint-Andre, P., "Extensible Messaging and Presence Protocol (XMPP): Address Format", draft-ietf-xmpp-6122bis-09 (work in progress), November 2013.

[RFC2865] Rigney, C., Willens, S., Rubens, A., and W. Simpson, "Remote Authentication Dial In User Service (RADIUS)", RFC 2865, June 2000.

[RFC3454] Hoffman, P. and M. Blanchet, "Preparation of Internationalized Strings ("stringprep")", RFC 3454, December 2002.

[RFC3490] Faltstrom, P., Hoffman, P., and A. Costello, "Internationalizing Domain Names in Applications (IDNA)", RFC 3490, March 2003.

[RFC3491] Hoffman, P. and M. Blanchet, "Nameprep: A Stringprep Profile for Internationalized Domain Names (IDN)", RFC 3491, March 2003.

[RFC3629] Yergeau, F., "UTF-8, a transformation format of ISO 10646", STD 63, RFC 3629, November 2003.

[RFC4422] Melnikov, A. and K. Zeilenga, "Simple Authentication and Security Layer (SASL)", RFC 4422, June 2006.

[RFC4510] Zeilenga, K., "Lightweight Directory Access Protocol (LDAP): Technical Specification Road Map", RFC 4510, June 2006.

[RFC4690] Klensin, J., Faltstrom, P., Karp, C., and IAB, "Review and Recommendations for Internationalized Domain Names (IDNs)", RFC 4690, September 2006.

[RFC5226] Narten, T. and H. Alvestrand, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 5226, May 2008.

[RFC5234] Crocker, D. and P. Overell, "Augmented BNF for Syntax Specifications: ABNF", STD 68, RFC 5234, January 2008.

[RFC5246] Dierks, T. and E. Rescorla, "The Transport Layer Security (TLS) Protocol Version 1.2", RFC 5246, August 2008.

[RFC5890] Klensin, J., "Internationalized Domain Names for Applications (IDNA): Definitions and Document Framework", RFC 5890, August 2010.

[RFC5891] Klensin, J., "Internationalized Domain Names in Applications (IDNA): Protocol", RFC 5891, August 2010.

[RFC5892] Faltstrom, P., "The Unicode Code Points and Internationalized Domain Names for Applications (IDNA)", RFC 5892, August 2010.

[RFC5893] Alvestrand, H. and C. Karp, "Right-to-Left Scripts for Internationalized Domain Names for Applications (IDNA)", RFC 5893, August 2010.

[RFC5894] Klensin, J., "Internationalized Domain Names for Applications (IDNA): Background, Explanation, and Rationale", RFC 5894, August 2010.

[RFC5895] Resnick, P. and P. Hoffman, "Mapping Characters for Internationalized Domain Names in Applications (IDNA) 2008", RFC 5895, September 2010.

[RFC6365] Hoffman, P. and J. Klensin, "Terminology Used in

Internationalization in the IETF", BCP 166, RFC 6365, September 2011.

[RFC6885] Blanchet, M. and A. Sullivan, "Stringprep Revision and Problem Statement for the Preparation and Comparison of Internationalized Strings (PRECIS)", RFC 6885, March 2013.

[RFC6943] Thaler, D., "Issues in Identifier Comparison for Security Purposes", RFC 6943, May 2013.

[UAX9] The Unicode Consortium, "Unicode Standard Annex #9: Unicode Bidirectional Algorithm", September 2012, <http://unicode.org/reports/tr9/>.

[UAX11] The Unicode Consortium, "Unicode Standard Annex #11: East Asian Width", September 2012, <http://unicode.org/reports/tr11/>.

[UAX15] The Unicode Consortium, "Unicode Standard Annex #15: Unicode Normalization Forms", August 2012, <http://unicode.org/reports/tr15/>.

[UTR36] The Unicode Consortium, "Unicode Technical Report #36: Unicode Security Considerations", July 2012, <http://unicode.org/reports/tr36/>.

[UTS39] The Unicode Consortium, "Unicode Technical Standard #39: Unicode Security Mechanisms", July 2012, <http://unicode.org/reports/tr39/>.

URIs

[1] <http://unicode.org/Public/UNIDATA/PropertyAliases.txt>

[2] <http://unicode.org/Public/UNIDATA/DerivedCoreProperties.txt>

Appendix A. Codepoint Table

If one applies the property calculation rules from Section 8 to the code points 0x0000 to 0x10FFFF in Unicode 6.3, the result is as shown in the following table, in Unicode Character Database (UCD) format. The columns of the table are as follows:

1. The code point or codepoint range.
2. The assignment for the code point or range, where the value is one of PVALID, DISALLOWED, UNASSIGNED, CONTEXTO, CONTEXTJ, or FREE_PVAL (where the latter includes ID_DIS).

3. The name or names for the code point or range.

This table is non-normative, is included only for illustrative purposes, and applies only to Unicode 6.3, not to past or future versions of Unicode. Please note that the strings displayed in the third column are not necessarily the formal name of the code point (as defined in [UNICODE]) because the fixed width of the RFC format necessitated truncation of many names.

```
0000..001F  ; DISALLOWED  # <control>
0020        ; FREE_PVAL   # SPACE
0021..007E  ; PVALID      # EXCLAM MARK..TILDE
007F..009F  ; DISALLOWED  # <control>
00A0..00AC  ; FREE_PVAL   # NO-BREAK SPACE..NOT SIGN
00AD        ; DISALLOWED  # SOFT HYPH
00AE..00B6  ; FREE_PVAL   # REGISTERED SIGN..PILCROW SIGN
00B7        ; CONTEXTO    # MIDDLE DOT
00B8..00BF  ; FREE_PVAL   # CEDILLA..INV QUEST IND
00C0..00D6  ; PVALID      # LAT CAP LET A W GRAV..LAT CAP O
00D7        ; FREE_PVAL   # MULTIPLICATION SIGN
00D8..00F6  ; PVALID      # LAT CAP LET O W STROKE..LAT SM
00F7        ; FREE_PVAL   # DIVISION SIGN
00F8..0131  ; PVALID      # LAT SM LET O W STROKE..LAT SM LET
0132..0133  ; FREE_PVAL   # LAT CAP LIG IJ..LAT SM LIB IJ
0134..013E  ; PVALID      # LAT CAP LET J W CIRCUM..LAT SM LET
013F..0140  ; FREE_PVAL   # LAT CAP LET L W MID DOT..LAT SM LET
0141..0148  ; PVALID      # LAT CAP LET L W STROKE..LAT SM LET
0149        ; FREE_PVAL   # LAT SM LET N PRECEDED BY APOS
014A..017E  ; PVALID      # LAT CAP LET ENG..LAT SM LET Z W CA
017F        ; FREE_PVAL   # LAT SM LET LONG S
0180..01C3  ; PVALID      # LAT SM LET B W STROKE..LAT LET RETR
01C4..01CC  ; FREE_PVAL   # LAT CAP LET DZ W CARON..LAT SM
01CD..01F0  ; PVALID      # LAT CAP LET A W CARON..LAT SM LET J
01F1..01F3  ; FREE_PVAL   # LAT CAP LET DZ..LAT SM LET DZ
01F4..02AF  ; PVALID      # LAT CAP LET G W ACUTE..LAT SM
02B0..02B8  ; FREE_PVAL   # MOD LET SM H..MOD LET SM Y
02B9..02C1  ; PVALID      # MOD LET PRIME..MOD LET REV GLOT ST
02C2..02C5  ; FREE_PVAL   # MOD LET L ARROW..MOD LET D ARROW
02C6..02D1  ; PVALID      # MOD LET CIRCUM ACC..MOD LET HALF TR
02D2..02EB  ; FREE_PVAL   # MOD LET CENT R HALF RING..MOD LET Y
02EC        ; PVALID      # MOD LET VOICING
02ED        ; FREE_PVAL   # MOD LET UNASPIRATED
02EE        ; PVALID      # MOD LET DOUBLE APOS
02EF..02FF  ; FREE_PVAL   # MOD LET LOW D ARR..MOD LET LOW L AR
0300..034E  ; PVALID      # COMB GRAVE ACCENT..COMB UP ARROW BE
034F        ; DISALLOWED  # COMB GRAPHEME JOINER
0350..0374  ; PVALID      # COMB RIGHT ARROWHEAD..GREEK NUM SIG
0375        ; CONTEXTO    # GREEK LOW NUM SIGN
```

```
   0376..0377  ; PVALID      # GR CAP LET PAMPHYLIAN DIGAMMA..GR S
   0378..0379  ; UNASSIGNED  # <reserved>..<reserved>
   037A        ; FREE_PVAL   # GR YPOGEGRAMMENI..GR SM REV DOT LUN
   037B..037D  ; PVALID      # GR SM REV LUN SIG..GR SM REV DOT LU
   037E        ; FREE_PVAL   # GREEK QUEST MARK
   037F..0383  ; UNASSIGNED  # <reserved>..<reserved>
   0384..0385  ; FREE_PVAL   # GREEK TONOS..GREEK DIALYTIKA TONOS
   0386        ; PVALID      # GR CAP LET ALPHA W TONOS
   0387        ; FREE_PVAL   # GREEK ANO TELEIA
   0388..038A  ; PVALID      # GR CAP LET EPSILON W TONOS..GR CAP
   038B        ; UNASSIGNED  # <reserved>
   038C        ; PVALID      # GREEK CAP LET OMICRON W TONOS
   038D        ; UNASSIGNED  # <reserved>
   038E..03A1  ; PVALID      # GR CAP LET EPSILON W TONOS..GR CAP
   03A2        ; UNASSIGNED  # <reserved>
   03A3..03CF  ; PVALID      # GREEK CAP LET SIGMA..GR CAP
   03D0..03D2  ; FREE_PVAL   # GR BETA SYM..GR UPSILON W HOOK
   03D3..03D4  ; FREE_PVAL   # GR UPSILON W ACUTE AND HOOK..GR UP
   03D5..03D6  ; FREE_PVAL   # GR PHI SYM..GR PI SYM
   03D7..03EF  ; PVALID      # GR KAI SYM..COPT SM LET DEI
   03F0..03F2  ; FREE_PVAL   # GR KAPPA SYM..GR LUNATE SIGMA
   03F3        ; PVALID      # GREEK LET YOT
   03F4..03F6  ; FREE_PVAL   # GR CAP THETA..GR REV LUNATE EPSILON
   03F7..03F8  ; PVALID      # GR CAP LET SHO..GR SM LET SHO
   03F9        ; FREE_PVAL   # GREEK CAP LUNATE SIGMA SYM
   03FA..0481  ; PVALID      # GR CAP LET SAN..CYR SML LET KOPPA
   0482        ; FREE_PVAL   # CYR THOUSANDS SIGN
   0483..0487  ; PVALID      # COMB CYR TITLO..COMB CYR POK
   0488..0489  ; FREE_PVAL   # COMB CYR HUNDRED THOUSANDS SIGN..C
   048A..0527  ; PVALID      # CYR CAP LET SH I W TAIL..CYR S
   0528..0530  ; UNASSIGNED  # <reserved>..<reserved>
   0531..0556  ; PVALID      # ARM CAP LET AYB..ARM CAP LET FEH
   0557..0558  ; UNASSIGNED  # <reserved>..<reserved>
   0559        ; PVALID      # ARM MOD LET LEFT HALF RING
   055A..055F  ; FREE_PVAL   # ARM APOS..ARM ABBREV
   0560        ; UNASSIGNED  # <reserved>
   0561..0586  ; PVALID      # ARM SM LET AYB..ARMENIAN SM LE
   0587        ; FREE_PVAL   # ARM SM LIG ECH YIWN
   0588        ; UNASSIGNED  # <reserved>
   0589..058A  ; FREE_PVAL   # ARMENIAN FULL STOP..ARMENIAN HYPH
   058B..058E  ; UNASSIGNED  # <reserved>..<reserved>
   058F        ; FREE_PVAL   # ARMENIAN DRAM SIGN
   0590        ; UNASSIGNED  # <reserved>
   0591..05BD  ; PVALID      # HEBR ACC ETNAHTA..HEBR PNT ME
   05BE        ; FREE_PVAL   # HEBR PUNCT MAQAF
   05BF        ; PVALID      # HEBR PNT RAFE
   05C0        ; FREE_PVAL   # HEBR PUNCT PASEQ
   05C1..05C2  ; PVALID      # HEBR PNT SHIN DOT..HEBR PNT SIN DOT
```

```
   05C3        ; FREE_PVAL   # HEBR PUNCT SOF PASUQ
   05C4..05C5  ; PVALID      # HEBR MARK UP DOT..HEBR MARK LOW DOT
   05C6        ; FREE_PVAL   # HEBR PUNCT NUN HAFUKHA
   05C7        ; PVALID      # HEBR PNT QAMATS QATAN
   05C8..05CF  ; UNASSIGNED  # <reserved>..<reserved>
   05D0..05EA  ; PVALID      # HEBR LET ALEF..HEBR LET TAV
   05EB..05EF  ; UNASSIGNED  # <reserved>..<reserved>
   05F0..05F2  ; PVALID      # HEBR LIG YIDDISH DOUBLE VAV..HEBR L
   05F3..05F4  ; CONTEXTO    # HEBR PUNCT GERESH..HEBR PUNCTUATIO
   05F5..05FF  ; UNASSIGNED  # <reserved>..<reserved>
   0600..0604  ; DISALLOWED  # ARAB NUM SIGN..ARAB SIGN SAM
   0605        ; UNASSIGNED  # <reserved>..<reserved>
   0606..060F  ; FREE_PVAL   # AR-IND CUBE ROOT..ARAB SIGN MISRA
   0610..061A  ; PVALID      # ARAB SIGN SALLALLAHOU ALAYHE ..AR
   061B        ; FREE_PVAL   # ARAB SEMICOLON
   061C        ; DISALLOWED  # ARAB LET MARK
   061D..061D  ; UNASSIGNED  # <reserved>..<reserved>
   061E..061F  ; FREE_PVAL   # ARAB TRIPLE DOT PUNCT MARK..ARAB Q
   0620..063F  ; PVALID      # ARAB LET KASH..ARAB LET FARSI YEH
   0640        ; DISALLOWED  # ARAB TATWEEL
   0641..065F  ; PVALID      # ARAB LET FEH..ARAB WAVY HAMZA BEL
   0660..0669  ; CONTEXTO    # AR-IND DIG ZERO..AR-IND DIG
   066A..066D  ; FREE_PVAL   # ARAB PCT SIGN..ARAB FIVE PNTED STA
   066E..0674  ; PVALID      # ARAB LET DOTLESS BEH..ARAB LET HIG
   0675..0678  ; FREE_PVAL   # ARAB LET HIGH HAMZA ALEF..ARAB LET
   0679..06D3  ; PVALID      # ARAB LET TTEH..ARAB LET YEH BARREE
   06D4        ; FREE_PVAL   # ARAB FULL STOP
   06D5..06DC  ; PVALID      # ARAB LET AE..ARAB SM HIGH SEEN
   06DD        ; DISALLOWED  # ARAB END OF AYAH
   06DE        ; FREE_PVAL   # ARAB START OF RUB EL HIZB
   06DF..06E8  ; PVALID      # ARAB SM HIGH ROUNDED ZERO..ARAB SM
   06E9        ; FREE_PVAL   # ARAB PLACE OF SAJDAH
   06EA..06EF  ; PVALID      # ARAB EMPTY CENTRE LOW STOP..ARAB LET
   06F0..06F9  ; CONTEXTO    # EXT AR-IND DIG ZERO..EXT A
   06FA..06FF  ; PVALID      # ARAB LET SHEEN W DOT BEL..ARAB
   0700..070D  ; FREE_PVAL   # SYR END OF PARA..SYR HARKLEAN AST
   070E        ; UNASSIGNED  # <reserved>
   070F        ; DISALLOWED  # SYR ABBR MARK
   0710..074A  ; PVALID      # SYR LET ALAPH..SYR BARREKH
   074B..074C  ; UNASSIGNED  # <reserved>..<reserved>
   074D..07B1  ; PVALID      # SYR LET SOGDIAN ZHAIN..THAANA LET N
   07B2..07BF  ; UNASSIGNED  # <reserved>..<reserved>
   07C0..07F5  ; PVALID      # NKO DIG ZERO..NKO LOW TONE APOS
   07F6..07F9  ; FREE_PVAL   # NKO SYM OO DENNEN..NKO EXCLAMATI
   07FA        ; DISALLOWED  # NKO LAJANYALAN
   07FB..07FF  ; UNASSIGNED  # <reserved>..<reserved>
   0800..082D  ; PVALID      # SAMAR LET ALAF..SAMAR MARK NEQUDA
   082E..082F  ; UNASSIGNED  # <reserved>..<reserved>
```

```
0830..083E  ; FREE_PVAL   # SAMAR PUNCT NEQUDAA..SAMAR PUN
083F        ; UNASSIGNED  # <reserved>
0840..085B  ; PVALID      # MANDAIC LET HALQA..MANDAIC GEM
085C..085D  ; UNASSIGNED  # <reserved>..<reserved>
085E        ; FREE_PVAL   # MANDAIC PUNCTUATION
085F..089F  ; UNASSIGNED  # <reserved>..<reserved>
08A0        ; PVALID      # ARAB LET BEH W SM V BEL
08A1        ; UNASSIGNED  # <reserved>
08A2..08AC  ; PVALID      # ARAB LET JEEM W 2 DOTS AB..ARAB
08AD..08E3  ; UNASSIGNED  # <reserved>..<reserved>
08E4..08FE  ; PVALID      # ARAB CURLY FATHA..ARAB DAMMA W
08FF        ; UNASSIGNED  # <reserved>
0900..0963  ; PVALID      # DEVAN SIGN INV CANDRABINDU..DEVAN V
0964..0965  ; FREE_PVAL   # DEVAN DANDA..DEVAN DOUBLE DANDA
0966..096F  ; PVALID      # DEVAN DIG ZERO..DEVAN DIG NINE
0970        ; FREE_PVAL   # DEVAN ABBR SIGN
0971..0977  ; PVALID      # DEVAN SIGN HIGH SPACING DOT..DEVAN
0978        ; UNASSIGNED  # <reserved>
0979..097F  ; PVALID      # DEVAN SIGN HIGH SPACING DOT..DEVAN
0980        ; UNASSIGNED  # <reserved>
0981..0983  ; PVALID      # BENG SIGN CANDRABINDU..BENG SIGN VIS
0984        ; UNASSIGNED  # <reserved>
0985..098C  ; PVALID      # BENG LET A..BENG LET VOC L
098D..098E  ; UNASSIGNED  # <reserved>..<reserved>
098F..0990  ; PVALID      # BENG LET E..BENG LET AI
0991..0992  ; UNASSIGNED  # <reserved>..<reserved>
0993..09A8  ; PVALID      # BENG LET O..BENG LET NA
09A9        ; UNASSIGNED  # <reserved>
09AA..09B0  ; PVALID      # BENG LET PA..BENG LET RA
09B1        ; UNASSIGNED  # <reserved>
09B2        ; PVALID      # BENG LET LA
09B3..09B5  ; UNASSIGNED  # <reserved>..<reserved>
09B6..09B9  ; PVALID      # BENG LET SHA..BENG LET HA
09BA..09BB  ; UNASSIGNED  # <reserved>..<reserved>
09BC..09C4  ; PVALID      # BENG SIGN NUKTA..BENG VOW SIGN VOCAL
09C5..09C6  ; UNASSIGNED  # <reserved>..<reserved>
09C7..09C8  ; PVALID      # BENG VOW SIGN E..BENG VOW SIGN AI
09C9..09CA  ; UNASSIGNED  # <reserved>..<reserved>
09CB..09CE  ; PVALID      # BENG VOW SIGN O..BENG LET KHANDA
09CF..09D6  ; UNASSIGNED  # <reserved>..<reserved>
09D7        ; PVALID      # BENG AU LEN MARK
09D8..09DB  ; UNASSIGNED  # <reserved>..<reserved>
09DC..09DD  ; PVALID      # BENG LET RRA..BENG LET RHA
09DE        ; UNASSIGNED  # <reserved>
09DF..09E3  ; PVALID      # BENG LET YYA..BENG VOW SIG
09E4..09E5  ; UNASSIGNED  # <reserved>..<reserved>
09E6..09F1  ; PVALID      # BENG DIG ZERO..BENG LET RA W L
09F2..09FB  ; FREE_PVAL   # BENG RUPEE MARK..BENG GANDA MARK
```

```
   09FC..0A00  ; UNASSIGNED  # <reserved>..<reserved>
   0A01..0A03  ; PVALID      # GURMUKHI SIGN ADAK BINDI..GURMUKHI
   0A04        ; UNASSIGNED  # <reserved>
   0A05..0A0A  ; PVALID      # GURMUKHI LET A..GURMUKHI LET UU
   0A0B..0A0E  ; UNASSIGNED  # <reserved>..<reserved>
   0A0F..0A10  ; PVALID      # GURMUKHI LET EE..GURMUKHI LET AI
   0A11..0A12  ; UNASSIGNED  # <reserved>..<reserved>
   0A13..0A28  ; PVALID      # GURMUKHI LET OO..GURMUKHI LET NA
   0A29        ; UNASSIGNED  # <reserved>
   0A2A..0A30  ; PVALID      # GURMUKHI LET PA..GURMUKHI LET RA
   0A31        ; UNASSIGNED  # <reserved>
   0A32..0A33  ; PVALID      # GURMUKHI LET LA..GURMUKHI LET LLA
   0A34        ; UNASSIGNED  # <reserved>
   0A35.OA36   ; PVALID      # GURMUKHI LET VA..GURMUKHI LET SHA
   0A37        ; UNASSIGNED  # <reserved>
   0A38..0A39  ; PVALID      # GURMUKHI LET SA..GURMUKHI LET HA
   0A3A..0A3B  ; UNASSIGNED  # <reserved>..<reserved>
   0A3C        ; PVALID      # GURMUKHI SIGN NUKTA
   0A3D        ; UNASSIGNED  # <reserved>
   0A3E..0A42  ; PVALID      # GURMUKHI VOW SIGN AA..GURMUKHI V
   0A43..0A46  ; UNASSIGNED  # <reserved>..<reserved>
   0A47..0A48  ; PVALID      # GURMUKHI VOW SIGN EE..GURMUKHI V
   0A49..0A4A  ; UNASSIGNED  # <reserved>..<reserved>
   0A4B..0A4D  ; PVALID      # GURMUKHI VOW SIGN OO..GURMUKHI S
   0A4E..0A50  ; UNASSIGNED  # <reserved>..<reserved>
   0A51        ; PVALID      # GURMUKHI SIGN UDAAT
   0A52..0A58  ; UNASSIGNED  # <reserved>..<reserved>
   0A59..0A5C  ; PVALID      # GURMUKHI LET KHHA..GURMUKHI LET RRA
   0A5D        ; UNASSIGNED  # <reserved>
   0A5E        ; PVALID      # GURMUKHI LET FA
   0A5F..0A65  ; UNASSIGNED  # <reserved>..<reserved>
   0A66..0A75  ; PVALID      # GURMUKHI DIG ZERO..GURMUKHI SIGN YA
   0A76..0A80  ; UNASSIGNED  # <reserved>..<reserved>
   0A81..0A83  ; PVALID      # GUJARATI SIGN CANDRABINDU..GUJARATI
   0A84        ; UNASSIGNED  # <reserved>
   0A85..0A8D  ; PVALID      # GUJARATI LET A..GUJARATI VOW CAND
   0A8E        ; UNASSIGNED  # <reserved>
   0A8F..0A91  ; PVALID      # GUJARATI LET E..GUJARATI VOW CAND
   0A92        ; UNASSIGNED  # <reserved>
   0A93..0AA8  ; PVALID      # GUJARATI LET O..GUJARATI LET NA
   0AA9        ; UNASSIGNED  # <reserved>
   0AAA..0AB0  ; PVALID      # GUJARATI LET PA..GUJARATI LET RA
   0AB1        ; UNASSIGNED  # <reserved>
   0AB2..0AB3  ; PVALID      # GUJARATI LET LA..GUJARATI LET LLA
   0AB4        ; UNASSIGNED  # <reserved>
   0AB5..0AB9  ; PVALID      # GUJARATI LET VA..GUJARATI LET HA
   0ABA..0ABB  ; UNASSIGNED  # <reserved>..<reserved>
   0ABC..0AC5  ; PVALID      # GUJARATI SIGN NUKTA..GUJARATI VOW
```

```
   0AC6        ; UNASSIGNED  # <reserved>
   0AC7..0AC9  ; PVALID      # GUJARATI VOW SIGN E..GUJARATI VOW
   0ACA        ; UNASSIGNED  # <reserved>
   0ACB..0ACD  ; PVALID      # GUJARATI VOW SIGN O..GUJARATI SIG
   0ACE..0ACF  ; UNASSIGNED  # <reserved>..<reserved>
   0AD0        ; PVALID      # GUJARATI OM
   0AD1..0ADF  ; UNASSIGNED  # <reserved>..<reserved>
   0AE0..0AE3  ; PVALID      # GUJARATI LET VOC RR..GUJARATI V
   0AE4..0AE5  ; UNASSIGNED  # <reserved>..<reserved>
   0AE6..0AEF  ; PVALID      # GUJARATI DIG ZERO..GUJARATI DIG NINE
   0AF0..0AF1  ; FREE_PVAL   # GUJARATI ABBR SIGN..GUJARATI RUPEE S
   0AF2..0B00  ; UNASSIGNED  # <reserved>..<reserved>
   0B01..0B03  ; PVALID      # ORIYA SIGN CANDRABINDU..ORIYA SIGN V
   0B04        ; UNASSIGNED  # <reserved>
   0B05..0B0C  ; PVALID      # ORIYA LET A..ORIYA LET VOC L
   0B0D..0B0E  ; UNASSIGNED  # <reserved>..<reserved>
   0B0F..0B10  ; PVALID      # ORIYA LET E..ORIYA LET AI
   0B11..0B12  ; UNASSIGNED  # <reserved>..<reserved>
   0B13..0B28  ; PVALID      # ORIYA LET O..ORIYA LET NA
   0B29        ; UNASSIGNED  # <reserved>
   0B2A..0B30  ; PVALID      # ORIYA LET PA..ORIYA LET RA
   0B31        ; UNASSIGNED  # <reserved>
   0B32..0B33  ; PVALID      # ORIYA LET LA..ORIYA LET LLA
   0B34        ; UNASSIGNED  # <reserved>
   0B35..0B39  ; PVALID      # ORIYA LET VA..ORIYA LET HA
   0B3A..0B3B  ; UNASSIGNED  # <reserved>..<reserved>
   0B3C..0B44  ; PVALID      # ORIYA SIGN NUKTA..ORIYA VOW SIGN
   0B45..0B46  ; UNASSIGNED  # <reserved>..<reserved>
   0B47..0B48  ; PVALID      # ORIYA VOW SIGN E..ORIYA VOW SIG
   0B49..0B4A  ; UNASSIGNED  # <reserved>..<reserved>
   0B4B..0B4D  ; PVALID      # ORIYA VOW SIGN O..ORIYA SIGN VIRA
   0B4E..0B55  ; UNASSIGNED  # <reserved>..<reserved>
   0B56..0B57  ; PVALID      # ORIYA AI LEN MARK..ORIYA AU LENG
   0B58..0B5B  ; UNASSIGNED  # <reserved>..<reserved>
   0B5C..0B5D  ; PVALID      # ORIYA LET RRA..ORIYA LET RHA
   0B5E        ; UNASSIGNED  # <reserved>
   0B5F..0B63  ; PVALID      # ORIYA LET YYA..ORIYA VOW SIGN VOCA
   0B64..0B65  ; UNASSIGNED  # <reserved>..<reserved>
   0B66..0B6F  ; PVALID      # ORIYA DIG ZERO..ORIYA DIG NINE
   0B70        ; FREE_PVAL   # ORIYA ISSHAR
   0B71        ; PVALID      # ORIYA LET WA
   0B72..0B77  ; FREE_PVAL   # ORIYA FRACT ONE QUART..ORIYA FRACT
   0B78..0B81  ; UNASSIGNED  # <reserved>..<reserved>
   0B82..0B83  ; PVALID      # TAMIL SIGN ANUSVARA..TAMIL SIGN VIS
   0B84        ; UNASSIGNED  # <reserved>
   0B85..0B8A  ; PVALID      # TAMIL LET A..TAMIL LET UU
   0B8B..0B8D  ; UNASSIGNED  # <reserved>..<reserved>
   0B8E..0B90  ; PVALID      # TAMIL LET E..TAMIL LET AI
```

```
0B91        ; UNASSIGNED  # <reserved>
0B92..0B95  ; PVALID      # TAMIL LET O..TAMIL LET KA
0B96..0B98  ; UNASSIGNED  # <reserved>..<reserved>
0B99..0B9A  ; PVALID      # TAMIL LET NGA..TAMIL LET CA
0B9B        ; UNASSIGNED  # <reserved>
0B9C        ; PVALID      # TAMIL LET JA
0B9D        ; UNASSIGNED  # <reserved>
0B9E..0B9F  ; PVALID      # TAMIL LET NYA..TAMIL LET TTA
0BA0..0BA2  ; UNASSIGNED  # <reserved>..<reserved>
0BA3..0BA4  ; PVALID      # TAMIL LET NNA..TAMIL LET TA
0BA5..0BA7  ; UNASSIGNED  # <reserved>..<reserved>
0BA8..0BAA  ; PVALID      # TAMIL LET NA..TAMIL LET PA
0BAB..0BAD  ; UNASSIGNED  # <reserved>..<reserved>
0BAE..0BB9  ; PVALID      # TAMIL LET MA..TAMIL LET HA
0BBA..0BBD  ; UNASSIGNED  # <reserved>..<reserved>
0BBE..0BC2  ; PVALID      # TAMIL VOW SIGN AA..TAMIL VOW SI
0BC3..0BC5  ; UNASSIGNED  # <reserved>..<reserved>
0BC6..0BC8  ; PVALID      # TAMIL VOW SIGN E..TAMIL VOW SIG
0BC9        ; UNASSIGNED  # <reserved>
0BCA..0BCD  ; PVALID      # TAMIL VOW SIGN O..TAMIL SIGN VIRA
0BCE..0BCF  ; UNASSIGNED  # <reserved>..<reserved>
0BD0        ; PVALID      # TAMIL OM
0BD1..0BD6  ; UNASSIGNED  # <reserved>..<reserved>
0BD7        ; PVALID      # TAMIL AU LEN MARK
0BD8..0BE5  ; UNASSIGNED  # <reserved>..<reserved>
0BE6..0BEF  ; PVALID      # TAMIL DIG ZERO..TAMIL DIG NINE
0BF0..0BFA  ; FREE_PVAL   # TAMIL NUM TEN..TAMIL NUM SIGN
0BFB..0C00  ; UNASSIGNED  # <reserved>..<reserved>
0C01..0C03  ; PVALID      # TELUGU SIGN CANDRABINDU..TELUGU SIG
0C04        ; UNASSIGNED  # <reserved>
0C05..0C0C  ; PVALID      # TELUGU LET A..TELUGU LET VOC L
0C0D        ; UNASSIGNED  # <reserved>
0C0E..0C10  ; PVALID      # TELUGU LET E..TELUGU LET AI
0C11        ; UNASSIGNED  # <reserved>
0C12..0C28  ; PVALID      # TELUGU LET O..TELUGU LET NA
0C29        ; UNASSIGNED  # <reserved>
0C2A..0C33  ; PVALID      # TELUGU LET PA..TELUGU LET LLA
0C34        ; UNASSIGNED  # <reserved>
0C35..0C39  ; PVALID      # TELUGU LET VA..TELUGU LET HA
0C3A..0C3C  ; UNASSIGNED  # <reserved>..<reserved>
0C3D..0C44  ; PVALID      # TELUGU SIGN AVAGRAHA..TELUGU VOW SI
0C45        ; UNASSIGNED  # <reserved>
0C46..0C48  ; PVALID      # TELUGU VOW SIGN E..TELUGU VOW SIGN
0C49        ; UNASSIGNED  # <reserved>
0C4A..0C4D  ; PVALID      # TELUGU VOW SIGN O..TELUGU SIGN VIRA
0C4E..0C54  ; UNASSIGNED  # <reserved>..<reserved>
0C55..0C56  ; PVALID      # TELUGU LEN MARK..TELUGU AI LEN MARK
0C57        ; UNASSIGNED  # <reserved>
```

```
   0C58..0C59  ; PVALID      # TELUGU LET TSA..TELUGU LET DZA
   0C5A..0C5F  ; UNASSIGNED  # <reserved>..<reserved>
   0C60..0C63  ; PVALID      # TELUGU LET VOC RR..TELUGU VOW S
   0C64..0C65  ; UNASSIGNED  # <reserved>..<reserved>
   0C66..0C6F  ; PVALID      # TELUGU DIG ZERO..TELUGU DIG NINE
   0C70..0C77  ; UNASSIGNED  # <reserved>..<reserved>
   0C78..0C7F  ; FREE_PVAL   # TELUGU FRACTION DIG ZERO..TELUGU S
   0C80..0C81  ; UNASSIGNED  # <reserved>..<reserved>
   0C82..0C83  ; PVALID      # KANNADA SIGN ANUSVARA..KANNADA SIGN
   0C84        ; UNASSIGNED  # <reserved>
   0C85..0C8C  ; PVALID      # KANNADA LET A..KANNADA LET VOC L
   0C8D        ; UNASSIGNED  # <reserved>
   0C8E..0C90  ; PVALID      # KANNADA LET E..KANNADA LET AI
   0C91        ; UNASSIGNED  # <reserved>
   0C92..0CA8  ; PVALID      # KANNADA LET O..KANNADA LET NA
   0CA9        ; UNASSIGNED  # <reserved>
   0CAA..0CB3  ; PVALID      # KANNADA LET PA..KANNADA LET LLA
   0CB4        ; UNASSIGNED  # <reserved>
   0CB5..0CB9  ; PVALID      # KANNADA LET VA..KANNADA LET HA
   0CBA..0CBB  ; UNASSIGNED  # <reserved>..<reserved>
   0CBC..0CC4  ; PVALID      # KANNADA SIGN NUKTA..KANNADA VOW SIG
   0CC5        ; UNASSIGNED  # <reserved>
   0CC6..0CC8  ; PVALID      # KANNADA VOW SIGN E..KANNADA VOW SIG
   0CC9        ; UNASSIGNED  # <reserved>
   0CCA..0CCD  ; PVALID      # KANNADA VOW SIGN O..KANNADA SIGN VI
   0CCE..0CD4  ; UNASSIGNED  # <reserved>..<reserved>
   0CD5..0CD6  ; PVALID      # KANNADA LEN MARK..KANNADA AI LEN MA
   0CD7..0CDD  ; UNASSIGNED  # <reserved>..<reserved>
   0CDE        ; PVALID      # KANNADA LET FA
   0CDF        ; UNASSIGNED  # <reserved>
   0CE0..0CE3  ; PVALID      # KANNADA LET VOC RR..KANNADA VOW SIG
   0CE4..0CE5  ; UNASSIGNED  # <reserved>..<reserved>
   0CE6..0CEF  ; PVALID      # KANNADA DIG ZERO..KANNADA DIG NINE
   0CF0        ; UNASSIGNED  # <reserved>
   0CF1..0CF2  ; PVALID      # KANNADA SIGN JIHVAMULIYA..KANNADA S
   0CF3..0D01  ; UNASSIGNED  # <reserved>..<reserved>
   0D02..0D03  ; PVALID      # MALAY SIGN ANUSVARA..MALAY SIGN VIS
   0D04        ; UNASSIGNED  # <reserved>
   0D05..0D0C  ; PVALID      # MALAY LET A..MALAY LET VOC
   0D0D        ; UNASSIGNED  # <reserved>
   0D0E..0D10  ; PVALID      # MALAY LET E..MALAY LET AI
   0D11        ; UNASSIGNED  # <reserved>
   0D12..0D3A  ; PVALID      # MALAY LET O..MALAY LET TTTA
   0D3B..0D3C  ; UNASSIGNED  # <reserved>..<reserved>
   0D3D..0D44  ; PVALID      # MALAY SIGN AVAGRAHA..MALAY VOW SIG
   0D45        ; UNASSIGNED  # <reserved>
   0D46..0D48  ; PVALID      # MALAY VOW SIGN E..MALAY VOW SIGN
   0D49        ; UNASSIGNED  # <reserved>
```

```
0D4A..0D4E  ; PVALID      # MALAY VOW SIGN O..MALAY LET DOT REP
0D4F..0D56  ; UNASSIGNED  # <reserved>..<reserved>
0D57        ; PVALID      # MALAY AU LEN MARK
0D58..0D5F  ; UNASSIGNED  # <reserved>..<reserved>
0D60..0D63  ; PVALID      # MALAY LET VOC RR..MALAY VOW
0D64..0D65  ; UNASSIGNED  # <reserved>..<reserved>
0D66..0D6F  ; PVALID      # MALAY DIG ZERO..MALAY DIG NINE
0D70..0D75  ; FREE_PVAL   # MALAY NUM TEN..MALAY FRACTION THR
0D76..0D78  ; UNASSIGNED  # <reserved>..<reserved>
0D79        ; FREE_PVAL   # MALAY DATE MARK
0D7A..0D7F  ; PVALID      # MALAY LET CHILLU NN..MALAY LET
0D80..0D81  ; UNASSIGNED  # <reserved>..<reserved>
0D82..0D83  ; PVALID      # SINH SIGN ANUSVARAYA..SINH SIGN VIS
0D84        ; UNASSIGNED  # <reserved>
0D85..0D96  ; PVALID      # SINH LET AYANNA..SINH LET AUYANN
0D97..0D99  ; UNASSIGNED  # <reserved>..<reserved>
0D9A..0DB1  ; PVALID      # SINH LET ALPAPRAANA KAYANNA..SINH L
0DB2        ; UNASSIGNED  # <reserved>
0DB3..0DBB  ; PVALID      # SINH LET SANYAKA DAYANNA..SINH LETT
0DBC        ; UNASSIGNED  # <reserved>
0DBD        ; PVALID      # SINH LET DANTAJA LAYANNA
0DBE..0DBF  ; UNASSIGNED  # <reserved>..<reserved>
0DC0..0DC6  ; PVALID      # SINH LET VAYANNA..SINH LET FAYAN
0DC7..0DC9  ; UNASSIGNED  # <reserved>..<reserved>
0DCA        ; PVALID      # SINH SIGN AL-LAKUNA
0DCB..0DCE  ; UNASSIGNED  # <reserved>..<reserved>
0DCF..0DD4  ; PVALID      # SINH VOW SIGN AELA-PILLA..SINH VOW
0DD5        ; UNASSIGNED  # <reserved>
0DD6        ; PVALID      # SINH VOW SIGN DIGA PAA-PILLA
0DD7        ; UNASSIGNED  # <reserved>
0DD8..0DDF  ; PVALID      # SINH VOW SIGN GAETTA-PILLA..SINH VO
0DE0..0DF1  ; UNASSIGNED  # <reserved>..<reserved>
0DF2..0DF3  ; PVALID      # SINH VOW SIGN DIGA GAETTA-PILLA..SI
0DF4        ; FREE_PVAL   # SINH PUNCT KUNDDALIYA
0DF5..0E00  ; UNASSIGNED  # <reserved>..<reserved>
0E01..0E32  ; PVALID      # THAI CHAR KO KAI..THAI CHAR SARA A
0E33        ; FREE_PVAL   # THAI CHAR SARA AM
0E34..0E3A  ; PVALID      # THAI CHAR SARA I..THAI CHAR PHINTH
0E3B..0E3E  ; UNASSIGNED  # <reserved>..<reserved>
0E3F        ; FREE_PVAL   # THAI CURRENCY SYM BAHT
0E40..0E4E  ; PVALID      # THAI CHAR SARA E..THAI CHAR YAMAKK
0E4F        ; FREE_PVAL   # THAI CHAR FONGMAN
0E50..0E59  ; PVALID      # THAI DIG ZERO..THAI DIG NINE
0E5A..0E5B  ; FREE_PVAL   # THAI CHAR ANGKHANKHU..THAI CHAR KH
0E5C..0E80  ; UNASSIGNED  # <reserved>..<reserved>
0E81..0E82  ; PVALID      # LAO LET KO..LAO LET KHO SUNG
0E83        ; UNASSIGNED  # <reserved>
0E84        ; PVALID      # LAO LET KHO TAM
```

```
   0E85..0E86  ; UNASSIGNED  # <reserved>..<reserved>
   0E87..0E88  ; PVALID      # LAO LET NGO..LAO LET CO
   0E89        ; UNASSIGNED  # <reserved>
   0E8A        ; PVALID      # LAO LET SO TAM
   0E8B..0E8C  ; UNASSIGNED  # <reserved>..<reserved>
   0E8D        ; PVALID      # LAO LET NYO
   0E8E..0E93  ; UNASSIGNED  # <reserved>..<reserved>
   0E94..0E97  ; PVALID      # LAO LET DO..LAO LET THO TAM
   0E98        ; UNASSIGNED  # <reserved>
   0E99..0E9F  ; PVALID      # LAO LET NO..LAO LET FO SUNG
   0EA0        ; UNASSIGNED  # <reserved>
   0EA1..0EA3  ; PVALID      # LAO LET MO..LAO LET LO LING
   0EA4        ; UNASSIGNED  # <reserved>
   0EA5        ; PVALID      # LAO LET LO LOOT
   0EA6        ; UNASSIGNED  # <reserved>
   0EA7        ; PVALID      # LAO LET WO
   0EA8..0EA9  ; UNASSIGNED  # <reserved>..<reserved>
   0EAA..0EAB  ; PVALID      # LAO LET SO SUNG..LAO LET HO SUNG
   0EAC        ; UNASSIGNED  # <reserved>
   0EAD..0EB2  ; PVALID      # LAO LET O..LAO VOW SIGN AA
   0EB3        ; FREE_PVAL   # LAO VOW SIGN AM
   0EB4..0EB9  ; PVALID      # LAO VOW SIGN I..LAO VOW SIGN UU
   0EBA        ; UNASSIGNED  # <reserved>
   0EBB..0EBD  ; PVALID      # LAO VOW SIGN MAI KON..LAO SEMIVOW SIG
   0EBE..0EBF  ; UNASSIGNED  # <reserved>..<reserved>
   0EC0..0EC4  ; PVALID      # LAO VOW SIGN E..LAO VOW SIGN AI
   0EC5        ; UNASSIGNED  # <reserved>
   0EC6        ; PVALID      # LAO KO LA
   0EC7        ; UNASSIGNED  # <reserved>
   0EC8..0ECD  ; PVALID      # LAO TONE MAI EK..LAO NIGGAHITA
   0ECE..0ECF  ; UNASSIGNED  # <reserved>..<reserved>
   0ED0..0ED9  ; PVALID      # LAO DIG ZERO..LAO DIG NINE
   0EDA..0EDB  ; UNASSIGNED  # <reserved>..<reserved>
   0EDC..0EDD  ; FREE_PVAL   # LAO HO NO..LAO HO MO
   0EDE..0EDF  ; PVALID      # LAO LET KHMU GO..TIB SYL OM
   0EE0..0EEF  ; UNASSIGNED  # <reserved>..<reserved>
   0F00        ; PVALID      # TIB SYLL OM
   0F01..0F0A  ; FREE_PVAL   # TIB MARK GTER YIG MGO TRUNC A..TIB
   0F0B        ; PVALID      # TIB MARK INTERSYLLABIC TSHEG
   0F0C..0F17  ; FREE_PVAL   # TIB MARK DELIMITER TSHEG BSTAR..TIB
   0F18..0F19  ; PVALID      # TIB ASTROLOGICAL SIGN -KHYUD PA..TIB
   0F1A..0F1F  ; FREE_PVAL   # TIB SIGN RDEL DKAR GCIG..TIB SIGN RD
   0F20..0F29  ; PVALID      # TIB DIG ZERO..TIB DIG NINE
   0F2A..0F34  ; FREE_PVAL   # TIB DIG HALF ONE..TIB MARK BSDUS R
   0F35        ; PVALID      # TIB MARK NGAS BZUNG NYI ZLA
   0F36        ; FREE_PVAL   # TIB MARK CARET DZUD RTAGS BZHI MIG C
   0F37        ; PVALID      # TIB MARK NGAS BZUNG SGOR RTAGS
   0F38        ; FREE_PVAL   # TIB MARK CHE MGO
```

```
   0F39        ; PVALID      # TIB MARK TSA PHRU
   0F3A..0F3D  ; FREE_PVAL   # TIB MARK GUG RTAGS GYON..TIB MARK AN
   0F3E..0F47  ; PVALID      # TIB SIGN YAR TSHES..TIB LET JA
   0F48        ; UNASSIGNED  # <reserved>
   0F49..0F6C  ; PVALID      # TIB LET NYA..TIB LET RRA
   0F6D..0F70  ; UNASSIGNED  # <reserved>..<reserved>
   0F71..0F76  ; PVALID      # TIB VOW SIGN AA..TIB VOW SIGN VO
   0F77        ; FREE_PVAL   # TIB VOW SIGN VO RR
   0F78        ; PVALID      # TIB VOW SIGN VO L
   0F79        ; FREE_PVAL   # TIB VOW SIGN VO LL
   0F7A..0F84  ; PVALID      # TIB VOW SIGN E..TIB MARK H
   0F85        ; FREE_PVAL   # TIB MARK PALUTA
   0F86..0F8F  ; PVALID      # TIB SIGN LCI RTAGS..TIB SUBJOIN S
   0F90..0F97  ; PVALID      # TIB SUBJOIN LET KA..TIB SUBJOIN
   0F98        ; UNASSIGNED  # <reserved>
   0F99..0FBC  ; PVALID      # TIB SUBJOIN LET NYA..TIB SUBJOI
   0FBD        ; UNASSIGNED  # <reserved>
   0FBE..0FC5  ; FREE_PVAL   # TIB KU RU KHA..TIB SYM RDO RJE
   0FC6        ; PVALID      # TIB SYM PADMA GDAN
   0FC7..0FCC  ; FREE_PVAL   # TIB SYM RDO RJE RGYA GRAM..TIB SY
   0FCD        ; UNASSIGNED  # <reserved>
   0FCE..0FDA  ; FREE_PVAL   # TIB SIGN RDEL NAG RDEL DKAR..TIB MA
   0FDB..0FFF  ; UNASSIGNED  # <reserved>..<reserved>
   1000..1049  ; PVALID      # MYAN LET KA..MYAN DIG NINE
   104A..104F  ; FREE_PVAL   # MYAN SIGN LITTLE SECTION..MYAN SYM
   1050..109D  ; PVALID      # MYAN LET SHA..MYAN VOW SIGN AITON
   109E..109F  ; FREE_PVAL   # MYAN SYM SHAN ONE..MYAN SYM SHAN EX
   10A0..10C5  ; PVALID      # GEORG CAP LET AN..GEORG CAP LET HOE
   10C6        ; UNASSIGNED  # <reserved>
   10C7        ; PVALID      # GEORG CAP LET YN
   10C8..10CC  ; UNASSIGNED  # <reserved>..<reserved>
   10CD        ; PVALID      # GEORG CAP LET AEN
   10CE..10CF  ; UNASSIGNED  # <reserved>..<reserved>
   10D0..10FA  ; PVALID      # GEORG LET AN..GEORG LET AIN
   10FB..10FC  ; FREE_PVAL   # GEORG PARA SEP..MOD LET GEORG NAR
   10FD..10FF  ; PVALID      # GEORG LET AEN..GEORG LET LABIAL
   1100..11FF  ; DISALLOWED  # HANGUL CHO KIYEOK..HANGUL JONG SSA
   1200..1248  ; PVALID      # ETHI SYL HA..ETHI SYL QWA
   1249        ; UNASSIGNED  # <reserved>
   124A..124D  ; PVALID      # ETHI SYL QWI..ETHI SYL QWE
   124E..124F  ; UNASSIGNED  # <reserved>..<reserved>
   1250..1256  ; PVALID      # ETHI SYL QHA..ETHI SYL QHO
   1257        ; UNASSIGNED  # <reserved>
   1258        ; PVALID      # ETHI SYL QHWA
   1259        ; UNASSIGNED  # <reserved>
   125A..125D  ; PVALID      # ETHI SYL QHWI..ETHI SYL QH
   125E..125F  ; UNASSIGNED  # <reserved>..<reserved>
   1260..1288  ; PVALID      # ETHI SYL BA..ETHI SYL XWA
```

```
1289        ; UNASSIGNED  # <reserved>
128A..128D  ; PVALID      # ETHI SYL XWI..ETHI SYL XWE
128E..128F  ; UNASSIGNED  # <reserved>..<reserved>
1290..12B0  ; PVALID      # ETHI SYL NA..ETHI SYL KWA
12B1        ; UNASSIGNED  # <reserved>
12B2..12B5  ; PVALID      # ETHI SYL KWI..ETHI SYL KWE
12B6..12B7  ; UNASSIGNED  # <reserved>..<reserved>
12B8..12BE  ; PVALID      # ETHI SYL KXA..ETHI SYL KXO
12BF        ; UNASSIGNED  # <reserved>
12C0        ; PVALID      # ETHI SYL KXWA
12C1        ; UNASSIGNED  # <reserved>
12C2..12C5  ; PVALID      # ETHI SYL KXWI..ETHI SYL KX
12C6..12C7  ; UNASSIGNED  # <reserved>..<reserved>
12C8..12D6  ; PVALID      # ETHI SYL WA..ETHI SYL PHAR
12D7        ; UNASSIGNED  # <reserved>
12D8..1310  ; PVALID      # ETHI SYL ZA..ETHI SYL GWA
1311        ; UNASSIGNED  # <reserved>
1312..1315  ; PVALID      # ETHI SYL GWI..ETHI SYL GWE
1316..1317  ; UNASSIGNED  # <reserved>..<reserved>
1318..135A  ; PVALID      # ETHI SYL GGA..ETHI SYL FYA
135B..135C  ; UNASSIGNED  # <reserved>..<reserved>
135D..135F  ; PVALID      # ETHI COMB GEM AND VOW..ETHI COMB GE
1360..137C  ; FREE_PVAL   # ETHI SECT MARK..ETHI NUM TEN THOUS
137D..137F  ; UNASSIGNED  # <reserved>..<reserved>
1380..138F  ; PVALID      # ETHI SYL SEBATBEIT MWA..ETHI SYL PW
1390..1399  ; FREE_PVAL   # ETHI TON MARK YIZET..ETHI TON MARK
139A..139F  ; UNASSIGNED  # <reserved>..<reserved>
13A0..13F4  ; PVALID      # CHEROKEE LET A..CHEROKEE LET YV
13F5..13FF  ; UNASSIGNED  # <reserved>..<reserved>
1400        ; FREE_PVAL   # CANAD SYL HYPHEN
1401..166C  ; PVALID      # CANAD SYL E..CANAD SYL CAR
166D..166E  ; FREE_PVAL   # CANAD SYL CHI SIGN..CANAD SYLLAB
166F..167F  ; PVALID      # CANAD SYL QAI..CANAD SYL B
1680        ; FREE_PVAL   # OGHAM SPACE MARK
1681..169A  ; PVALID      # OGHAM LET BEITH..OGHAM LET PEITH
169B..169C  ; FREE_PVAL   # OGHAM FEATHER MARK..OGHAM REV FEAT
169D..169F  ; UNASSIGNED  # <reserved>..<reserved>
16A0..16EA  ; PVALID      # RUNIC LET FEHU FEOH FE F..RUNIC LET
16EB..16F0  ; FREE_PVAL   # RUNIC SINGLE PUNCT..RUNIC BELGTHOR
16F1..16FF  ; UNASSIGNED  # <reserved>..<reserved>
1700..170C  ; PVALID      # TAGALOG LET A..TAGALOG LET YA
170D        ; UNASSIGNED  # <reserved>
170E..1714  ; PVALID      # TAGALOG LET LA..TAGALOG SIGN VIRAMA
1715..171F  ; UNASSIGNED  # <reserved>..<reserved>
1720..1734  ; PVALID      # HANUNOO LET A..HANUNOO SIGN PAMUDPO
1735..1736  ; FREE_PVAL   # PHILIP SINGLE PUNCT..PHILIP DOUBLE
1737..173F  ; UNASSIGNED  # <reserved>..<reserved>
1740..1753  ; PVALID      # BUHID LET A..BUHID VOW SIGN U
```

```
   1754..175F  ; UNASSIGNED  # <reserved>..<reserved>
   1760..176C  ; PVALID      # TAGBANWA LET A..TAGBANWA LET YA
   176D        ; UNASSIGNED  # <reserved>
   176E..1770  ; PVALID      # TAGBANWA LET LA..TAGBANWA LET SA
   1771        ; UNASSIGNED  # <reserved>
   1772..1773  ; PVALID      # TAGBANWA VOW SIGN I..TAGBANWA VOW S
   1774..177F  ; UNASSIGNED  # <reserved>..<reserved>
   1780..17B3  ; PVALID      # KHMER LET KA..KHMER IND VOW QAU
   17B4..17B5  ; DISALLOWED  # KHMER VOW INH AQ..KHMER VOW INH AA
   17B6..17D3  ; PVALID      # KHMER VOW SIGN AA..KHMER SIGN BATHA
   17D4..17D6  ; FREE_PVAL   # KHMER SIGN KHAN..KHMER SIGN CAMNUC
   17D7        ; PVALID      # KHMER SIGN LEK TOO
   17D8..17DB  ; FREE_PVAL   # KHMER SIGN BEYYAL..KHMER CURR SYM R
   17DC..17DD  ; PVALID      # KHMER SIGN AVAKRAHASANYA..KHMER SIG
   17DE..17DF  ; UNASSIGNED  # <reserved>..<reserved>
   17E0..17E9  ; PVALID      # KHMER DIG ZERO..KHMER DIG NINE
   17EA..17EF  ; UNASSIGNED  # <reserved>..<reserved>
   17F0..17F9  ; FREE_PVAL   # KHMER SYM LEK ATTAK SON..KHMER SYM
   17FA..17FF  ; UNASSIGNED  # <reserved>..<reserved>
   1800..180A  ; FREE_PVAL   # MONG BIRGA..MONG NIRUGU
   180B..180E  ; DISALLOWED  # MONG FREE VAR SEL ONE..MONG VOW SEP
   180F        ; UNASSIGNED  # <reserved>
   1810..1819  ; PVALID      # MONG DIG ZERO..MONG DIG NINE
   181A..181F  ; UNASSIGNED  # <reserved>..<reserved>
   1820..1877  ; PVALID      # MONG LET A..MONG LET MANCHU
   1878..187F  ; UNASSIGNED  # <reserved>..<reserved>
   1880..18AA  ; PVALID      # MONG LET ALI GALI ANUSVARA ONE..MON
   18AB..18AF  ; UNASSIGNED  # <reserved>..<reserved>
   18B0..18F5  ; PVALID      # CAN SYL OY..CAN SYL CA
   18F6..18FF  ; UNASSIGNED  # <reserved>..<reserved>
   1900..191C  ; PVALID      # LIMBU VOW-CARRIER LET..LIMBU LET HA
   191D..191F  ; UNASSIGNED  # <reserved>..<reserved>
   1920..192B  ; PVALID      # LIMBU VOW SIGN A..LIMBU SUBJOIN LET
   192C..192F  ; UNASSIGNED  # <reserved>..<reserved>
   1930..193B  ; PVALID      # LIMBU SM LET KA..LIMBU SIGN SA-I
   193C..193F  ; UNASSIGNED  # <reserved>..<reserved>
   1940        ; FREE_PVAL   # LIMBU SIGN LOO
   1941..1943  ; UNASSIGNED  # <reserved>..<reserved>
   1944..1945  ; FREE_PVAL   # LIMBU EXCLAM MARK..LIMBU QUEST MARK
   1946..196D  ; PVALID      # LIMBU DIG ZERO..TAI LE LET AI
   196E..196F  ; UNASSIGNED  # <reserved>..<reserved>
   1970..1974  ; PVALID      # TAI LE LET TONE-2..TAI LE LET TONE-
   1975..197F  ; UNASSIGNED  # <reserved>..<reserved>
   1980..19AB  ; PVALID      # NEW TAI LUE LET HIGH QA..NEW TAI LU
   19AC..19AF  ; UNASSIGNED  # <reserved>..<reserved>
   19B0..19C9  ; PVALID      # NEW TAI LUE VOW SIGN VOW SHORT..NEW
   19CA..19CF  ; UNASSIGNED  # <reserved>..<reserved>
   19D0..19D9  ; PVALID      # NEW TAI LUE DIG ZERO..NEW TAI DIG N
```

```
   19DA        ; DISALLOWED  # NEW TAI LUE THAM
   19DB..19DD  ; UNASSIGNED  # <reserved>..<reserved>
   19DE..19FF  ; FREE_PVAL   # NEW TAI LUE SIGN LAE..KHMER SYM DAP
   1A00..1A1B  ; PVALID      # BUGIN LET KA..BUGIN VOW SIGN AE
   1A1C..1A1D  ; UNASSIGNED  # <reserved>..<reserved>
   1A1E..1A1F  ; FREE_PVAL   # BUGIN PALLAWA..BUGIN END OF SECTION
   1A20..1A5E  ; PVALID      # TAI THAM LET HIGH KA..TAI THAM CONS
   1A5F        ; UNASSIGNED  # <reserved>
   1A60..1A7C  ; PVALID      # TAI THAM SIGN SAKOT..TAI THAM SIGN
   1A7D..1A7E  ; UNASSIGNED  # <reserved>..<reserved>
   1A7F..1A89  ; PVALID      # TAI THAM COMB CRYPT DOT..TAI THAM D
   1A8A..1A8F  ; UNASSIGNED  # <reserved>..<reserved>
   1A90..1A99  ; PVALID      # TAI THAM THAM DIG ZERO..TAI THAM TH
   1A9A..1A9F  ; UNASSIGNED  # <reserved>..<reserved>
   1AA0..1AA6  ; FREE_PVAL   # TAI THAM SIGN WIANG..TAI THAM SIGN
   1AA7        ; PVALID      # TAI THAM SIGN MAI YAMOK
   1AA8..1AAD  ; FREE_PVAL   # TAI THAM SIGN KAAN..TAI THAM SIGN C
   1AAE..1AFF  ; UNASSIGNED  # <reserved>..<reserved>
   1B00..1B4B  ; PVALID      # BAL SIGN ULU RICEM..BAL LET ASYURA
   1B4C..1B4F  ; UNASSIGNED  # <reserved>..<reserved>
   1B50..1B59  ; PVALID      # BAL DIG ZERO..BAL DIG NINE
   1B5A..1B6A  ; FREE_PVAL   # BAL PANTI..BAL MUS SYM DANG
   1B6B..1B73  ; PVALID      # BAL MUS SYM COMB TEGEH..BAL MUS
   1B74..1B7C  ; FREE_PVAL   # BAL MUS SYM RIGHT-HAND OPEN DUG
   1B7D..1B7F  ; UNASSIGNED  # <reserved>..<reserved>
   1B80..1BF3  ; PVALID      # SUND SIGN PANYECEK..BATAK PANONGONAN
   1BF4..1BFB  ; UNASSIGNED  # <reserved>..<reserved>
   1BFC..1BFF  ; FREE_PVAL   # BATAK SYM BINDU NA METEK..BATAK SYM
   1C00..1C37  ; PVALID      # LEPCHA LET KA..LEPCHA SIGN NUKTA
   1C38..1C3A  ; UNASSIGNED  # <reserved>..<reserved>
   1C3B..1C3F  ; FREE_PVAL   # LEPCHA PUNCT TA-ROL..LEPCHA PUNCT T
   1C40..1C49  ; PVALID      # LEPCHA DIG ZERO..LEPCHA DIG NINE
   1C4A..1C4C  ; UNASSIGNED  # <reserved>..<reserved>
   1C4D..1C7D  ; PVALID      # LEPCHA LET TTA..OL CHIKI AHAD
   1C7E..1C7F  ; FREE_PVAL   # OL CHIKI PUNCT MUCAAD..OL CHIKI PUN
   1C80..1CBF  ; UNASSIGNED  # <reserved>..<reserved>
   1CC0..1CC7  ; FREE_PVAL   # SUNDA PUNCT BINDU SURYA..SUNDA PUNC
   1CC8..1CCF  ; UNASSIGNED  # <reserved>..<reserved>
   1CD0..1CD2  ; PVALID      # VED TONE KARSHANA..VED TONE PRENKHA
   1CD3        ; FREE_PVAL   # VED SIGN NIHSHVASA
   1CD4..1CF6  ; PVALID      # VED SIGN YAJURVEDIC MID SVARITA..VE
   1CF7..1CFF  ; UNASSIGNED  # <reserved>..<reserved>
   1D00..1D2B  ; PVALID      # LAT LET SM CAP A..CYR LET SM
   1D2C..1D2E  ; FREE_PVAL   # MOD LET CAP A..MOD LET C
   1D2F        ; PVALID      # MOD LET CAP BARRED B
   1D30..1D3A  ; FREE_PVAL   # MOD LET CAP D..MOD LET C
   1D3B        ; PVALID      # MOD LET CAP REV N
   1D3C..1D4D  ; FREE_PVAL   # MOD LET CAP O..MOD LET S
```

```
   1D4E        ; PVALID      # MOD LET SM TURNED I
   1D4F..1D6A  ; FREE_PVAL   # MOD LET SM K..GREEK SUB SMA
   1D6B..1D77  ; PVALID      # LAT SM LET UE..LAT SM LET TU
   1D78        ; FREE_PVAL   # MOD LET CYR EN
   1D79..1D9A  ; PVALID      # LAT SM LET INSULAR G..LAT SM LE
   1D9B..1DBF  ; FREE_PVAL   # MOD LET SM TURNED ALPHA..MOD
   1DC0..1DE6  ; PVALID      # COMB DOTTED GRAVE ACCENT..COMB LAT
   1DE7..1DFB  ; UNASSIGNED  # <reserved>..<reserved>
   1DFC..1E99  ; PVALID      # COMB DOUBLE INV BREVE BEL..LAT SM L
   1E9A        ; FREE_PVAL   # LAT SM LET A W R HALF RING
   1E9B..1F15  ; PVALID      # LAT SM LET LONG S W BOT ABOVE..GR
   1F16..1F17  ; UNASSIGNED  # <reserved>..<reserved>
   1F18..1F1D  ; FREE_PVAL   # GREEK CAP LET EPSILON W PSILI..GRE
   1F1E..1F1F  ; UNASSIGNED  # <reserved>..<reserved>
   1F20..1F45  ; PVALID      # GREEK SM LET ETA W PSILI..GREEK SMA
   1F46..1F47  ; UNASSIGNED  # <reserved>..<reserved>
   1F48..1F4D  ; FREE_PVAL   # GREEK CAP LET OMICRON W PSILI..GRE
   1F4E..1F4F  ; UNASSIGNED  # <reserved>..<reserved>
   1F50..1F57  ; PVALID      # GREEK SM LET UPSILON W PSILI..GREEK
   1F58        ; UNASSIGNED  # <reserved>
   1F59        ; PVALID      # GREEK CAP LET UPSILON W DASIA
   1F5A        ; UNASSIGNED  # <reserved>
   1F5B        ; PVALID      # GREEK CAP LET UPSILON W DASIA AND
   1F5C        ; UNASSIGNED  # <reserved>
   1F5D        ; PVALID      # GREEK CAP LET UPSILON W DASIA AND
   1F5E        ; UNASSIGNED  # <reserved>
   1F5F..1F7D  ; PVALID      # GREEK CAP LET UPSILON W DASIA A..GR
   1F7E..1F7F  ; UNASSIGNED  # <reserved>..<reserved>
   1F80..1F87  ; PVALID      # GREEK SM LET ALPHA W PSILI AND YPOG
   1F88..1F8F  ; FREE_PVAL   # GREEK CAP LET ALPHA W PSILI AND..GR
   1F90..1F97  ; PVALID      # GREEK SM LET ETA W PSILI AND YP..GR
   1F98..1F9F  ; FREE_PVAL   # GREEK CAP LET ETA W PSILI AND P..GR
   1FA0..1FA7  ; PVALID      # GREEK SM LET OMEGA W PSILI AND ..GR
   1FA8..1FAF  ; FREE_PVAL   # GREEK CAPL LET OMEGA W PSILI AN..GR
   1FB0..1FB4  ; PVALID      # GREEK SM LET ALPHA W VRACHY..GREEK
   1FB5        ; UNASSIGNED  # <reserved>
   1FB6..1FBB  ; PVALID      # GREEK SM LET ALPHA W PERISPOMEN..GR
   1FBC..1FBD  ; FREE_PVAL   # GREEK CAP LET ALPHA W PROSGEGRA..GR
   1FBE        ; PVALID      # GREEK PROSGEGRAMMENI
   1FBF..1FC1  ; FREE_PVAL   # GREEK PSILI..GREEK DIALYTIKA AND PE
   1FC2..1FC4  ; PVALID      # GREEK SM LET ETA W VARIA AND YP..GR
   1FC5        ; UNASSIGNED  # <reserved>
   1FC6..1FCB  ; PVALID      # GREEK SM LET ETA W PERISPOMENI..GR
   1FCC..1FCF  ; FREE_PVAL   # GREEK CAP LET ETA W PROSGEGRAM..GR
   1FD0..1FD3  ; PVALID      # GREEK SM LET IOTA W VRACHY..GREEK S
   1FD4..1FD5  ; UNASSIGNED  # <reserved>..<reserved>
   1FD6..1FDB  ; PVALID      # GREEK SM LET IOTA W PERISPOMENI..GR
   1FDC        ; UNASSIGNED  # <reserved>
```

```
1FDD..1FDF  ; FREE_PVAL   # GREEK DASIA AND VARIA..GREEK DASIA
1FE0..1FEC  ; PVALID      # GREEK SM LET UPSILON W VRACHY..GREE
1FED..1FEF  ; FREE_PVAL   # GREEK DIALYTIKA AND VARIA..GREEK VA
1FF0..1FF1  ; UNASSIGNED  # <reserved>..<reserved>
1FF2..1FF4  ; FREE_PVAL   # GREEK SM LET OMEGA W VARIA AND YPOG
1FF5        ; UNASSIGNED  # <reserved>
1FF6..1FFB  ; PVALID      # GREEK SM LET OMEGA W PERISPOMEN..GR
1FFC..1FFE  ; FREE_PVAL   # GREEK CAP LET OMEGA W PROSGEGRA..GR
1FFF        ; UNASSIGNED  # <reserved>
2000..200A  ; FREE_PVAL   # EN QUAD..HAIR SPACE
200B        ; DISALLOWED  # ZERO WIDTH SPACE
200C..200D  ; CONTEXTJ    # ZERO WIDTH NON-JOINER..ZERO WIDTH J
200E..200F  ; DISALLOWED  # LEFT-TO-RIGHT MARK..RIGHT-TO-LEFT M
2010..2027  ; FREE_PVAL   # HYPHEN..HYPHENATION POINT
2028..202E  ; DISALLOWED  # LINE SEP..RIGHT-TO-LEFT OVERRIDE
202F..205F  ; FREE_PVAL   # NARROW NO-BREAK SPACE..MED MATH SP
2060..2064  ; DISALLOWED  # WORD JOINER..INVISIBLE PLUS
2065        ; UNASSIGNED  # <reserved>
2066..206F  ; DISALLOWED  # LEFT-TO-RIGHT IS..NOM DIGIT SHAPES
2070..2071  ; FREE_PVAL   # SUPER ZERO..SUPER LAT SM LET I
2072..2073  ; UNASSIGNED  # <reserved>..<reserved>
2074..208E  ; FREE_PVAL   # SUPER FOUR..SUB RIGHT PARENTHESIS
208F        ; UNASSIGNED  # <reserved>
2090..209C  ; FREE_PVAL   # LAT SUB SM LET A..LAT SUB SM LET T
209D..209F  ; UNASSIGNED  # <reserved>..<reserved>
20A0..20BA  ; FREE_PVAL   # EURO-CURRENCY SIGN..TURKISH LIRA SI
20BB..20CF  ; UNASSIGNED  # <reserved>..<reserved>
20D0..20DC  ; PVALID      # COMB LEFT HARPOON ABOVE..COMB FOUR
20DD..20E0  ; FREE_PVAL   # COMB ENC CIRC..COMB ENC CIRC BACKS
20E1        ; PVALID      # COMB L R ARROW ABOVE
20E2..20E4  ; FREE_PVAL   # COMB ENC SCREEN..COMB ENC UPWARD PO
20E5..20F0  ; PVALID      # COMB REV SOLIDUS OVERLAY..COMB ASTE
20F1..20FF  ; UNASSIGNED  # <reserved>..<reserved>
2100..2129  ; FREE_PVAL   # ACCOUNT OF..TURNED GREEK SM LET IOT
212A..212B  ; PVALID      # KELVIN SIGN..ANGSTROM SIGN
212C..2131  ; FREE_PVAL   # SCRIPT CAP C..SCRIPT CAP F
2132        ; PVALID      # TURNED CAP F
2133..214D  ; FREE_PVAL   # SCRIPT CAP M..AKTIESELSKAB
214E        ; PVALID      # TURNED SM F
214F..2182  ; FREE_PVAL   # SYM FOR SAMAR SOURCE..ROM NUM TEN T
2183..2184  ; PVALID      # ROM NUM REV ONE HUNDRED..LAT SM LET
2185..2189  ; FREE_PVAL   # ROM NUM SIX LATE FORM..VULGAR FRACT
218A..218F  ; UNASSIGNED  # <reserved>..<reserved>
2190..23F3  ; FREE_PVAL   # LEFTWARDS ARROW..HOURGLASS W FLO
23F4..23FF  ; UNASSIGNED  # <reserved>..<reserved>
2400..2426  ; FREE_PVAL   # SYM FOR NULL..SYM FOR SUB FORM
2427..243F  ; UNASSIGNED  # <reserved>..<reserved>
2440..244A  ; FREE_PVAL   # OCR HOOK..OCR DOUBLE BACKSLASH
```

```
   244B..245F  ; UNASSIGNED  # <reserved>..<reserved>
   2460..26FF  ; FREE_PVAL   # CIRCLED DIG ONE..WHITE FLAG W HORIZ
   2700        ; UNASSIGNED  # <reserved>
   2701..2B4C  ; FREE_PVAL   # UP BLADE SCISSORS..RIGHTWARDS ARROW
   2B4D..2B4F  ; UNASSIGNED  # <reserved>..<reserved>
   2B50..2B59  ; FREE_PVAL   # WHITE MEDIUM STAR..HEAVY CIRCLED SA
   2B5A..2BFF  ; UNASSIGNED  # <reserved>..<reserved>
   2C00..2C2E  ; PVALID      # GLAG CAP LET AZU..GLAG CA
   2C2F        ; UNASSIGNED  # <reserved>
   2C30..2C5E  ; PVALID      # GLAG SM LET AZU..GLAG SMAL
   2C5F        ; UNASSIGNED  # <reserved>
   2C60..2C7B  ; PVALID      # LAT CAP LET L W DOUBLE BAR..LAT SM
   2C7C..2C7D  ; FREE_PVAL   # LAT SUB SM LET J..MOD LET CAP V
   2C7E..2CE4  ; PVALID      # LAT CAP LET S W SWASH TAIL..COPT SY
   2CE5..2CEA  ; FREE_PVAL   # COPT SYM MI RO..COPT SYM SHIMA SIMA
   2CEB..2CF3  ; PVALID      # COPT CAP LET CRYPTOGRAMMIC SHEI..CO
   2CF4..2CF8  ; UNASSIGNED  # <reserved>..<reserved>
   2CF9..2CFF  ; FREE_PVAL   # COPT OLD NUB FULL STOP..COPT MORPHO
   2D00..2D25  ; PVALID      # GEORG SM LET AN..GEORG SM LET
   2D26        ; UNASSIGNED  # <reserved>
   2D27        ; PVALID      # GEORG SM LET YN
   2D28..2D2C  ; UNASSIGNED  # <reserved>..<reserved>
   2D2D        ; PVALID      # GEORG SM LET AEN
   2D2E..2D2F  ; UNASSIGNED  # <reserved>..<reserved>
   2D30..2D67  ; PVALID      # TIFINAGH LET YA..TIFINAGH LETTER YO
   2D68..2D6E  ; UNASSIGNED  # <reserved>..<reserved>
   2D6F..2D70  ; FREE_PVAL   # TIFINAGH MOD LET LABIALIZATION MARK
   2D71..2D7E  ; UNASSIGNED  # <reserved>..<reserved>
   2D7F..2D96  ; PVALID      # TIFINAGH CONS JOINER..ETHI SYL GGW
   2D97..2D9F  ; UNASSIGNED  # <reserved>..<reserved>
   2DA0..2DA6  ; PVALID      # ETHI SYL SSA..ETHI SYL SSO
   2DA7        ; UNASSIGNED  # <reserved>
   2DA8..2DAE  ; PVALID      # ETHI SYL CCA..ETHI SYL CCO
   2DAF        ; UNASSIGNED  # <reserved>
   2DB0..2DB6  ; PVALID      # ETHI SYL ZZA..ETHI SYL ZZO
   2DB7        ; UNASSIGNED  # <reserved>
   2DB8..2DBE  ; PVALID      # ETHI SYL CCHA..ETHI SYL CC
   2DBF        ; UNASSIGNED  # <reserved>
   2DC0..2DC6  ; PVALID      # ETHI SYL QYA..ETHI SYL QYO
   2DC7        ; UNASSIGNED  # <reserved>
   2DC8..2DCE  ; PVALID      # ETHI SYL KYA..ETHI SYL KYO
   2DCF        ; UNASSIGNED  # <reserved>
   2DD0..2DD6  ; PVALID      # ETHI SYL XYA..ETHI SYL XYO
   2DD7        ; UNASSIGNED  # <reserved>
   2DD8..2DDE  ; PVALID      # ETHI SYL GYA..ETHI SYL GYO
   2DDF        ; UNASSIGNED  # <reserved>
   2DE0..2DFF  ; PVALID      # COMB CYR LET BE..COMB CYRI
   2E00..2E2E  ; FREE_PVAL   # RIGHT ANGLE SUB MARK..REV QUEST MAR
```

```
2E2F        ; PVALID      # VERT TILDE
2E30..2E3B  ; FREE_PVAL   # RING PNT..THREE-EM DASH
2E3C..2E7F  ; UNASSIGNED  # <reserved>..<reserved>
2E80..2E99  ; FREE_PVAL   # CJK RAD REPEAT..CJK RAD RAP
2E9A        ; UNASSIGNED  # <reserved>
2E9B..2EF3  ; FREE_PVAL   # CJK RAD CHOKE..CJK RAD C-SIMPLIFIED
2EF4..2EFF  ; UNASSIGNED  # <reserved>..<reserved>
2F00..2FD5  ; FREE_PVAL   # KANGXI RAD ONE..KANGXI RAD FLUTE
2FD6..2FEF  ; UNASSIGNED  # <reserved>..<reserved>
2FF0..2FFB  ; FREE_PVAL   # IDEO DESC CHAR LEFT TO RIGHT..IDEO
2FFC..2FFF  ; UNASSIGNED  # <reserved>..<reserved>
3000..3004  ; FREE_PVAL   # IDEO SPACE..JAPAN INDUST STAND
3005..3007  ; PVALID      # IDEO ITER MARK..IDEO NUMB ZERO
3008..3029  ; FREE_PVAL   # LEFT ANGLE BRACKET..HANGZH NUM NINE
302A..302D  ; PVALID      # IDEO LEVEL TONE MARK..IDEO ENT
302E..302F  ; DISALLOWED  # HANGUL SING DOT TONE MARK..WAVY DAS
3030        ; FREE_PVAL   # WAVY DASH
3031..3035  ; DISALLOWED  # VERT KANA REP MARK..VERT KANA REP M
3036..303A  ; FREE_PVAL   # CIRCLED POSTAL MARK..HANGZH NUM THI
303B        ; DISALLOWED  # VERT IDEO ITER MARK
303C        ; PVALID      # MASU MARK
303D..303F  ; FREE_PVAL   # PART ALTER MARK..IDEO HALF FILL
3040        ; UNASSIGNED  # <reserved>
3041..3096  ; PVALID      # HIRAGANA LET SM A..HIRAGANA LET SMA
3097..3098  ; UNASSIGNED  # <reserved>..<reserved>
3099..309A  ; PVALID      # COMB KAT-HIR VOICED SOUND
309B..309C  ; FREE_PVAL   # KAT-HIR VOICED SOUND MARK..KAT-HIR
309D..309E  ; PVALID      # HIRAGANA ITER MARK..HIRAGANA VOICED
309F..30A0  ; FREE_PVAL   # HIRAGANA DIGRAPH YORI..KAT-HIR DOU
30A1..30FA  ; PVALID      # KATAKANA LET SM A..KATAKANA LET VO
30FB        ; CONTEXTO    # KATAKANA MIDDLE DOT
30FC..30FE  ; PVALID      # KAT-HIR PROLONGED SOUND MARK..KATA
30FF        ; FREE_PVAL   # KATAKANA DIGRAPH KOTO
3100..3104  ; UNASSIGNED  # <reserved>..<reserved>
3105..312D  ; PVALID      # BOPOMOFO LET B..BOPOMOFO LET IH
312E..3130  ; UNASSIGNED  # <reserved>..<reserved>
3131..3163  ; FREE_PVAL   # HANGUL LET KIYEOK..HANGUL LET I
3164        ; DISALLOWED  # HANGUL FILLER
3165..318E  ; FREE_PVAL   # HANGUL LET SSANGNIEUN..HANGUL LET
318F        ; UNASSIGNED  # <reserved>
3190..319F  ; FREE_PVAL   # IDEO ANNO LINK MARK..IDEO ANNO MAN
31A0..31BA  ; PVALID      # BOPOMOFO LET BU..BOPOMOFO LET ZY
31BB..31BF  ; UNASSIGNED  # <reserved>..<reserved>
31C0..31E3  ; FREE_PVAL   # CJK STROKE T..CJK STROKE Q
31E4..31EF  ; UNASSIGNED  # <reserved>..<reserved>
31F0..31FF  ; PVALID      # KATAKANA LET SM KU..KATAKANA LET SM
3200..321E  ; FREE_PVAL   # PAREN HANGUL KIYEOK..PAREN KOREAN C
321F        ; UNASSIGNED  # <reserved>
```

```
3220..32FE  ; FREE_PVAL   # PAREN IDEO ONE..CIRCLED KATAKANA WO
32FF        ; UNASSIGNED  # <reserved>
3300..33FF  ; FREE_PVAL   # SQUARE APAATO..SQUARE GAL
3400..4DB5  ; PVALID      # <CJK Ideograph Extension A>
4DB6..4DBF  ; UNASSIGNED  # <reserved>..<reserved>
4DC0..4DFF  ; FREE_PVAL   # HEX FOR THE CREATIVE HEAVEN..HEX FO
4E00..9FCC  ; PVALID      # <CJK Ideograph>
9FCD..9FFF  ; UNASSIGNED  # <reserved>..<reserved>
A000..A48C  ; PVALID      # YI SYL IT..YI SYL YYR
A48D..A48F  ; UNASSIGNED  # <reserved>..<reserved>
A490..A4C6  ; FREE_PVAL   # YI RAD QOT..YI RAD KE
A4C7..A4CF  ; UNASSIGNED  # <reserved>..<reserved>
A4D0..A4FD  ; PVALID      # LISU LET BA..LISU LET TONE MYA JEU
A4FE..A4FF  ; FREE_PVAL   # LISU PUNCT COMMA..LISU PUNCT FUL
A500..A60C  ; PVALID      # VAI SYL EE..VAI SYL LENENER
A60D..A60F  ; FREE_PVAL   # VAI COMMA..VAI QUEST MARK
A610..A62B  ; PVALID      # VAI SYL NDOLE FA..VAI SYL NDOLE DO
A62C..A63F  ; UNASSIGNED  # <reserved>..<reserved>
A640..A66F  ; PVALID      # CYR CAP LET ZEMLYA..COMB CYR VZMET
A670..A673  ; FREE_PVAL   # COMB CYR TEN MILLIONS SIGN..SLAVON
A674..A67D  ; PVALID      # COMB CYR KAVYKA..COMB CYR PAYEROK
A67E        ; FREE_PVAL   # CYR KAVYKA
A67F..A697  ; PVALID      # CYR PAYEROK..CYR SM LET SHWE
A698..A69E  ; UNASSIGNED  # <reserved>..<reserved>
A69F..A6E5  ; PVALID      # COMB CYR LET IOTIFIED E..BAMUM LET
A6E6..A6EF  ; FREE_PVAL   # BAMUM LET MO..BAMUM LET KOGHOM
A6F0..A6F1  ; PVALID      # BAMUM COMB MARK KOQNDON..BAMUM COMB
A6F2..A6F7  ; FREE_PVAL   # BAMUM NJAEMLI..BAMUM QUEST MARK
A6F8..A6FF  ; UNASSIGNED  # <reserved>..<reserved>
A700..A716  ; FREE_PVAL   # MOD LET CHIN TONE YIN PING..MOD
A717..A71F  ; PVALID      # MOD LET DOT VERT BAR..MOD L
A720..A721  ; FREE_PVAL   # MOD LET STRESS AND HIGH TONE..MOD
A722..A76F  ; PVALID      # LAT CAP LET EGYPT ALEF..LAT SM LET
A770        ; FREE_PVAL   # MODIFIER LETTER US
A771..A788  ; PVALID      # LATIN SMALL LETTER DUM..MOD LET LOW
A789..A78A  ; FREE_PVAL   # MOD LET COLON..MOD LET SH EQUALS SI
A78B..A78E  ; PVALID      # LAT SM LET SALTILLO..LAT SM LET L W
A78F        ; UNASSIGNED  # <reserved>
A790..A793  ; PVALID      # LAT CAP LET N W DESC..LAT SM LET C
A794..A79F  ; UNASSIGNED  # <reserved>..<reserved>
A7A0..A7AA  ; PVALID      # LAT CAP LET G W OBLIQUE STROKE..LAT
A7AB..A7F7  ; UNASSIGNED  # <reserved>..<reserved>
A7F8..A7F9  ; FREE_PVAL   # MOD LET CAP H W STROKE..MOD LET SM
A7FA..A827  ; PVALID      # LAT LET SM CAP TURNED M..SYLOTI NA
A828..A82B  ; FREE_PVAL   # SYLOTI NAGRI POET MARK-1..SYLOTI NA
A82C..A82F  ; UNASSIGNED  # <reserved>..<reserved>
A830..A839  ; FREE_PVAL   # N INDIC FRACT ONE QUART..N INDIC QU
A83A..A83F  ; UNASSIGNED  # <reserved>..<reserved>
```

```
A840..A873  ; PVALID      # PHAGS-PA LET KA..PHAGS-PA LET CANDR
A874..A877  ; FREE_PVAL   # PHAGS-PA SINGLE HEAD MARK..PHAGS-PA
A878..A87F  ; UNASSIGNED  # <reserved>..<reserved>
A880..A8C4  ; PVALID      # SAUR SIGN ANUSVARA..SAUR SIGN VIRAM
A8C5..A8CD  ; UNASSIGNED  # <reserved>..<reserved>
A8CE..A8CF  ; FREE_PVAL   # SAUR DANDA..SAUR DOUBLE DANDA
A8D0..A8D9  ; PVALID      # SAUR DIG ZERO..SAUR DIG NINE
A8DA..A8DF  ; UNASSIGNED  # <reserved>..<reserved>
A8E0..A8F7  ; PVALID      # COMB DEVAN DIG ZERO..DEVAN SIGN CAN
A8F8..A8FA  ; FREE_PVAL   # DEVAN SIGN PUSHPIKA..DEVAN CARET
A8FB        ; PVALID      # DEVAN HEADSTROKE
A8FC..A8FF  ; UNASSIGNED  # <reserved>..<reserved>
A900..A92D  ; PVALID      # KAYAH LI DIG ZERO..KAYAH LI TONE CA
A92E..A92F  ; FREE_PVAL   # KAYAH LI SIGN CWI..KAYAH LI SIGN SH
A930..A953  ; PVALID      # REJANG LET KA..REJANG VIRAMA
A954..A95E  ; UNASSIGNED  # <reserved>..<reserved>
A95F        ; FREE_PVAL   # REJANG SECTION MARK
A960..A97C  ; DISALLOWED  # HANGUL CHO TIKEUT-MIUEM..HANGUL CHO
A97D..A97F  ; UNASSIGNED  # <reserved>..<reserved>
A980..A9C0  ; PVALID      # JAV SIGN PANYANGGA..JAV PANGKON
A9C1..A9CD  ; FREE_PVAL   # JAV LEFT RERENGGAN..JAV TURNED PADA
A9CE        ; UNASSIGNED  # <reserved>
A9CF..A9D9  ; PVALID      # JAV PANGRANGKEP..JAV DIG NINE
A9DA..A9DD  ; UNASSIGNED  # <reserved>..<reserved>
A9DE..A9DF  ; FREE_PVAL   # JAV PADA TIRTA TUMETES..JAV PADA I
A9E0..A9FF  ; UNASSIGNED  # <reserved>..<reserved>
AA00..AA36  ; PVALID      # CHAM LET A..CHAM CONS SIGN WA
AA37..AA3F  ; UNASSIGNED  # <reserved>..<reserved>
AA40..AA4D  ; PVALID      # CHAM LET FIN K..CHAM CONS SIGN FIN
AA4E..AA4F  ; UNASSIGNED  # <reserved>..<reserved>
AA50..AA59  ; PVALID      # CHAM DIG ZERO..CHAM DIG NINE
AA5A..AA5B  ; UNASSIGNED  # <reserved>..<reserved>
AA5C..AA5F  ; FREE_PVAL   # CHAM PUNCT SPIRAL..CHAM PUNCT TR
AA60..AA76  ; PVALID      # MYAN LET KHAMTI GA..MYAN LOGOGRAM K
AA77..AA79  ; FREE_PVAL   # MYAN SYM AITON EXCLAM..MYAN SYM AIT
AA7A..AA7B  ; PVALID      # MYAN LET AITON RA..MYAN SIGN PAO KA
AA7C..AA7F  ; UNASSIGNED  # <reserved>..<reserved>
AA80..AAC2  ; PVALID      # TAI VIET LET LOW KO..TAI VIET TONE
AAC3..AADA  ; UNASSIGNED  # <reserved>..<reserved>
AADB..AADD  ; PVALID      # TAI VIET SYM KON..TAI VIET SYM SAM
AADE..AADF  ; FREE_PVAL   # TAI VIET SYM HO HOI..TAI VIET SYM K
AAE0..AAEF  ; PVALID      # MEETEI MAYEK LET E..MEETEI MAYEK VO
AAF0..AAF1  ; FREE_PVAL   # MEETEI MAYEK CHEIKHAN..MEETEI MAYEK
AAF2..AAF6  ; PVALID      # MEETEI MAYEK ANJI..MEETEI MAYEK VIR
AAF7..AB00  ; UNASSIGNED  # <reserved>..<reserved>
AB01..AB06  ; PVALID      # ETHI SYL TTHU..ETHI SYL TTHO
AB07..AB08  ; UNASSIGNED  # <reserved>..<reserved>
AB09..AB0E  ; PVALID      # ETHI SYL DDHAA..ETHI SYL DDHO
```

```
   AB0F..AB10  ; UNASSIGNED  # <reserved>..<reserved>
   AB11..AB16  ; PVALID      # ETHI SYL DZU..ETHI SYL DZO
   AB17..AB1F  ; UNASSIGNED  # <reserved>..<reserved>
   AB20..AB26  ; PVALID      # ETHI SYL CCHHA..ETHI SYL CCHHO
   AB27        ; UNASSIGNED  # <reserved>..<reserved>
   AB28..AB2E  ; PVALID      # ETHI SYL BBAA..ETHI SYL BBO
   AB2F..ABBF  ; UNASSIGNED  # <reserved>..<reserved>
   ABC0..ABEA  ; PVALID      # MEETEI MAYEK LET KOK..MEETEI MAYEK
   ABEB        ; FREE_PVAL   # MEETEI MAYEK CHEIKHEI
   ABEC..ABED  ; PVALID      # MEETEI MAYEK LUM IYEK..MEETEI MAYEK
   ABEE..ABEF  ; UNASSIGNED  # <reserved>..<reserved>
   ABF0..ABF9  ; PVALID      # MEETEI MAYEK DIG ZERO..MEETEI MAYEK
   ABFA..ABFF  ; UNASSIGNED  # <reserved>..<reserved>
   AC00..D7A3  ; PVALID      # <Hangul Syllable>
   D7A4..D7AF  ; UNASSIGNED  # <reserved>..<reserved>
   D7B0..D7C6  ; DISALLOWED  # HANGUL JUNG O-YEO..HANGUL JUNG ARAE
   D7C7..D7CA  ; UNASSIGNED  # <reserved>..<reserved>
   D7CB..D7FB  ; DISALLOWED  # HANGUL JONG NIEUN-RIEUL..HANGUL JON
   D7FC..D7FF  ; UNASSIGNED  # <reserved>..<reserved>
   D800..F8FF  ; DISALLOWED  # <Non Private Use High Surrogate>
   F900..FA6D  ; PVALID      # CJK COMP IDEO-F900..CJK COMP IDEO
   FA6E..FA6F  ; UNASSIGNED  # <reserved>..<reserved>
   FA70..FAD9  ; PVALID      # CJK COMP IDEO-FA70..CJK COMP IDEO
   FADA..FAFF  ; UNASSIGNED  # <reserved>..<reserved>
   FB00..FB06  ; FREE_PVAL   # LAT SM LIG FF..LAT SM LIG ST
   FB07..FB12  ; UNASSIGNED  # <reserved>..<reserved>
   FB13..FB17  ; FREE_PVAL   # ARMENIAN SM LIG MEN NOW..ARMENIAN SM
   FB18..FB1C  ; UNASSIGNED  # <reserved>..<reserved>
   FB1D..FB1F  ; PVALID      # HEBR LET YOD W HIRIQ..HEBR LIG YID Y
   FB20..FB29  ; FREE_PVAL   # HEBR LET ALT AYIN..HEB LET ALT PLUS
   FB2A..FB36  ; PVALID      # HEBR LET SHIN W SHIN DOT..HEBR LET Z
   FB37        ; UNASSIGNED  # <reserved>
   FB38..FB3C  ; PVALID      # HEBR LET TET W DAGESH..HEBR LET
   FB3D        ; UNASSIGNED  # <reserved>
   FB3E        ; PVALID      # HEBR LET MEM W DAGESH
   FB3F        ; UNASSIGNED  # <reserved>
   FB40..FB41  ; PVALID      # HEBR LET NUN W DAGESH..HEBR LET
   FB42        ; UNASSIGNED  # <reserved>
   FB43..FB44  ; PVALID      # HEBR LET FIN PE W DAGESH..HEBR L
   FB45        ; UNASSIGNED  # <reserved>
   FB46..FB4E  ; PVALID      # HEBR LET TSADI W DAGESH..HEBR LET P
   FB4F..FBC1  ; FREE_PVAL   # HEBR LIG ALEF LAMED..ARAB SYM S
   FBC2..FBD2  ; UNASSIGNED  # <reserved>..<reserved>
   FBD3..FD3F  ; FREE_PVAL   # ARAB LET NG ISO FORM..ORNATE RIGHT
   FD40..FD4F  ; UNASSIGNED  # <reserved>..<reserved>
   FD50..FD8F  ; FREE_PVAL   # ARAB LIG TEH W JEEM W MEEM INIT
   FD90..FD91  ; UNASSIGNED  # <reserved>..<reserved>
   FD92..FDC7  ; FREE_PVAL   # ARAB LIG MEEM W JEEM W KHAH INI
```

```
   FDC8..FDCF  ; UNASSIGNED  # <reserved>..<reserved>
   FDD0..FDEF  ; DISALLOWED  # <noncharacter>..<noncharacter>
   FDF0..FDFD  ; FREE_PVAL   # ARAB LIG SALLA USED..ARAB LIG BISMI
   FDFE..FDFF  ; UNASSIGNED  # <reserved>..<reserved>
   FE00..FE0F  ; DISALLOWED  # VAR SEL-1..VAR SEL-16
   FE10..FE19  ; FREE_PVAL   # PRES FORM FOR VERT COMMA..PRES FORM
   FE1A..FE1F  ; UNASSIGNED  # <reserved>..<reserved>
   FE20..FE26  ; PVALID      # COMB LIG LEFT HALF..COMB CONJ MACRO
   FE27..FE2F  ; UNASSIGNED  # <reserved>..<reserved>
   FE30..FE52  ; FREE_PVAL   # PRES FORM FOR VERT TWO DOT LEAD..SM
   FE53        ; UNASSIGNED  # <reserved>
   FE54..FE66  ; FREE_PVAL   # SM SEMICOLON..SM EQUALS SIGN
   FE67        ; UNASSIGNED  # <reserved>
   FE68..FE6B  ; FREE_PVAL   # SM REV SOLIDUS..SM COMM AT
   FE6C..FE6F  ; UNASSIGNED  # <reserved>..<reserved>
   FE70..FE72  ; FREE_PVAL   # ARAB FATHATAN ISO FORM..ARAB DAMMAT
   FE73        ; PVALID      # ARAB TAIL FRAGMENT
   FE74        ; FREE_PVAL   # ARAB KASRATAN ISO FORM
   FE75        ; UNASSIGNED  # <reserved>
   FE76..FEFC  ; FREE_PVAL   # ARAB FATHA ISO FORM..ARAB LIG LAM W
   FEFD..FEFE  ; UNASSIGNED  # <reserved>..<reserved>
   FEFF        ; DISALLOWED  # ZERO WIDTH NO-BREAK SPACE
   FF00        ; UNASSIGNED  # <reserved>
   FF01..FF9F  ; FREE_PVAL   # FULLW EXCLAM MARK..HALFW KATA SE
   FFA0        ; DISALLOWED  # HALFW HANGUL FILLER
   FFA1..FFBE  ; FREE_PVAL   # HALFW HANGUL LET KIYEOK..HALFW H
   FFBF..FFC1  ; UNASSIGNED  # <reserved>..<reserved>
   FFC2..FFC7  ; FREE_PVAL   # HALFW HANGUL LET A..HALFW HANGUL
   FFC8..FFC9  ; UNASSIGNED  # <reserved>..<reserved>
   FFCA..FFCF  ; FREE_PVAL   # HALFW HANGUL LET YEO..HALFW HANGU
   FFD0..FFD1  ; UNASSIGNED  # <reserved>..<reserved>
   FFD2..FFD7  ; FREE_PVAL   # HALFW HANGUL LET YO..HALFW HANGUL
   FFD8..FFD9  ; UNASSIGNED  # <reserved>..<reserved>
   FFDA..FFDC  ; FREE_PVAL   # HALFW HANGUL LET EU..HALFW HANGUL
   FFDD..FFDF  ; UNASSIGNED  # <reserved>..<reserved>
   FFE0..FFE6  ; FREE_PVAL   # FULLW CENT SIGN..FULLW WON SIGN
   FFE7        ; UNASSIGNED  # <reserved>
   FFE8..FFEE  ; FREE_PVAL   # HALFW FORMS LIGHT VERT..HALFW WH
   FFEF..FFF8  ; UNASSIGNED  # <reserved>..<reserved>
   FFF9..FFFB  ; DISALLOWED  # INTERL ANNO ANCHOR..INTERL ANNO TER
   FFFC..FFFD  ; FREE_PVAL   # OBJECT REPL CHAR..REPL CHAR
   FFFE..FFFF  ; DISALLOWED  # <noncharacter>..<noncharacter>
   10000..1000B; PVALID      # LIN B SYL B008 A..LIN B SYL
   1000C       ; UNASSIGNED  # <reserved>
   1000D..10026; PVALID      # LIN B SYL B036 JO..LIN B SYL
   10027       ; UNASSIGNED  # <reserved>
   10028..1003A; PVALID      # LIN B SYL B060 RA..LIN B SYL
   1003B       ; UNASSIGNED  # <reserved>
```

```
1003C..1003D; PVALID      # LIN B SYL B017 ZA..LIN B SYL
1003E       ; UNASSIGNED  # <reserved>
1003F..1004D; PVALID      # LIN B SYL B020 ZO..LIN B SYL
1004E..1004F; UNASSIGNED  # <reserved>..<reserved>
10050..1005D; PVALID      # LIN B SYM B018..LIN B SYM B089
1005E..1007F; UNASSIGNED  # <reserved>..<reserved>
10080..100FA; PVALID      # LIN B IDEO B100 MAN..LIN B IDEO
100FB..100FF; UNASSIGNED  # <reserved>..<reserved>
10100..10102; FREE_PVAL   # AEG WORD SEP LINE..AEG CHECK MAR
10103..10106; UNASSIGNED  # <reserved>..<reserved>
10107..10133; FREE_PVAL   # AEG NUM ONE..AEG NUM NINETY THOU
10134..10136; UNASSIGNED  # <reserved>..<reserved>
10137..1018A; FREE_PVAL   # AEG WEIGHT BASE UNIT..GREEK ZERO SI
1018B..1018F; UNASSIGNED  # <reserved>..<reserved>
10190..1019B; FREE_PVAL   # ROM SEXTANS SIGN..ROM CENTURIAL SIG
1019C..101CF; UNASSIGNED  # <reserved>..<reserved>
101D0..101FC; FREE_PVAL   # PHAISTOS DISC SIGN PED..PHAISTOS DI
101FD       ; PVALID      # PHAISTOS DISC SIGN COMB OBLIQUE STR
101FE..1027F; UNASSIGNED  # <reserved>..<reserved>
10280..1029C; PVALID      # LYCIAN LET A..LYCIAN LET X
1029D..1029F; UNASSIGNED  # <reserved>..<reserved>
102A0..102D0; PVALID      # CARIAN LET A..CARIAN LET UUU3
102D1..102FF; UNASSIGNED  # <reserved>..<reserved>
10300..1031E; PVALID      # OLD ITAL LET A..OLD ITAL LET UU
1031F       ; UNASSIGNED  # <reserved>
10320..10323; FREE_PVAL   # OLD ITAL NUM ONE..OLD ITAL NUM F
10324..1032F; UNASSIGNED  # <reserved>..<reserved>
10330..10340; PVALID      # GOTH LET AHSA..GOTH LET PAIRTHRA
10341       ; FREE_PVAL   # GOTH LET NINETY
10342..10349; PVALID      # GOTH LET RAIDA..GOTH LET OTHAL
1034A       ; FREE_PVAL   # GOTH LET NINE HUNDRED
1034B..1037F; UNASSIGNED  # <reserved>..<reserved>
10380..1039D; PVALID      # UGAR LET ALPA..UGAR LET SSU
1039E       ; UNASSIGNED  # <reserved>
1039F       ; FREE_PVAL   # UGAR WORD DIVIDER
103A0..103C3; PVALID      # OLD PERS SIGN A..OLD PERS SIGN HA
103C4..103C7; UNASSIGNED  # <reserved>..<reserved>
103C8..103CF; PVALID      # OLD PERS SIGN AURAMAZDAA..OLD PERS
103D0..103D5; FREE_PVAL   # OLD PERS WORD DIVIDER..OLD PERS NUM
103D6..103FF; UNASSIGNED  # <reserved>..<reserved>
10400..1049D; PVALID      # DESERET CAP LET LONG I..OSMANYA LET
1049E..1049F; UNASSIGNED  # <reserved>..<reserved>
104A0..104A9; PVALID      # OSMANYA DIG ZERO..OSMANYA DIG NINE
104AA..107FF; UNASSIGNED  # <reserved>..<reserved>
10800..10805; PVALID      # CYPRIOT SYL A..CYPRIOT SYL JA
10806..10807; UNASSIGNED  # <reserved>..<reserved>
10808       ; PVALID      # CYPRIOT SYL JO
10809       ; UNASSIGNED  # <reserved>
```

```
   1080A..10835; PVALID      # CYPRIOT SYL KA..CYPRIOT SYL WO
   10836       ; UNASSIGNED  # <reserved>
   10837..10838; PVALID      # CYPRIOT SYL XA..CYPRIOT SYL XE
   10839..1083B; UNASSIGNED  # <reserved>..<reserved>
   1083C       ; PVALID      # CYPRIOT SYL ZA
   1083D..1083E; UNASSIGNED  # <reserved>..<reserved>
   1083F..10855; PVALID      # CYPRIOT SYL ZO..IMP ARAM LET TAW
   10856       ; UNASSIGNED  # <reserved>
   10857..1085F; FREE_PVAL   # IMP ARAM SECT SIGN..IMP ARAM
   10860..108FF; UNASSIGNED  # <reserved>..<reserved>
   10900..10915; PVALID      # PHOEN LET ALF..PHOEN LET TAU
   10916..1091B; FREE_PVAL   # PHOEN NUM ONE..PHOEN NUM THR
   1091C..1091E; UNASSIGNED  # <reserved>..<reserved>
   1091F       ; FREE_PVAL   # PHOEN WORD SEP
   10920..10939; PVALID      # LYDIAN LET A..LYDIAN LET C
   1093A..1093E; UNASSIGNED  # <reserved>..<reserved>
   1093F       ; FREE_PVAL   # LYDIAN TRIANGULAR MARK
   10940..1097F; UNASSIGNED  # <reserved>..<reserved>
   10980..109B7; PVALID      # MERO HIER LET A..MERO CURS LET
   109B8..109BD; UNASSIGNED  # <reserved>..<reserved>
   109BE..109BF; PVALID      # MERO CURS LOG RMT..MERO CURS L
   109C0..109FF; UNASSIGNED  # <reserved>..<reserved>
   10A00..10A03; PVALID      # KHARO LET A..KHARO VOW SIGN V
   10A04       ; UNASSIGNED  # <reserved>
   10A05..10A06; PVALID      # KHARO VOW SIGN E..KHARO VOW SI
   10A07..10A0B; UNASSIGNED  # <reserved>..<reserved>
   10A0C..10A13; PVALID      # KHARO VOW LEN MARK..KHARO LET
   10A14       ; UNASSIGNED  # <reserved>
   10A15..10A17; PVALID      # KHARO LET CA..KHARO LET JA
   10A18       ; UNASSIGNED  # <reserved>
   10A19..10A33; PVALID      # KHARO LET NYA..KHARO LET TTT
   10A34..10A37; UNASSIGNED  # <reserved>..<reserved>
   10A38..10A3A; PVALID      # KHARO SIGN BAR ABOVE..KHARO SIGN D
   10A3B..10A3E; UNASSIGNED  # <reserved>..<reserved>
   10A3F       ; PVALID      # KHARO VIRAMA
   10A40..10A47; FREE_PVAL   # KHARO DIG ONE..KHARO NUM ONE
   10A48..10A4F; UNASSIGNED  # <reserved>..<reserved>
   10A50..10A58; FREE_PVAL   # KHARO PUNCT DOT..KHARO PUNCT
   10A59..10A5F; UNASSIGNED  # <reserved>..<reserved>
   10A60..10A7C; PVALID      # OLD S ARAB LET HE..OLD SOUTH ARAB
   10A7D..10A7F; FREE_PVAL   # OLD S ARAB NUM ONE..OLD SOUTH ARAB
   10A80..10AFF; UNASSIGNED  # <reserved>..<reserved>
   10B00..10B35; PVALID      # AVESTAN LET A..AVESTAN LET HE
   10B36..10B38; UNASSIGNED  # <reserved>..<reserved>
   10B39..10B3F; FREE_PVAL   # AVESTAN ABBR MARK..LARGE ONE RING O
   10B40..10B55; PVALID      # INSCRIPT PARTHIAN LET ALEPH..INSCRI
   10B56..10B57; UNASSIGNED  # <reserved>..<reserved>
   10B58..10B5F; FREE_PVAL   # INSCRIPT PARTHIAN NUM ONE..INSCRIPT
```

```
10B60..10B72; PVALID      # INSCRIPT PAHLAVI LET ALEPH..INSCRIP
10B73..10B77; UNASSIGNED  # <reserved>..<reserved>
10B78..10B7F; FREE_PVAL   # INSCRIPT PAHLAVI NUM ONE..INSCRIPT
10B80..10BFF; UNASSIGNED  # <reserved>..<reserved>
10C00..10C48; PVALID      # OLD TURK LET ORKHON A..OLD TURK LET
10C49..10E5F; UNASSIGNED  # <reserved>..<reserved>
10E60..10E7E; FREE_PVAL   # RUMI DIG ONE..RUMI FRACTION TWO THI
10E7F..10FFF; UNASSIGNED  # <reserved>..<reserved>
11000..11046; PVALID      # BRAHMI SIGN CANDRABINDU..BRAHMI VIR
11047..1104D; FREE_PVAL   # BRAHMI DANDA..BRAHMI PUNCT LOTUS
1104E..11051; UNASSIGNED  # <reserved>..<reserved>
11052..11065; FREE_PVAL   # BRAHMI NUM ONE..BRAHMI NUM ONE THOU
11066..1106F; PVALID      # BRAHMI DIG ZERO..BRAHMI DIG NINE
11070..1107F; UNASSIGNED  # <reserved>..<reserved>
11080..110BA; PVALID      # KAITHI SIGN CANDRABINDU..KAITHI SIG
110BB..110BC; FREE_PVAL   # KAITHI ABBR SIGN..KAITHI ENUM SIGN
110BD       ; DISALLOWED  # KAITHI NUM SIGN
110BE..110C1; FREE_PVAL   # KAITHI SECT MARK..KAITHI DOUBLE DAN
110C2..110CF; UNASSIGNED  # <reserved>..<reserved>
110D0..110F8; PVALID      # SORA SOMPENG LETTER SAH..SORA SOMPE
110F9..110EF; UNASSIGNED  # <reserved>..<reserved>
110F0..110F9; PVALID      # SORA SOMPENG DIG ZERO..SORA SOMPENG DI
110FA..110FF; UNASSIGNED  # <reserved>..<reserved>
11100..11134; PVALID      # CHAKMA SIGN CANDRABINDU..CHAKMA MAAYY
11135       ; UNASSIGNED  # <reserved>
11136..1113F; PVALID      # CHAKMA DIG ZERO..CHAKMA DIG NINE
11140..11143; FREE_PVAL   # CHAKMA SECT MARK..CHAKMA QUEST MARK
11144..1117F; UNASSIGNED  # <reserved>..<reserved>
11180..111C4; PVALID      # SHARADA SIGN CANDRABINDU..SHARADA OM
111C5..111C8; FREE_PVAL   # SHARADA DANDA..SHARADA SEPARATOR
111C9..111CF; UNASSIGNED  # <reserved>..<reserved>
111D0..111D9; PVALID      # SHARADA DIG ZERO..SHARADA DIG NINE
111DA..1167F; UNASSIGNED  # <reserved>..<reserved>
11680..116B7; PVALID      # TAKRI LET A..TAKRI SIGN NUKTA
116B8..116BF; UNASSIGNED  # <reserved>..<reserved>
116C0..116C9; PVALID      # TAKRI DIGIT ZERO..TAKRI DIG NINE
116CA..1FFFF; UNASSIGNED  # <reserved>..<reserved>
12000..1236E; PVALID      # CUNEI SIGN A..CUNEI SIGN ZUM
1236F..123FF; UNASSIGNED  # <reserved>..<reserved>
12400..12462; FREE_PVAL   # CUNEI NUM SIGN TWO ASH..CUNEI NUM
12463..1246F; UNASSIGNED  # <reserved>..<reserved>
12470..12473; FREE_PVAL   # CUNEI PUNCT SIGN OLD ASSYRIAN WORD
12474..12FFF; UNASSIGNED  # <reserved>..<reserved>
13000..1342E; PVALID      # EGYPT HIERO A001..EGYPT HIERO AA032
1342F..167FF; UNASSIGNED  # <reserved>..<reserved>
16800..16A38; PVALID      # BAMUM LET PHASE-A NGKUE MFON..BAMUN LE
16A39..16EFF; UNASSIGNED  # <reserved>..<reserved>
16F00..16F44; PVALID      # MIAO LET PA..MIAO LET HHA
```

```
16F45..16F4F; UNASSIGNED  # <reserved>..<reserved>
16F50..16F7E; PVALID      # MIAO LET NAS..MIAO VOWEL SIGN NG
16F7F..16F8E; UNASSIGNED  # <reserved>..<reserved>
16F8F..16F9F; PVALID      # MIAO TONE RIGHT..MIAO LET REF TON
16FA0..1AFFF; UNASSIGNED  # <reserved>..<reserved>
1B000..1B001; PVALID      # KATA LET ARCH E..KATA LET ARCH YE
1B002..1CFFF; UNASSIGNED  # <reserved>..<reserved>
1D000..1D0F5; FREE_PVAL   # BYZ MUS SYM PSILI..BYZ MUS
1D0F6..1D0FF; UNASSIGNED  # <reserved>..<reserved>
1D100..1D126; FREE_PVAL   # MUS SYM SINGLE BARLINE..MUS SYMBOL
1D127..1D128; UNASSIGNED  # <reserved>..<reserved>
1D129..1D164; FREE_PVAL   # MUS SYM MULT MEASURE REST..MUS SYM ONE
1D165..1D169; PVALID      # MUS SYM COMB STEM..MUS SYM COMB TREMOL
1D16A..1D16C; FREE_PVAL   # MUS SYM FING TREM-1..MUS SYM FING TREM
1D16D..1D172; PVALID      # MUS SYM COMB AUG DOT..MUS SYM COMB FL
1D173..1D17A; DISALLOWED  # MUS SYM BEGIN BEAM..MUS SYM END PHRASE
1D17B..1D182; PVALID      # MUS SYM COMB ACCENT..MUS SYM COMB LOUR
1D183..1D184; FREE_PVAL   # MUS SYM ARP UP..MUS SYM ARP DOWN
1D185..1D18B; PVALID      # MUS SYM COMB DOIT..MUS SYM COMB TRIPLE
1D18C..1D1A9; FREE_PVAL   # MUS SYM RINFORZANDO..MUS SYM DEG SLASH
1D1AA..1D1AD; PVALID      # MUS SYM COMB DOWN BOW..MUS SYM COMB SN
1D1AE..1D1DD; FREE_PVAL   # MUS SYM PEDAL MARK..MUS SYM PES SUBPUN
1D1DE..1D1FF; UNASSIGNED  # <reserved>..<reserved>
1D200..1D241; FREE_PVAL   # GREEK VOCAL NOTATION SYM-1..GREEK INS
1D242..1D244; FREE_PVAL   # COMB GREEK MUS TRISEME..COMB GREEK MU
1D245       ; FREE_PVAL   # GREEK MUSICAL LEIMMA
1D246..1D2FF; UNASSIGNED  # <reserved>..<reserved>
1D300..1D356; DISALLOWED  # MONOG FOR EARTH..TETRAG FOR FOSTERING
1D357..1D35F; UNASSIGNED  # <reserved>..<reserved>
1D360..1D371; DISALLOWED  # COUNT ROD UNIT DIG ONE..COUNT ROD TE
1D372..1D3FF; UNASSIGNED  # <reserved>..<reserved>
1D400..1D454; FREE_PVAL   # MATH BOLD CAP A..MATH IT
1D455       ; UNASSIGNED  # <reserved>
1D456..1D49C; FREE_PVAL   # MATH ITAL SM I..MATH SC
1D49D       ; UNASSIGNED  # <reserved>
1D49E..1D49F; FREE_PVAL   # MATH SCRIPT CAP C..MATH
1D4A0..1D4A1; UNASSIGNED  # <reserved>..<reserved>
1D4A2       ; FREE_PVAL   # MATH SCRIPT CAP G
1D4A3..1D4A4; UNASSIGNED  # <reserved>..<reserved>
1D4A5..1D4A6; FREE_PVAL   # MATH SCRIPT CAP J..MATH
1D4A7..1D4A8; UNASSIGNED  # <reserved>..<reserved>
1D4A9..1D4AC; FREE_PVAL   # MATH SCRIPT CAP N..MATH
1D4AD       ; UNASSIGNED  # <reserved>
1D4AE..1D4B9; FREE_PVAL   # MATH SCRIPT CAP S..MATH
1D4BA       ; UNASSIGNED  # <reserved>
1D4BB       ; FREE_PVAL   # MATH SCRIPT SM F
1D4BC       ; UNASSIGNED  # <reserved>
1D4BD..1D4C3; FREE_PVAL   # MATH SCRIPT SM H..MATH SC
```

```
   1D4C4       ; UNASSIGNED  # <reserved>
   1D4C5..1D505; FREE_PVAL   # MATH SCRIPT SM P..MATH FR
   1D506       ; UNASSIGNED  # <reserved>
   1D507..1D50A; FREE_PVAL   # MATH FRAKTUR CAP D..MATH
   1D50B..1D50C; UNASSIGNED  # <reserved>..<reserved>
   1D50D..1D514; FREE_PVAL   # MATH FRAKTUR CAP J..MATH
   1D515       ; UNASSIGNED  # <reserved>
   1D516..1D51C; FREE_PVAL   # MATH FRAKTUR CAP S..MATH
   1D51D       ; UNASSIGNED  # <reserved>
   1D51E..1D539; FREE_PVAL   # MATH FRAKTUR SM A..MATH D
   1D53A       ; UNASSIGNED  # <reserved>
   1D53B..1D53E; FREE_PVAL   # MATH DOUBLE-STRUCK CAP D..MATHEM
   1D53F       ; UNASSIGNED  # <reserved>
   1D540..1D544; FREE_PVAL   # MATH DOUBLE-STRUCK CAP I..MATHEM
   1D545       ; UNASSIGNED  # <reserved>
   1D546       ; FREE_PVAL   # MATH DOUBLE-STRUCK CAP O
   1D547..1D549; UNASSIGNED  # <reserved>..<reserved>
   1D54A..1D550; FREE_PVAL   # MATH DOUBLE-STRUCK CAP S..MATHEM
   1D551       ; UNASSIGNED  # <reserved>
   1D552..1D6A5; FREE_PVAL   # MATH DOUBLE-STRUCK SM A..MATHEMAT
   1D6A6..1D6A7; UNASSIGNED  # <reserved>..<reserved>
   1D6A8..1D7CB; FREE_PVAL   # MATH BOLD CAP ALPHA..MATHEMATICA
   1D7CC..1D7CD; UNASSIGNED  # <reserved>..<reserved>
   1D7CE..1D7FF; FREE_PVAL   # MATH BOLD DIG ZERO..MATH M
   1D800..1EDFF; UNASSIGNED  # <reserved>..<reserved>
   1EE00..1EE03; FREE_PVAL   # ARAB MATH ALEF..ARAB MATH DAL
   1EE04       ; UNASSIGNED  # <reserved>
   1EE05..1EE1F; FREE_PVAL   # ARAB MATH WAW..ARAB MATH DOTLESS QAF
   1EE20       ; UNASSIGNED  # <reserved>
   1EE21..1EE22; FREE_PVAL   # ARAB MATH INIT BEH..ARAB MATH INIT JEE
   1EE23       ; UNASSIGNED  # <reserved>
   1EE24       ; FREE_PVAL   # ARAB MATH INIT HEH
   1EE25..1EE26; UNASSIGNED  # <reserved>..<reserved>
   1EE27       ; FREE_PVAL   # ARAB MATH INIT HAH
   1EE28       ; UNASSIGNED  # <reserved>
   1EE29..1EE32; FREE_PVAL   # ARAB MATH INIT YEH..ARAB MATH INIT QAF
   1EE33       ; UNASSIGNED  # <reserved>
   1EE34..1EE37; FREE_PVAL   # ARAB MATH INIT SHEEN..ARAB MATH INITIA
   1EE38       ; UNASSIGNED  # <reserved>
   1EE39       ; FREE_PVAL   # ARAB MATH INIT SHEEN
   1EE3A       ; UNASSIGNED  # <reserved>
   1EE3B       ; FREE_PVAL   # ARAB MATH INIT GHAIN
   1EE3C..1EE41; UNASSIGNED  # <reserved>..<reserved>
   1EE42       ; FREE_PVAL   # ARAB MATH TAILED JEEM
   1EE43..1EE46; UNASSIGNED  # <reserved>..<reserved>
   1EE47       ; FREE_PVAL   # ARAB MATH TAILED HAH
   1EE48       ; UNASSIGNED  # <reserved>
   1EE49       ; FREE_PVAL   # ARAB MATH TAILED YEH
```

```
   1EE4A       ; UNASSIGNED  # <reserved>
   1EE4B       ; FREE_PVAL   # ARAB MATH TAILED LAM
   1EE4C       ; UNASSIGNED  # <reserved>
   1EE4D..1EE4F; FREE_PVAL   # ARAB MATH TAILED NOON..ARAB MATH TAILE
   1EE50       ; UNASSIGNED  # <reserved>
   1EE51..1EE52; FREE_PVAL   # ARAB MATH TAILED QAF..ARAB MATH TAILED
   1EE53       ; UNASSIGNED  # <reserved>
   1EE54       ; FREE_PVAL   # ARAB MATH TAILED SHEEN
   1EE55..1EE56; UNASSIGNED  # <reserved>..<reserved>
   1EE57       ; FREE_PVAL   # ARAB MATH TAILED KHAH
   1EE58       ; UNASSIGNED  # <reserved>
   1EE59       ; FREE_PVAL   # ARAB MATH TAILED DAD
   1EE5A       ; UNASSIGNED  # <reserved>
   1EE5B       ; FREE_PVAL   # ARAB MATH TAILED GHAIN
   1EE5C       ; UNASSIGNED  # <reserved>
   1EE5D       ; FREE_PVAL   # ARAB MATH TAILED DOTLESS NOON
   1EE5E       ; UNASSIGNED  # <reserved>
   1EE5F       ; FREE_PVAL   # ARAB MATH TAILED DOTLESS GHAIN
   1EE60       ; UNASSIGNED  # <reserved>
   1EE61..1EE62; FREE_PVAL   # ARAB MATH STRETCHED BEH..ARAB MATH STR
   1EE63       ; UNASSIGNED  # <reserved>
   1EE64       ; FREE_PVAL   # ARAB MATH STRETCHED HEH
   1EE65..1EE66; UNASSIGNED  # <reserved>..<reserved>
   1EE67..1EE6A; FREE_PVAL   # ARAB MATH STRETCHED HAH..ARAB MATH STR
   1EE6B       ; UNASSIGNED  # <reserved>
   1EE6C..1EE72; FREE_PVAL   # ARAB MATH STRETCHED MEEM..ARAB MATH ST
   1EE73       ; UNASSIGNED  # <reserved>
   1EE74..1EE77; FREE_PVAL   # ARAB MATH STRETCHED SHEEN..ARAB MATH S
   1EE78       ; UNASSIGNED  # <reserved>
   1EE79..1EE7C; FREE_PVAL   # ARAB MATH STRETCHED DAD..ARAB MATH STR
   1EE7D       ; UNASSIGNED  # <reserved>
   1EE7E       ; FREE_PVAL   # ARAB MATH STRETCHED DOTLESS FEH
   1EE7F       ; UNASSIGNED  # <reserved>
   1EE80..1EE89; FREE_PVAL   # ARAB MATH LOOPED ALEF..ARAB MATH LOOPE
   1EE8A       ; UNASSIGNED  # <reserved>
   1EE8B..1EE9B; FREE_PVAL   # ARAB MATH LOOPED LAM..ARAB MATH LOOPED
   1EE9C..1EEA0; UNASSIGNED  # <reserved>..<reserved>
   1EEA1..1EEA3; FREE_PVAL   # ARAB MATH DOUBLE-STRUCK BEH..ARAB MATH
   1EEA4       ; UNASSIGNED  # <reserved>
   1EEA5..1EEA9; FREE_PVAL   # ARAB MATH DOUBLE-STRUCK WAW..ARAB MATH
   1EEAA       ; UNASSIGNED  # <reserved>
   1EEAB..1EEBB; FREE_PVAL   # ARAB MATH DOUBLE-STRUCK LAM..ARAB MATH
   1EEBC..1EEEF; UNASSIGNED  # <reserved>..<reserved>
   1EEF0..1EEF1; FREE_PVAL   # ARAB MATH OP MEEM W HAH W TATWHEEL..AR
   1EEF2..1EFFF; UNASSIGNED  # <reserved>..<reserved>
   1F000..1F02B; FREE_PVAL   # MAHJONG TILE EAST WIND..MAHJONG TILE B
   1F02C..1F02F; UNASSIGNED  # <reserved>..<reserved>
   1F030..1F093; FREE_PVAL   # DOMINO TILE HORIZ BACK..DOMINO TILE VE
```

```
   1F094..1F09F; UNASSIGNED  # <reserved>..<reserved>
   1F0A0..1F0AE; FREE_PVAL   # PLAY CARD BACK..PLAY CARD KING OF SPAD
   1F0AF..1F0B0; UNASSIGNED  # <reserved>..<reserved>
   1F0B1..1F0BE; FREE_PVAL   # PLAY CARD ACE OF HEARTS..PLAY CARD KIN
   1F0BF..1F0C0; UNASSIGNED  # <reserved>..<reserved>
   1F0C1..1F0CF; FREE_PVAL   # PLAY CARD ACE OF DIAMONDS..PLAY CARD B
   1F0D0       ; UNASSIGNED  # <reserved>
   1F0D1..1F0DF; FREE_PVAL   # PLAY CARD ACE OF CLUBS..PLAY CARD WHIT
   1F0E0..1F0FF; UNASSIGNED  # <reserved>..<reserved>
   1F100..1F10A; FREE_PVAL   # DIG ZERO FULL STOP..DIG NINE COMMA
   1F10B..1F10F; UNASSIGNED  # <reserved>..<reserved>
   1F110..1F12E; FREE_PVAL   # PARENTHESIZED LAT CAP LET A..CIRCLE
   1F12F       ; UNASSIGNED  # <reserved>
   1F130..1F16B; FREE_PVAL   # SQUARED LAT CAP LET A..RAISED MD SIGN
   1F16C..1F16F; UNASSIGNED  # <reserved>..<reserved>
   1F170..1F19A; FREE_PVAL   # NEG SQ LAT CAP LET A..SQUARED VS
   1F19B..1F1E5; UNASSIGNED  # <reserved>..<reserved>
   1F1E6..1F202; FREE_PVAL   # REG IND SYMB LET A..SQ KATAKANA SA
   1F203..1F20F; UNASSIGNED  # <reserved>..<reserved>
   1F210..1F23A; FREE_PVAL   # SQ CJK UNIF IDEO-624B..SQ CJK UNIF IDE
   1F23B..1F23F; UNASSIGNED  # <reserved>..<reserved>
   1F240..1F248; FREE_PVAL   # TORT SH BRACK CJK UNIF IDEO-672C..TORT
   1F249..1F24F; UNASSIGNED  # <reserved>..<reserved>
   1F250..1F251; FREE_PVAL   # CIRC IDEO ADVANTAGE..CIRC IDEO ACCEPT
   1F252..1F2FF; UNASSIGNED  # <reserved>..<reserved>
   1F300..1F320; FREE_PVAL   # CYCLONE..SHOOTING STAR
   1F321..1F32F; UNASSIGNED  # <reserved>..<reserved>
   1F330..1F335; FREE_PVAL   # CHESTNUT..CACTUS
   1F336       ; UNASSIGNED  # <reserved>
   1F337..1F37C; FREE_PVAL   # TULIP..BABY BOTTLE
   1F37D..1F37F; UNASSIGNED  # <reserved>..<reserved>
   1F380..1F393; FREE_PVAL   # RIBBON..GRADUATION CAP
   1F394..1F39F; UNASSIGNED  # <reserved>..<reserved>
   1F3A0..1F3C4; FREE_PVAL   # CAROUSEL HORSE..SURFER
   1F3C5       ; UNASSIGNED  # <reserved>
   1F3C6..1F3CA; FREE_PVAL   # TROPHY..SWIMMER
   1F3CB..1F3DF; UNASSIGNED  # <reserved>..<reserved>
   1F3E0..1F3F0; FREE_PVAL   # HOUSE BUILDING..EUROPEAN CASTLE
   1F3F1..1F3FF; UNASSIGNED  # <reserved>..<reserved>
   1F400..1F43E; FREE_PVAL   # RAT..PAW PRINTS
   1F43F       ; UNASSIGNED  # <reserved>
   1F440       ; FREE_PVAL   # EYES
   1F441       ; UNASSIGNED  # <reserved>
   1F442..1F4F7; FREE_PVAL   # EAR..CAMERA
   1F4F8       ; UNASSIGNED  # <reserved>
   1F4F9..1F4FC; FREE_PVAL   # VIDEO CAMERA..VIDEOCASSETTE
   1F4FD..1F4FF; UNASSIGNED  # <reserved>..<reserved>
   1F500..1F53D; FREE_PVAL   # TWISTED RIGHTWARDS ARROWS..DOWN-POINTI
```

```
1F53E..1F53F; UNASSIGNED  # <reserved>..<reserved>
1F540..1F543; FREE_PVAL   # CIRCLED CROSS POMMEE..NOTCHED LEFT SEM
1F544..1F54F; UNASSIGNED  # <reserved>..<reserved>
1F550..1F567; FREE_PVAL   # CLOCK FACE ONE OCLOCK..CLOCK FACE TWEL
1F568..1F5FA; UNASSIGNED  # <reserved>..<reserved>
1F5FB..1F640; FREE_PVAL   # MOUNT FUJI..WEARY CAT FACE
1F641..1F644; UNASSIGNED  # <reserved>..<reserved>
1F645..1F650; FREE_PVAL   # FACE W NO GOOD GESTURE..PERSON W FO
1F650..1F67F; UNASSIGNED  # <reserved>..<reserved>
1F680..1F6C5; FREE_PVAL   # ROCKET..LEFT LUGGAGE
1F6C6..1F6FF; UNASSIGNED  # <reserved>..<reserved>
1F700..1F773; FREE_PVAL   # ALCHEMICAL SYMBOL FOR QUINTESSENCE..AL
1F774..1FFFF; UNASSIGNED  # <reserved>..<reserved>
20000..2A6D6; PVALID      # <CJK Ideograph Extension B>
2A6D7..2A6FF; UNASSIGNED  # <reserved>..<reserved>
2A700..2B734; PVALID      # <CJK Ideograph Extension C>
2A735..2A739; UNASSIGNED  # <reserved>..<reserved>
2A740..2B81D; PVALID      # <CJK Ideograph Extension D>
2B81E..2F7FF; UNASSIGNED  # <reserved>..<reserved>
2F800..2FA1D; PVALID      # CJK COMP IDEO-2F800..CJK COMPA
2FA1E..2FFFD; UNASSIGNED  # <reserved>..<reserved>
2FFFE..2FFFF; DISALLOWED  # <noncharacter>..<noncharacter>
30000..3FFFD; UNASSIGNED  # <reserved>..<reserved>
3FFFE..3FFFF; DISALLOWED  # <noncharacter>..<noncharacter>
40000..4FFFD; UNASSIGNED  # <reserved>..<reserved>
4FFFE..4FFFF; DISALLOWED  # <noncharacter>..<noncharacter>
50000..5FFFD; UNASSIGNED  # <reserved>..<reserved>
5FFFE..5FFFF; DISALLOWED  # <noncharacter>..<noncharacter>
60000..6FFFD; UNASSIGNED  # <reserved>..<reserved>
6FFFE..6FFFF; DISALLOWED  # <noncharacter>..<noncharacter>
70000..7FFFD; UNASSIGNED  # <reserved>..<reserved>
7FFFE..7FFFF; DISALLOWED  # <noncharacter>..<noncharacter>
80000..8FFFD; UNASSIGNED  # <reserved>..<reserved>
8FFFE..8FFFF; DISALLOWED  # <noncharacter>..<noncharacter>
90000..9FFFD; UNASSIGNED  # <reserved>..<reserved>
9FFFE..9FFFF; DISALLOWED  # <noncharacter>..<noncharacter>
A0000..AFFFD; UNASSIGNED  # <reserved>..<reserved>
AFFFE..AFFFF; DISALLOWED  # <noncharacter>..<noncharacter>
B0000..BFFFD; UNASSIGNED  # <reserved>..<reserved>
BFFFE..BFFFF; DISALLOWED  # <noncharacter>..<noncharacter>
C0000..CFFFD; UNASSIGNED  # <reserved>..<reserved>
CFFFE..CFFFF; DISALLOWED  # <noncharacter>..<noncharacter>
D0000..DFFFD; UNASSIGNED  # <reserved>..<reserved>
DFFFE..DFFFF; DISALLOWED  # <noncharacter>..<noncharacter>
E0000       ; UNASSIGNED  # <reserved>
E0001       ; DISALLOWED  # LANGUAGE TAG
E0002..E001F; UNASSIGNED  # <reserved>..<reserved>
E0020..E007F; DISALLOWED  # TAG SPACE..CANCEL TAG
```

```
E0080..E00FF; UNASSIGNED  # <reserved>..<reserved>
E0100..E01EF; DISALLOWED  # VAR SEL-17..VAR SEL-256
E01F0..EFFFD; UNASSIGNED  # <reserved>..<reserved>
EFFFE..10FFFF; DISALLOWED # <noncharacter>..<noncharacter>
```

Appendix B. Acknowledgements

The authors would like to acknowledge the comments and contributions of the following individuals: David Black, Mark Davis, Alan DeKok, Martin Duerst, Patrik Faltstrom, Ted Hardie, Joe Hildebrand, Bjoern Hoehrmann, Paul Hoffman, Jeffrey Hutzelman, Simon Josefsson, John Klensin, Alexey Melnikov, Takahiro Nemoto, Yoav Nir, Mike Parker, Pete Resnick, Andrew Sullivan, Dave Thaler, Yoshiro Yoneya, and Florian Zeitz.

Some algorithms and textual descriptions have been borrowed from [RFC5892]. Some text regarding security has been borrowed from [RFC5890] and [I-D.ietf-xmpp-6122bis].

Authors' Addresses

Peter Saint-Andre
Cisco Systems, Inc.
1899 Wynkoop Street, Suite 600
Denver, CO 80202
USA

Phone: +1-303-308-3282
Email: psaintan@cisco.com

Marc Blanchet
Viagenie
246 Aberdeen
Quebec, QC G1R 2E1
Canada

Email: Marc.Blanchet@viagenie.ca
URI: http://www.viagenie.ca/