NAME

Unicode::UTF8 – Encoding and decoding of UTF–8 encoding form

SYNOPSIS

```
use Unicode::UTF8 qw[decode_utf8 encode_utf8];

use warnings FATAL => 'utf8'; # fatalize encoding glitches
$string = decode_utf8($octets);
$octets = encode_utf8($string);
```

DESCRIPTION

This module provides functions to encode and decode UTF–8 encoding form as specified by Unicode and ISO/IEC 10646:2011.

FUNCTIONS

decode_utf8

```
$string = decode_utf8($octets);
$string = decode_utf8($octets, $fallback);
```

Returns an decoded representation of `$octets` in UTF–8 encoding as a character string.

`$fallback` is an optional `CODE` reference which provides a error-handling mechanism, allowing customization of error handling. The default error-handling mechanism is to replace any ill-formed UTF–8 sequences or encoded code points which can't be interchanged with REPLACEMENT CHARACTER (U+FFFD).

```
$string = $fallback->($octets, $is_usv, $position);
```

`$fallback` is invoked with three arguments: `$octets`, `$is_usv` and `$position`. `$octets` is a sequence of one or more octets containing the maximal subpart of the ill-formed subsequence or encoded code point which can't be interchanged. `$is_usv` is a boolean indicating whether or not `$octets` represent a encoded Unicode scalar value. `$position` is a unsigned integer containing the zero based octet position at which the error occurred within the octets provided to `decode_utf8()`. `$fallback` must return a character string consisting of zero or more Unicode scalar values. Unicode scalar values consist of code points in the range U+0000..U+D7FF and U+E000..U+10FFFF.

encode_utf8

```
$octets = encode_utf8($string);
$octets = encode_utf8($string, $fallback);
```

Returns an encoded representation of `$string` in UTF–8 encoding as an octet string.

`$fallback` is an optional `CODE` reference which provides a error-handling mechanism, allowing customization of error handling. The default error-handling mechanism is to replace any code points which can't be interchanged or represented in UTF–8 encoding form with REPLACEMENT CHARACTER (U+FFFD).

```
$string = $fallback->($codepoint, $is_usv, $position);
```

`$fallback` is invoked with three arguments: `$codepoint`, `$is_usv` and `$position`. `$codepoint` is a unsigned integer containing the code point which can't be interchanged or represented in UTF–8 encoding form. `$is_usv` is a boolean indicating whether or not `$codepoint` is a Unicode scalar value. `$position` is a unsigned integer containing the zero based character position at which the error occurred within the string provided to `encode_utf8()`. `$fallback` must return a character string consisting of zero or more Unicode scalar values.Unicode scalar values consist of code points in the range U+0000..U+D7FF and U+E000..U+10FFFF.

valid_utf8

```
$boolean = valid_utf8($octets);
```

Returns a boolean indicating whether or not the given `$octets` consist of well-formed UTF–8 sequences.

EXPORTS

None by default. All functions can be exported using the `:all` tag or individually.

DIAGNOSTICS

Can't decode a wide character string
(F) Wide character in octets.

Can't validate a wide character string
(F) Wide character in octets.

Can't decode ill-formed UTF-8 octet sequence <%s> in position `%u`
(W utf8) Encountered an ill-formed UTF-8 octet sequence. <%s> contains a hexadecimal representation of the maximal subpart of the ill-formed subsequence.

Can't interchange noncharacter code point U+%X in position `%u`
(W utf8, nonchar) Noncharacters are code points that are permanently reserved in the Unicode Standard for internal use. They are forbidden for use in open interchange of Unicode text data. Noncharacters consist of the values U+nFFFE and U+nFFFF (where n is from 0 to 10^16) and the values U+FDD0..U+FDEF.

Can't represent surrogate code point U+%X in position `%u`
(W utf8, surrogate) Surrogate code points are designated only for surrogate code units in the UTF-16 character encoding form. Surrogates consist of code points in the range U+D800 to U+DFFF.

Can't represent super code point \x{%X} in position `%u`
(W utf8, non_unicode) Code points greater than U+10FFFF. Perl's extended codespace.

Can't decode ill-formed UTF-X octet sequence <%s> in position `%u`
(F) Encountered an ill-formed octet sequence in Perl's internal representation of wide characters.

The sub-categories: `nonchar`, `surrogate` and `non_unicode` is only available on Perl 5.14 or greater. See perllexwarn for available categories and hierarchies.

COMPARISON

Here is a summary of features for comparison with Encode's UTF-8 implementation:

- Simple API which makes use of Perl's standard warning categories.
- Recognizes all noncharacters regardless of Perl version
- Implements Unicode's recommended practice for using U+FFFD.
- Better diagnostics in warning messages
- Detects and reports inconsistency in Perl's internal representation of wide characters (UTF-X)
- Preserves taintedness of decoded `$octets` or encoded `$string`
- Better performance ~ 600% – 1200% (JA: 600%, AR: 700%, SV: 900%, EN: 1200%, see benchmarks directory in git repository)

CONFORMANCE

It's the author's belief that this UTF-8 implementation is conformant with the Unicode Standard Version 6.0. Any deviations from the Unicode Standard is to be considered a bug.

SEE ALSO

Encode
<http://www.unicode.org/>

SUPPORT

BUGS

Please report any bugs by email to `bug-unicode-utf8 at rt.cpan.org`, or through the web interface at <http://rt.cpan.org/Public/Dist/Display.html?Name=Unicode-UTF8>. You will be automatically notified of any progress on the request by the system.

SOURCE CODE

This is open source software. The code repository is available for public review and contribution under the terms of the license.

<http://github.com/chansen/p5-unicode-utf8>

```
git clone http://github.com/chansen/p5-unicode-utf8
```

AUTHOR

Christian Hansen chansen@cpan.org

COPYRIGHT