

**NAME**

punycode\_decode - API function

**SYNOPSIS**

```
#include <punycode.h>
```

```
int punycode_decode(size_t input_length, const char [] input, size_t * output_length, punycode_uint [] output, unsigned char [] case_flags);
```

**ARGUMENTS**

size\_t input\_length

The number of ASCII code points in the *input* array.

const char [] input

An array of ASCII code points (0..7F).

size\_t \* output\_length

The caller passes in the maximum number of code points that it can receive into the *output* array (which is also the maximum number of flags that it can receive into the *case\_flags* array, if *case\_flags* is not a **NULL** pointer). On successful return it will contain the number of code points actually output (which is also the number of flags actually output, if *case\_flags* is not a null pointer). The decoder will never need to output more code points than the number of ASCII code points in the input, because of the way the encoding is defined. The number of code points output cannot exceed the maximum possible value of a punycode\_uint, even if the supplied *output\_length* is greater than that.

punycode\_uint [] output

An array of code points like the input argument of **punycode\_encode()** (see above).

unsigned char [] case\_flags

A **NULL** pointer (if the flags are not needed by the caller) or an array of boolean values parallel to the *output* array. Nonzero (true, flagged) suggests that the corresponding Unicode character be forced to uppercase by the caller (if possible), and zero (false, unflagged) suggests that it be forced to lowercase (if possible). ASCII code points (0..7F) are output already in the proper case, but their flags will be set appropriately so that applying the flags would be harmless.

**DESCRIPTION**

Converts Punycode to a sequence of code points (presumed to be Unicode code points).

Return value: The return value can be any of the **Punycode\_status** values defined above. If not **PUNYCODE\_SUCCESS**, then *output\_length* , *output* , and *case\_flags* might contain garbage.

## REPORTING BUGS

Report bugs to <help-libidn@gnu.org>.

General guidelines for reporting bugs: <https://www.gnu.org/gethelp/>

GNU Libidn home page: <https://www.gnu.org/software/libidn/>

## COPYRIGHT

Copyright (C) 2002-2024 Simon Josefsson.

Copying and distribution of this file, with or without modification, are permitted in any medium without royalty provided the copyright notice and this notice are preserved.

## SEE ALSO

The full documentation for **libidn** is maintained as a Texinfo manual. If the **info** and **libidn** programs are properly installed at your site, the command

**info libidn**

should give you access to the complete manual. As an alternative you may obtain the manual from:

**<https://www.gnu.org/software/libidn/manual/>**