

NAME

stringprep_4i - API function

SYNOPSIS

```
#include <stringprep.h>
```

```
int stringprep_4i(uint32_t * ucs4, size_t * len, size_t maxucs4len, Stringprep_profile_flags flags, const Stringprep_profile * profile);
```

ARGUMENTS

uint32_t * ucs4

input/output array with string to prepare.

size_t * len on input, length of input array with Unicode code points, on exit, length of output array with Unicode code points.

size_t maxucs4len

maximum length of input/output array.

Stringprep_profile_flags flags

a **Stringprep_profile_flags** value, or 0.

const Stringprep_profile * profile

pointer to **Stringprep_profile** to use.

DESCRIPTION

Prepare the input UCS-4 string according to the stringprep profile, and write back the result to the input string.

The input is not required to be zero terminated (*ucs4* [*len*] = 0). The output will not be zero terminated unless *ucs4* [*len*] = 0. Instead, see **stringprep_4zi()** if your input is zero terminated or if you want the output to be.

Since the stringprep operation can expand the string, *maxucs4len* indicate how large the buffer holding the string is. This function will not read or write to code points outside that size.

The *flags* are one of **Stringprep_profile_flags** values, or 0.

The *profile* contain the **Stringprep_profile** instructions to perform. Your application can define new profiles, possibly re-using the generic stringprep tables that always will be part of the library, or use

one of the currently supported profiles.

Return value: Returns **STRINGPREP_OK** iff successful, or an **Stringprep_rc** error code.

DESCRIPTION

Prepare the input UCS-4 string according to the stringprep profile, and write back the result to the input string.

The input is not required to be zero terminated (*ucs4* [*len*] = 0). The output will not be zero terminated unless *ucs4* [*len*] = 0. Instead, see **stringprep_4zi()** if your input is zero terminated or if you want the output to be.

Since the stringprep operation can expand the string, *maxucs4len* indicate how large the buffer holding the string is. This function will not read or write to code points outside that size.

The *flags* are one of **Stringprep_profile_flags** values, or 0.

The *profile* contain the **Stringprep_profile** instructions to perform. Your application can define new profiles, possibly re-using the generic stringprep tables that always will be part of the library, or use one of the currently supported profiles.

Return value: Returns **STRINGPREP_OK** iff successful, or an **Stringprep_rc** error code.

REPORTING BUGS

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

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

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

COPYRIGHT

Copyright (C) 2002-2021 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:

<http://www.gnu.org/software/libidn/manual/>