

NAME

humanize_number - format a number into a human readable form

LIBRARY

System Utilities Library (libutil, -lutil)

SYNOPSIS

```
#include <libutil.h>
```

int

```
humanize_number(char *buf, size_t len, int64_t number, const char *suffix, int scale, int flags);
```

DESCRIPTION

The **humanize_number**() function formats the signed 64-bit quantity given in *number* into *buf*. A space and then *suffix* is appended to the end. The buffer pointed to by *buf* must be at least *len* bytes long.

If the formatted number (including *suffix*) would be too long to fit into *buf*, then divide *number* by 1024 until it will. In this case, prefix *suffix* with the appropriate designator. The **humanize_number**() function follows the traditional computer science conventions by default, rather than the IEEE/IEC (and now also SI) power of two convention or the power of ten notion. This behaviour however can be altered by specifying the HN_DIVISOR_1000 and HN_IEC_PREFIXES flags.

The traditional (default) prefixes are:

Prefix	Description	Multiplier	Multiplier 1000x
(note)	kilo	1024	1000
M	mega	1048576	1000000
G	giga	1073741824	1000000000
T	tera	1099511627776	1000000000000
P	peta	1125899906842624	1000000000000000
E	exa	1152921504606846976	1000000000000000000

Note: An uppercase K indicates a power of two, a lowercase k a power of ten.

The IEEE/IEC (and now also SI) power of two prefixes are:

Prefix	Description	Multiplier
Ki	kibi	1024
Mi	mebi	1048576
Gi	gibi	1073741824

Ti	tebi	1099511627776
Pi	pebi	1125899906842624
Ei	exbi	1152921504606846976

The *len* argument must be at least 4 plus the length of *suffix*, in order to ensure a useful result is generated into *buf*. To use a specific prefix, specify this as *scale* (multiplier = 1024^{scale} ; when HN_DIVISOR_1000 is specified, multiplier = 1000^{scale}). This cannot be combined with any of the *scale* flags below.

The following flags may be passed in *scale*:

HN_AUTOSCALE	Format the buffer using the lowest multiplier possible.
HN_GETSCALE	Return the prefix index number (the number of times <i>number</i> must be divided to fit) instead of formatting it to the buffer.

The following flags may be passed in *flags*:

HN_DECIMAL	If the final result is less than 10, display it using one decimal place.
HN_NOSPACE	Do not put a space between <i>number</i> and the prefix.
HN_B	Use 'B' (bytes) as prefix if the original result does not have a prefix.
HN_DIVISOR_1000	Divide <i>number</i> with 1000 instead of 1024.
HN_IEC_PREFIXES	Use the IEE/IEC notion of prefixes (Ki, Mi, Gi...). This flag has no effect when HN_DIVISOR_1000 is also specified.

RETURN VALUES

Upon success, the **humanize_number** function returns the number of characters that would have been stored in *buf* (excluding the terminating NUL) if *buf* was large enough, or -1 upon failure. Even upon failure, the contents of *buf* may be modified. If HN_GETSCALE is specified, the prefix index number will be returned instead.

SEE ALSO

expand_number(3)

STANDARDS

The HN_DIVISOR_1000 and HN_IEC_PREFIXES flags conform to ISO/IEC Std 80000-13:2008 and IEEE Std 1541-2002.

HISTORY

The **humanize_number()** function first appeared in NetBSD 2.0 and then in FreeBSD 5.3. The HN_IEC_PREFIXES flag was introduced in FreeBSD 9.0.

CAVEATS

For numbers greater than 999 using buffer length of 4 and less can cause rounding errors. When using HN_IEC_PREFIXES the same error occurs for buffer length of 5 or less.