

NAME

confstr - get string-valued configurable variables

LIBRARY

Standard C Library (libc, -lc)

SYNOPSIS

```
#include <unistd.h>
```

size_t

```
confstr(int name, char *buf, size_t len);
```

DESCRIPTION

This interface is specified by IEEE Std 1003.1-2001 ("POSIX.1"). A more flexible (but non-portable) interface is provided by `sysctl(3)`.

The **confstr()** function provides a method for applications to get configuration defined string values. Shell programmers needing access to these parameters should use the `getconf(1)` utility.

The *name* argument specifies the system variable to be queried. Symbolic constants for each name value are found in the include file `<unistd.h>`. The *len* argument specifies the size of the buffer referenced by the argument *buf*. If *len* is non-zero, *buf* is a non-null pointer, and *name* has a value, up to *len* - 1 bytes of the value are copied into the buffer *buf*. The copied value is always null terminated.

The available values are as follows:

`_CS_PATH`

Return a value for the PATH environment variable that finds all the standard utilities.

RETURN VALUES

If the call to **confstr()** is not successful, 0 is returned and *errno* is set appropriately. Otherwise, if the variable does not have a configuration defined value, 0 is returned and *errno* is not modified. Otherwise, the buffer size needed to hold the entire configuration-defined value is returned. If this size is greater than the argument *len*, the string in *buf* was truncated.

ERRORS

The **confstr()** function may fail and set *errno* for any of the errors specified for the library functions `malloc(3)` and `sysctl(3)`.

In addition, the following errors may be reported:

[EINVAL] The value of the *name* argument is invalid.

SEE ALSO

getconf(1), pathconf(2), sysconf(3), sysctl(3)

HISTORY

The **confstr()** function first appeared in 4.4BSD.