



Full credit is given to the above companies including the OS that this PDF file was generated!

Red Hat Enterprise Linux Release 9.2 Manual Pages on 'nice.3p' command

\$ man nice.3p

NICE(3P) POSIX Programmer's Manual NICE(3P)

PROLOG

This manual page is part of the POSIX Programmer's Manual. The Linux implementation of this interface may differ (consult the corresponding Linux manual page for details of Linux behavior), or the interface may not be implemented on Linux.

NAME

nice ? change the nice value of a process

SYNOPSIS

```
#include <unistd.h>

int nice(int incr);
```

DESCRIPTION

The nice() function shall add the value of incr to the nice value of the calling process. A nice value of a process is a non-negative number for which a more positive value shall result in less favorable scheduling.

A maximum nice value of $2^{\{NZERO\}}-1$ and a minimum nice value of 0 shall be imposed by the system. Requests for values above or below these limits shall result in the nice value being set to the corresponding limit. Only a process with appropriate privileges can lower the nice value.

Calling the nice() function has no effect on the priority of processes or threads with policy SCHED_FIFO or SCHED_RR. The effect on processes

or threads with other scheduling policies is implementation-defined.

The `nice` value set with `nice()` shall be applied to the process. If the process is multi-threaded, the `nice` value shall affect all system scope threads in the process.

As `-1` is a permissible return value in a successful situation, an application wishing to check for error situations should set `errno` to `0`, then call `nice()`, and if it returns `-1`, check to see whether `errno` is non-zero.

RETURN VALUE

Upon successful completion, `nice()` shall return the new `nice` value `-(NZERO)`. Otherwise, `-1` shall be returned, the `nice` value of the process shall not be changed, and `errno` shall be set to indicate the error.

ERRORS

The `nice()` function shall fail if:

`EPERM` The `incr` argument is negative and the calling process does not have appropriate privileges.

The following sections are informative.

EXAMPLES

Changing the Nice Value

The following example adds the value of the `incr` argument, `-20`, to the `nice` value of the calling process.

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

```
...
```

```
int incr = -20;
```

```
int ret;
```

```
ret = nice(incr);
```

APPLICATION USAGE

None.

RATIONALE

None.

FUTURE DIRECTIONS

None.

SEE ALSO

`exec`, `getpriority()`

The Base Definitions volume of POSIX.1-2017, `<limits.h>`, `<unistd.h>`

COPYRIGHT

Portions of this text are reprinted and reproduced in electronic form from IEEE Std 1003.1-2017, Standard for Information Technology -- Portable Operating System Interface (POSIX), The Open Group Base Specifications Issue 7, 2018 Edition, Copyright (C) 2018 by the Institute of Electrical and Electronics Engineers, Inc and The Open Group. In the event of any discrepancy between this version and the original IEEE and The Open Group Standard, the original IEEE and The Open Group Standard is the referee document. The original Standard can be obtained online at <http://www.opengroup.org/unix/online.html>.

Any typographical or formatting errors that appear in this page are most likely to have been introduced during the conversion of the source files to man page format. To report such errors, see https://www.kernel.org/doc/man-pages/reporting_bugs.html.

IEEE/The Open Group

2017

NICE(3P)