NAME

nice — adjust scheduling priority

SYNOPSIS

nice
nice [**-n** *diff*] *program* [*argument*]...

DESCRIPTION

With *program*, adds *diff* (default **10**) to the niceness, then executes *program arguments*; otherwise writes current niceness to the standard output stream.

Niceness, ranging [**-20**, **19**], reflects the (inverse) scheduling priority. *diff* may be any integer, but will be clamped to that range by the system; lowering niceness (increasing priority) is a privileged operation: if it fails, a diagnostic is issued, but *program* is still executed.

OPTIONS

-n, **--adjustment**=*diff* Alter niceness by this much. Default: **10**.

ENVIRONMENT

PATH In which *program* is searched, confer execvp(3).

EXIT STATUS

127	*program* wasn't found.
126	*program* exists, but couldn't be executed for a different reason.
125	Internal error.
All others	returned by *program*.

SEE ALSO

nice(3)

STANDARDS

Conforms to IEEE Std 1003.1-2024 ("POSIX.1").

The no-*program* behaviour is an extension, also present on the GNU system.

The default **-n** value is **10** on all known systems; the upper end of the range is **20**, not **19**, on OpenBSD.

HISTORY

A way to alter scheduling priority first appeared in Version 2 AT&T UNIX as hog(II), "set low-priority status" in the index and:

```
NAME        hog -- set program in low priority
```

Moving the caller to the low-priority queue, which "background jobs that execute for a long time should do". The job was moved back to the regular queue "as soon as the process is dismissed for any reason other than quantum overflow" (when a syscall is made).

Version 3 AT&T UNIX renamed it to nice(II), noting "Once done, there is no way to restore a process to normal priority.".

Version 4 AT&T UNIX sees nice(II) amended with a priority argument and a noted range of "20 to -220" (incorrect, of course — it's a maximum of **19** and a minimum of **0** for regular users and CHAR_MIN for root). **16** is recommended for "users who wish to execute long-running programs without flak from the administration.". Indeed, this is what the new nice(I) command does.

Version 6 AT&T UNIX nice(I) defaults to **4** and accepts a **-***niceness* argument to override it.

Version 7 AT&T UNIX defaults to **10** (and uses PATH to find *program*). nice(2) now applies an increment, exactly like today, and notes a [**-20**, **20**] range (wrong, of course; the actual range is [**-20**, **19**], like today).

X/Open Portability Guide Issue 4 (“XPG4”) standardises it, except it invents **-n** *diff* to supersede **-***diff*, noting the latter as obsolete, as it violates the Utility Syntax Guidelines; IEEE Std 1003.1-2001 (“POSIX.1”) removes it. **-n** was likely chosen because the Version 6 AT&T UNIX and later usage strings are

```
usage: nice [ -n ] command
```