NAME

getpriority, setpriority - get/set program scheduling priority

SYNOPSIS

#include <sys/resource.h>

int getpriority (int which, id_t who);
int setpriority (int which, id_t who, int prio);

DESCRIPTION

The scheduling priority of the process, process group, or user, as
indicated by which and who is obtained with the getpriority call and set
with the setpriority call. Which is one of PRIO_PROCESS, PRIO_PGRP, or
PRIO_USER, and who is interpreted relative to which (a process identifier
for PRIO_PROCESS, process group identifier for PRIO_PGRP, and a user ID
for PRIO_USER). A zero value of who denotes the current process, process
group, or user. Prio is a value in the range -20 to 20. The default
priority is 0; lower priorities cause more favorable scheduling.

The getpriority call returns the highest priority (lowest numerical
value) enjoyed by any of the specified processes. The setpriority call
sets the priorities of all of the specified processes to the specified
value. Only the super-user may lower priorities.

getpriority and setpriority may return one of the following errors:

[ESRCH]   No process was located using the which and who values
               specified.

[EINVAL]   Which was not one of PRIO_PROCESS, PRIO_PGRP, or
               PRIO_USER.

In addition to the errors indicated above, setpriority may fail with one
of the following errors returned:

[EPERM]   A process was located, but neither its effective nor real
               user ID matched the effective user ID of the caller.

[EACCES]   A non super-user attempted to lower a process priority.

SEE ALSO

renice(1M), fork(2), nice(1), schedctl(2).

DIAGNOSTICS

Since getpriority can legitimately return the value -1, it is necessary
to clear the external variable errno prior to the call, then check it
afterward to determine if a -1 is an error or a legitimate value. The
setpriority call returns 0 if there is no error, or -1 if there is.