

**NAME**

`sched_get_priority_max`, `sched_get_priority_min` - get static priority range

**SYNOPSIS**

```
#include <sched.h>
```

```
int sched_get_priority_max(int policy);
```

```
int sched_get_priority_min(int policy);
```

**DESCRIPTION**

`sched_get_priority_max()` returns the maximum priority value that can be used with the scheduling algorithm identified by *policy*. `sched_get_priority_min()` returns the minimum priority value that can be used with the scheduling algorithm identified by *policy*. Supported *policy* values are **SCHED\_FIFO**, **SCHED\_RR**, **SCHED\_OTHER**, **SCHED\_BATCH**, **SCHED\_IDLE**, and **SCHED\_DEADLINE**. Further details about these policies can be found in [sched\(7\)](#).

Processes with numerically higher priority values are scheduled before processes with numerically lower priority values. Thus, the value returned by `sched_get_priority_max()` will be greater than the value returned by `sched_get_priority_min()`.

Linux allows the static priority range 1 to 99 for the **SCHED\_FIFO** and **SCHED\_RR** policies, and the priority 0 for the remaining policies. Scheduling priority ranges for the various policies are not alterable.

The range of scheduling priorities may vary on other POSIX systems, thus it is a good idea for portable applications to use a virtual priority range and map it to the interval given by `sched_get_priority_max()` and `sched_get_priority_min`. POSIX.1 requires a spread of at least 32 between the maximum and the minimum values for **SCHED\_FIFO** and **SCHED\_RR**.

POSIX systems on which `sched_get_priority_max()` and `sched_get_priority_min()` are available define **\_POSIX\_PRIORITY\_SCHEDULING** in `<unistd.h>`.

**RETURN VALUE**

On success, `sched_get_priority_max()` and `sched_get_priority_min()` return the maximum/minimum priority value for the named scheduling policy. On error, -1 is returned, and *errno* is set appropriately.

**ERRORS****EINVAL**

The argument *policy* does not identify a defined scheduling policy.

**CONFORMING TO**

POSIX.1-2001, POSIX.1-2008.

**SEE ALSO**

[sched\\_getaffinity\(2\)](#), [sched\\_getparam\(2\)](#), [sched\\_getscheduler\(2\)](#), [sched\\_setaffinity\(2\)](#), [sched\\_setparam\(2\)](#), [sched\\_setscheduler\(2\)](#), [sched\(7\)](#)

**COLOPHON**

This page is part of release 4.10 of the Linux *man-pages* project. A description of the project, information about reporting bugs, and the latest version of this page, can be found at <https://www.kernel.org/doc/man-pages/>.