

**NAME**

**sched\_get\_priority\_max**, **sched\_get\_priority\_min**, **sched\_rr\_get\_interval** - get scheduling parameter limits

**LIBRARY**

Standard C Library (libc, -lc)

**SYNOPSIS**

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

```
int
```

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

```
int
```

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

```
int
```

```
sched_rr_get_interval(pid_t pid, struct timespec *interval);
```

**DESCRIPTION**

The **sched\_get\_priority\_max()** and **sched\_get\_priority\_min()** system calls return the appropriate maximum or minimum, respectively, for the scheduling policy specified by *policy*. The **sched\_rr\_get\_interval()** system call updates the *timespec* structure referenced by the *interval* argument to contain the current execution time limit (i.e., time quantum) for the process specified by *pid*. If *pid* is zero, the current execution time limit for the calling process is returned.

The value of *policy* should be one of the scheduling policy values defined in *<sched.h>*:

[SCHED\_FIFO] First-in-first-out fixed priority scheduling with no round robin scheduling;

[SCHED\_OTHER] The standard time sharing scheduler;

[SCHED\_RR] Round-robin scheduling across same priority processes.

**RETURN VALUES**

If successful, the **sched\_get\_priority\_max()** and **sched\_get\_priority\_min()** system calls shall return the appropriate maximum or minimum values, respectively. If unsuccessful, they shall return a value of -1 and set *errno* to indicate the error.

The **sched\_rr\_get\_interval()** function returns the value 0 if successful; otherwise the value -1 is returned

and the global variable *errno* is set to indicate the error.

## ERRORS

On failure *errno* will be set to the corresponding value:

[EINVAL]           The value of the *policy* argument does not represent a defined scheduling policy.

[ENOSYS]           The **sched\_get\_priority\_max()**, **sched\_get\_priority\_min()**, and **sched\_rr\_get\_interval()** system calls are not supported by the implementation.

[ESRCH]            No process can be found corresponding to that specified by *pid*.

## SEE ALSO

`sched_getparam(2)`, `sched_getscheduler(2)`, `sched_setparam(2)`, `sched_setscheduler(2)`

## STANDARDS

The **sched\_get\_priority\_max()**, **sched\_get\_priority\_min()**, and **sched\_rr\_get\_interval()** system calls conform to IEEE Std 1003.1b-1993 ("POSIX.1b").