

PThreads

Pthread Functions

Thread Management

```
int pthread_create(pthread_t *restrict thread, const pthread_attr_t *restrict attr,
                 void *(*start_routine)(void*), void *restrict arg);
void pthread_exit(void *value_ptr);
int pthread_join(pthread_t thread, void **value_ptr);
int pthread_once(pthread_once_t *once_control, void (*init_routine)(void));
int pthread_kill(pthread_t thread, int sig);
pthread_t pthread_self(void);
int pthread_equal(pthread_t t1, pthread_t t2);
void pthread_yield ();
int pthread_detach(pthread_t thread);
```

Thread-Specific Data

```
int pthread_key_create(pthread_key_t *key, void (*destructor)(void*));
int pthread_key_delete(pthread_key_t key);
void *pthread_getspecific(pthread_key_t key);
int pthread_setspecific(pthread_key_t key, const void *value);
```

Thread Cancellation

```
int pthread_cancel(pthread_t thread);
void pthread_cleanup_pop(int execute);
void pthread_cleanup_push(void (*routine)(void*), void *arg);
int pthread_setcancelstate(int state, int *oldstate);
int pthread_setcanceltype(int type, int *oldtype);
void pthread_testcancel(void);
```

Thread Scheduling

```
int pthread_getschedparam(pthread_t thread, int *restrict policy, struct sched_param *restrict param);
int pthread_setschedparam(pthread_t thread, int policy, const struct sched_param *param);
```

Signals

```
int pthread_sigmask(int how, const sigset_t *restrict set, sigset_t *restrict oset);
int sigprocmask(int how, const sigset_t *restrict set, sigset_t *restrict oset);
```

Pthread Attribute Functions

Basic Management

```
int pthread_attr_destroy(pthread_attr_t *attr);
int pthread_attr_init(pthread_attr_t *attr);
```

Detachable or Joinable

```
int pthread_attr_getdetachstate(const pthread_attr_t *attr, int *detachstate);
int pthread_attr_setdetachstate(pthread_attr_t *attr, int detachstate);
```

Specifying Stack Information

```
int pthread_attr_getstackaddr(const pthread_attr_t *restrict attr, void **restrict stackaddr);
int pthread_attr_setstackaddr(pthread_attr_t *attr, void *stackaddr);
int pthread_attr_getstacksize(const pthread_attr_t *restrict attr, size_t *restrict stacksize);
int pthread_attr_setstacksize(pthread_attr_t *attr, size_t stacksize);
```

Thread Scheduling Attributes

```
int pthread_attr_getschedparam(const pthread_attr_t *restrict attr, struct sched_param *restrict param);
int pthread_attr_setschedparam(pthread_attr_t *restrict attr, const struct sched_param *restrict param);
int pthread_attr_getschedpolicy(const pthread_attr_t *restrict attr, int *restrict policy);
int pthread_attr_setschedpolicy(pthread_attr_t *attr, int policy);
int pthread_attr_getinheritsched(const pthread_attr_t *restrict attr, int *restrict inheritsched);
int pthread_attr_setinheritsched(pthread_attr_t *attr, int inheritsched);
int pthread_attr_getscope(const pthread_attr_t *restrict attr, int *restrict contentionscope);
int pthread_attr_setscope(pthread_attr_t *attr, int contentionscope);
```

Constants

```
PTHREAD_MUTEX_INITIALIZER
PTHREAD_RECURSIVE_MUTEX_INITIALIZER
PTHREAD_ERRORCHECK_MUTEX_INITIALIZER
PTHREAD_COND_INITIALIZER
PTHREAD_RWLOCK_INITIALIZER
PTHREAD_SPINLOCK_INITIALIZER
PTHREAD_CREATE_JOINABLE
PTHREAD_CREATE_DETACHED
```

Mutex Functions

Mutex Management

```
int pthread_mutex_destroy(pthread_mutex_t *mutex);
int pthread_mutex_init(pthread_mutex_t *restrict mutex, const pthread_mutexattr_t *restrict attr);
int pthread_mutex_lock(pthread_mutex_t *mutex);
int pthread_mutex_trylock(pthread_mutex_t *mutex);
int pthread_mutex_unlock(pthread_mutex_t *mutex);
```

Priority Management

```
int pthread_mutex_getprioceiling(const pthread_mutex_t *restrict mutex, int *restrict prioceiling);
int pthread_mutex_setprioceiling(pthread_mutex_t *restrict mutex, int prioceiling, int *restrict old_ceiling);
```

Mutex Attribute Functions

Basic Management

```
int pthread_mutexattr_destroy(pthread_mutexattr_t *attr);
int pthread_mutexattr_init(pthread_mutexattr_t *attr);
```

Sharing

```
int pthread_mutexattr_getpshared(const pthread_mutexattr_t * restrict attr, int *restrict pshared);
int pthread_mutexattr_setpshared(pthread_mutexattr_t *attr, int pshared);
```

Protocol Attributes

```
int pthread_mutexattr_getprotocol(const pthread_mutexattr_t * restrict attr, int *restrict protocol);
int pthread_mutexattr_setprotocol(pthread_mutexattr_t *attr, int protocol);
```

Priority Management

```
int pthread_mutexattr_getprioceiling(const pthread_mutexattr_t * restrict attr, int *restrict prioceiling);
int pthread_mutexattr_setprioceiling(pthread_mutexattr_t *attr, int prioceiling);
```

Condition Variable Functions

Basic Management

```
int pthread_cond_destroy(pthread_cond_t *cond);
int pthread_cond_init(pthread_cond_t *restrict cond, const pthread_condattr_t *restrict attr);
int pthread_cond_broadcast(pthread_cond_t *cond);
int pthread_cond_signal(pthread_cond_t *cond);
int pthread_cond_timedwait(pthread_cond_t *restrict cond, pthread_mutex_t *restrict mutex, const struct timespec *restrict abstime);
int pthread_cond_wait(pthread_cond_t *restrict cond, pthread_mutex_t *restrict mutex);
```

Condition Variable Attribute Functions

Basic Management

```
int pthread_condattr_destroy(pthread_condattr_t *attr);
int pthread_condattr_init(pthread_condattr_t *attr);
```

Sharing

```
int pthread_condattr_getpshared(const pthread_condattr_t *restrict attr, int *restrict pshared);
int pthread_condattr_setpshared(pthread_condattr_t *attr, int pshared);
```