

Condition Variables

A **Condition variable** is not really a variable in the normal sense. It is Really more of a data structure in which a thread waits until it receives a .

Once a condition is created, it can be destroyed, but no assignment can be made to it.

Condition variables usually must be used with mutexes to get correct results.

`pthread_cond_init` - Creates a new condition variable.

`pthread_cond_destroy` - Destroys an existing condition variable.

`pthread_cond_wait` - Thread blocks until signalled.

`pthread_cond_signal` - unblocks a blocked thread.

`pthread_cond_broadcast` - unblocks all blocked threads.

```
#include <pthread.h>
int pthread_cond_init (pthread_cond_t *cond,
                      Const pthread_cond_attr *attr);
```

This routine creates a new condition variable named `cond`, specified by `attr`.

If `attr` is `NULL`, then the default attributes will be used.

```
#include <pthread.h>
```

```
int pthread_cond_destroy (pthread_cond_t *cond);
```

This routine destroys the condition variable `cond`.

```
#include <pthread.h>
```

```
int pthread_cond_wait (pthread_cond_t *cond, pthread_mutex_t *mtx)
```

This routine automatically blocks the current thread, so that it is waiting on the condition variable `cond`, and unlocks the mutex variable `mtx`.

The calling thread unblocks only after another thread calls `pthread_cond_signal`.

```
#include <pthread.h>
```

```
int pthread_cond_signal (pthread_cond_t *cond);
```

This routine unblocks ONE thread blocked on the condition variable `cond`.

If no thread is blocked on `cond`, then it does nothing.