

CS551

Warm-up Project #2

Bill Cheng

<http://merlot.usc.edu/cs551-f12>



Multi-threading Exercise

- ➔ Make sure you are familiar with the *pthread* library
 - = good source is the book by Nichols, Buttlar, and Farrell “*Pthreads Programming*”, O’Rielly & Associates, 1996
 - = you must learn how to use mutex and condition variables correctly
 - `pthread_mutex_lock()` / `pthread_mutex_unlock()`
 - `pthread_cond_wait()` / `pthread_cond_signal()` / `pthread_cond_broadcast()`
 - = you must learn how to handle UNIX signals
 - `pthread_sigmask()` / `sigwait()`
 - `pthread_setcancelstate()`
 - `pthread_setcanceltype()`
 - `pthread_testcancel()`



pthread_sigmask()

➡ Look at the man pages of `pthread_sigmask()` on nunki and try to understand the example there

- = designate child thread to handler SIGINT
- = parent thread blocks SIGINT

```
#include <pthread.h>
/* #include <thread.h> */

thread_t user_threadID;
sigset_t new;

void *handler(), interrupt();

main( int argc, char *argv[] ) {
    sigemptyset(&new);
    sigaddset(&new, SIGINT);

    pthread_sigmask(SIG_BLOCK, &new, NULL);
    pthread_create(&user_threadID, NULL, handler, argv[1]);
    pthread_join(user_threadID, NULL);

    printf("thread handler, %d exited\n", user_threadID);
    sleep(2);
    printf("main thread, %d is done\n", thr_self());
} /* end main */
```

