```
#
# Makefile for nthreads
# Author: Michael Kepple
# Date:   02 apr 2013
#
all: nthreads.c nthreads.h
        gcc nthreads.c nthreads.h -lpthread -o nthreads

clean:
        rm -rf *.o nthreads
/*
 *
 * nthreads.c
 *
 * Author: Michael Kepple
 * Date:   02 Apr 2013
 * Description: Creates multiple threads based on command line arguments. These
 *   threads sleep for 60 seconds then exit.
 *
*/
#include "nthreads.h"
pthread_cond_t cond;
pthread_mutex_t lock;

int main(int argc, char* argv[])
{
    char ch;
    int runTime = 0;
    printf("Welcome to NTHREADS!\n");
    if (argc != 2 || !atoi(argv[1]))
        exit(EXIT_FAILURE);
    int i, numThreads = atoi(argv[1]);
    pthread_t threads[numThreads];
    for (i = 0; i < numThreads; i++)
        pthread_create(&threads[i], NULL, (void*) runThread, NULL);
    printf("Total Threads Created: %d\n", numThreads);
    printf("Output of 'ps -e -T | grep nthreads:'\n");
    system("ps -e -T | grep nthreads");
    for (i = 0; i < numThreads; i++)
        pthread_join(threads[i], NULL);
    printf("Output of 'ps -e -T | grep nthreads' after join'ing:\n");
    system("ps -e -T | grep nthreads");

}

/*
 * Function: runThread
 * Description: causes each created thread to sleep for sixty seconds then exit.
 * Params: None.
 * Returns: Nothing.
 * Modifies: Nothing.
*/
void runThread()
{
    sleep(60);
}
/*
 * File:   nthreads.h
 * Author: Michael Kepple
 * Date:   02 Apr 2013
 *
*/
#ifndef NTHREADS_H
```

```
#define NTHREADS_H
#include <stdlib.h>
#include <stdio.h>
#include <pthread.h>
#include <time.h>
void runThread();
#endif
/*
 * Results of running Nthreads (includes built-in test functionality).
 *
*/
Welcome to NTHREADS!
Total Threads Created: 4
Output of 'ps -e -T | grep nthreads:'
30564 30564 pts/23   00:00:00 nthreads
30564 30565 pts/23   00:00:00 nthreads
30564 30566 pts/23   00:00:00 nthreads
30564 30567 pts/23   00:00:00 nthreads
30564 30568 pts/23   00:00:00 nthreads
Output of 'ps -e -T | grep nthreads' after join'ing:
30564 30564 pts/23   00:00:00 nthreads
```