

#include <pthread.h>
#include <stdio.h>

int sum; /* this data is shared by the thread(s) */
void *runner(void *param); /* the thread */

int main(int argc, char *argv[]) 
{
    pthread_t tid; /* the thread identifier */
    pthread_attr_t attr; /* set of thread attributes */

    if (argc != 2) {
        fprintf(stderr,"usage: a.out <integer value>
        return -1;
    }

    if (atoi(argv[1]) < 0) {
        fprintf(stderr,"%d must be >= 0",atoi(argv[1]));
        return -1;
    }

    pthread_attr_init(&attr);
    pthread_create(&tid,&attr,runner,argv[1]);
    pthread_join(tid,NULL);

    printf("sum = %d\n",sum);
}

/* The thread will begin control in this function */
void *runner(void *param)
{
    int i, upper = atoi(param);
    sum = 0;

    for (i = 1; i <= upper; i++)
        sum += i;

    pthread_exit(0);
}

Figure 4.9 Multithreaded C program using the Pthreads API.

operation in the runner() function. After creating the summation thread, the parent thread will wait for it to complete by calling the pthread_join() function. The summation thread will complete when it calls the function pthread_exit(). Once the summation thread has returned, the parent thread will output the value of the shared data sum.