```
/* Jacobi iteration using Pthreads */

#include <pthread.h>
#include <stdio.h>
#define SHARED 1
#define MAXGRID 258   /* maximum size, with boundaries */
#define MAXWORKERS 16   /* maximum number of workers   */

void *Worker(void *);
void Barrier(int);

int gridSize, numWorkers, numIters, stripSize;
double maxDiff[MAXWORKERS];
double grid[MAXGRID][MAXGRID], new[MAXGRID][MAXGRID];
declarations of other global variables, such as barrier flags;

int main(int argc, char *argv[]) {
  pthread_t workerid[MAXWORKERS];  /* thread ids and */
  pthread_attr_t attr;             /* attributes     */
  int i; double maxdiff = 0.0;

  /* set global thread attributes */
  pthread_attr_init(&attr);
  pthread_attr_setscope(&attr, PTHREAD_SCOPE_SYSTEM);

  /* read command line arguments */
  /* assume gridSize is multiple of numWorkers */
  gridSize = atoi(argv[1]); numWorkers = atoi(argv[2]);
  numIters = atoi(argv[3]);
  stripSize = gridSize/numWorkers;

  initialize grids and flags for barrier;

  /* create the workers, then wait for them to finish */
  for (i = 0; i < numWorkers; i++)
    pthread_create(&workerid[i], &attr, Worker,
                   (void *) i);
  for (i = 0; i < numWorkers; i++)
    pthread_join(workerid[i], NULL);

  compute maxdiff and print results;
}
```

```
void *Worker(void *arg) {
  int myid = (int) arg;
  double mydiff;
  int i, j, iters, firstRow, lastRow;

  /* determine first and last rows of my strip */
  firstRow = myid*stripSize + 1;
  lastRow = firstRow + stripSize - 1;

  for (iters = 1; iters <= numIters; iters++) {
    /* update my points */
    for (i = firstRow; i <= lastRow; i++)
      for (j = 1; j <= gridSize; j++)
        new[i][j] = (grid[i-1][j] + grid[i+1][j] +
                     grid[i][j-1] + grid[i][j+1]) * 0.25;
    Barrier(myid);
    /* update my points again */
    for (i = firstRow; i <= lastRow; i++)
      for (j = 1; j <= gridSize; j++)
        grid[i][j] = (new[i-1][j] + new[i+1][j] +
                 new[i][j-1] + new[i][j+1]) * 0.25;
    Barrier(myid);
  }
  compute the maximum difference in my strip;
  maxDiff[myid] = mydiff;
}

void Barrier(int workerid) {
  /* details not shown */
}
```

Figure 12.1 Jacobi iteration using Pthreads.