MAT 450 — Operating Systems

Homework #2, Due on Wednesday, March 16, 2005.

In this homework you will be implementing Matrix Multiplication using pthreads. This is the programming project at the end of chapter 4 and starts on page 149. You should read the material on pages 149–152 before reading this material here.

Project Suggestions:

1. Matrix multiplication can be written as:
   $C_{i,j} = \sum_{n=1}^{K} A_{i,n} \times B_{n,j}$

2. The book demonstrates a simple example by defining M, K, N, A, B, and C on page 149.

3. In this case, the matrix C is the product of A and B. Each thread that we create will compute a single element of C as shown in the summation above. Thus, since C has M rows and N columns, there will be $M \times N$ worker threads.

4. The parent thread (*int main()*) will create these $M \times N$ worker threads, passing each worker the value of the component that is to be calculated. In the summation above, this is i and j. So each thread requires two parameters. Since we know the thread function only takes one argument of *(void *)*,we must create a struct. This struct is shown on page 150. Page 150 also demonstrates how to create the argument for the data structure. Your duty is simply to pass the variable *data* as the argument in the pthread creation function.

5. Keeping track of $M \times N$ thread identifiers (ids) can be annoying. Unlike the book, I suggest using a two-dimensional array to hold these thread ids. When you wish to *wait* for all the threads, you simply call a separate *pthread_join* on each thread id. This is demonstrated in the code below:

```
pthread_t workers[M][N];
...
for(int i = 0; i < M; i++)
    for(int j = 0; j < N; j++)
        pthread_join(workers[i][j],NULL);
```

6. What's left for you to do:

   (a) Write code to initialize the thread attributes (you can use the same *attr* variable in each *pthread_create*).

   (b) Write the exact *pthread_create* line when creating threads.

   (c) Write the thread function which computes the value of $C_{i,j}$ (hint: this should involve a for loop that executes K times as shown in the original summation.

   (d) Write code to output the matrix $C_{i,j}$ once calculation is complete.