ASSIGNMENT 3

Write a parallel *Pthreads* program computing the norm of the product of two **n×n** dense matrices on a **p**-processor SMP so that

- **p** threads are involved in the parallel computations.
- The 1-dimensional parallel algorithm of matrix multiplication is employed:
  - one of matrices is partitioned in one dimension into **p** equal slices
  - there is one-to-one mapping between the partitions and threads
  - each thread is responsible for computation of the corresponding slice of the resulting matrix
- Computation of the norm of the resulting matrix employs the *mutex* synchronization mechanism.

You can use BLAS or ATLAS for local computations.

Experiment with the program and build:

1) The dependence of the execution time of the program on the matrix size **n**.
2) The speedup over a serial counterpart of the program.

Explain the results.

Variants of the assignment:

1) Granularity of the program
   a. Two successive steps:
      i. Parallel matrix multiplication
      ii. Parallel computation of the norm of the resulting matrix
   b. One-step algorithm. No intermediate resulting matrix.
2) Partitioning scheme
   a. Left matrix is horizontally partitioned
   b. Right matrix is vertically partitioned
3) Matrix norm to be computed:
   a. The *maximum absolute column sum norm* (aka *one-norm*):

$$\|A\|_1 = \max_{0 \le j < n} \sum_{i=0}^{n-1} |a_{ij}|$$

   b. The *maximum absolute row sum norm* (aka *infinity-norm*):

$$\|A\|_\infty = \max_{0 \le i < n} \sum_{j=0}^{n-1} |a_{ij}|$$

==========