**Week 4 Assignment 4**

This week's assignment focuses on atomic and critical sections. Let's dive into the details:

1. Which one is not a synchronization construct? Leave it empty.
   - a. barrier
   - b. lock
   - c. atomic
   - d. sema

2. Which one is not a synchronization construct? Leave it empty.
   - a. barrier
   - b. lock
   - c. atomic
   - d. sema

3. The difference between atomic and critical sections is:
   - a. Atomic avoids race condition, critical section does not.
   - b. Both avoid race condition but critical section also avoids false sharing.
   - c. Atomic operations can be made to share a memory location, but critical sections cannot.
   - d. Atomic is parallelizable, but critical section is sequential.

4. In which matrix operation using OpenMP, performance reduction due to contention among threads will be highest?
   - a. Vector-vector dot product
   - b. Matrix-vector product
   - c. Matrix-matrix product
   - d. Cannot be said

5. Consider two operations: dot product of two n-dimensional vectors and matrix-vector product of an n-dimensional vector and n-dimensional matrix. For OpenMP parallelization, which one will give better speedup?
   - a. Depends on n
   - b. Matrix-vector product
   - c. Dot product
   - d. Cannot be said

6. Collapse nested or nested parallel loops in OpenMP can cause the overhead.
   - a. True
   - b. False

7. Which of these factors does not hinder good parallel performance for matrix-matrix multiplication?
   - a. High contention
   - b. Large amount of computing
   - c. Serial algorithm: Self may give poor performance due to race/competition access to one of the matrices, which is cached
   - d. Serial algorithm itself may give poor performance in large amount of data for each matrix multiplication, as to be used for computing one product element

8. How can a MPI process identify its own rank?
   - a. Using `MPI_Comm_rank`
   - b. Using `MPI_COMM_WORLD`
   - c. Specific to the code
   - d. None of the above

9. An MPI process has to send data with another specific process. How can this be done?
   - a. Adding a thread with its own rank
   - b. Communicator automatically takes care of it
   - c. MPI function call
   - d. It has to be specified in the program

10. An MPI program runs in the number of processes specified by the execution command. Each of the processes manipulates all lines of the program code specified by `MPI` statement.
    - a. True
    - b. False