Introduction to MPI - Worksheet 1
(See WebCT-HPC Course)

1. What are the two basic memory architectures in parallel programming?

2. Briefly describe the difference of each type.

3. The number of processors is most limited in which type of architecture? Why?

4. “Domain decomposition” of a problem is also known as “data parallelism”. SPMD follows this model where the code is identical on all processors. What does SPMD stand for?

5. In the Poisson equation example, what is this doing to u(i,j)?
   u(i,j) = 0.25*(u(i,j-1) + u(i,j+1) + u(i-1,j) + u(i+1,j))

6. If i and j range from 1 to 32, how does this example split up the processing among 4 processors?

7. Another technique to use besides domain decomposition is functional decomposition, also known as “task parallelism”. Briefly describe task parallelism.

8. The main goal of parallel programming is (as compared to serial programming):

9. Briefly describe load balancing.

10. If you had N processors, what would be the ideal fraction of computational time for a parallel program as opposed to a serial program?

11. The cost of communication time is measured in terms of ________ and _________. Briefly describe each.

12. A shared memory computer has access to:
A. the memory of other nodes via a proprietary high-speed communications network
B. a directives-based data-parallel language
C. a global memory space
D. communication time

13. A domain decomposition strategy turns out not to be the most efficient algorithm for a parallel program when:
A. data can be divided into pieces of approximately the same size.
B. the pieces of data assigned to the different processes require greatly different lengths of time to process.
C. one needs the advantage of maintaining a single flow of control.
D. one must parallelize a finite differencing scheme.

14.A domain decomposition strategy turns out not to be the most efficient algorithm for a parallel program when:
A. data can be divided into pieces of approximately the same size.
B. the pieces of data assigned to the different processes require greatly different lengths of time to process.
C. one needs the advantage of maintaining a single flow of control.
D. one must parallelize a finite differencing scheme.

15.Total execution time does not involve:
A. computation time.
B. compiling time.
C. communications time.
D. idle time.

16.One can minimize idle time by:
A. occupying a process with one or more new tasks while it waits for communication to finish so it can proceed on another task.
B. always using nonblocking communications.
C. never using nonblocking communications.
D. frequent use of barriers.