UPES

UNIVERSITY OF PETROLEUM & ENERGY STUDIES
DEHRADUN
Final Semester Examination – December, 2017

Program/course:	M.Tech CFD	**Semester –**	3
Subject:	High Performance and Parallel Computing Applications for CFD	**Max. Marks**	: 100
Code :	**MCFD 804**	**Duration**	: 3 Hrs
No. of page/s:	4		

SECTION A – 40 points

1) **Write 5 aspects of a parallel overhead: (5 pts)**

2) **Mark as True or False: (20 pts)**
   - In MISD single data stream is fed into multiple processing units.
   - Virtually all parallel tasks will have sections of a parallel program that must be executed serially.
   - Synchronization usually involves waiting by at least one task, and can therefore cause a parallel application's wall clock execution time to decrease.
   - MPI is now the "de facto" industry standard for message passing.
   - Data Parallel Model is not suitable if you have to perform the same operation on the whole set.
   - In SPMD, a single program is executed by all tasks simultaneously
   - In MPMD all tasks can use different data.
   - Fibonacci series is a good example of a parallelizable program
   - For parallel efficiency Observed Speedup should always be more then 1
   - Fine granularity means more communication events will happen instead of communication.
   - Functional Decomposition works well for ecosystem modeling.
   - 3d Heat Problem Parallelization doesn't require much data communications.
   - CFD applications are generally good candidates for parallel processing and parallel programming.
   - Asynchronous communications are also referred as Blocked Communications.
   - Gathering is a collective communications task.
   - When the last task reaches the barrier, all tasks are synchronized.
   - Coarse grain parallelism is very effective for load balancing.
   - Less I/O means a more efficient parallel programme.
   - Relaxation Technique for the solution of the Laplace Equation as a CFD technique is a very good candidate for parallelization as it is data independent.
   - Mccormack Technique is a good candidate for parallel processing.

3) Fill in the Blanks (5 pts)

a) _____ is the logically discrete section of computational work.

b) _____ Refers to a parallel system's (hardware and/or software) ability to demonstrate a proportionate increase in parallel speedup with the addition of more processors.

c) _____is the extension to FORTRAN 90 for parallel processing.

d) _____ states that potential program speedup is defined by the fraction of code (P) that can be parallelized

e) _____ refers to the practice of distributing work among tasks so that *all* tasks are kept busy *all* of the time.

4) Explain which applications and techniques in CFD are good candidates for parallel processing and for parallel programming. Explain each in detail and give specific examples of bottlenecks which can be solved with parallelization (10 pts)

SECTION B (50 points – All Questions are Mandatory)

5) Explain the data architecture below. What does it signify? How is the architecture formed? In what types of computational situation will it be advantageous? In what cases would it be disadvantageous? Explain how it can be implemented in CFD problems? (10 pts)
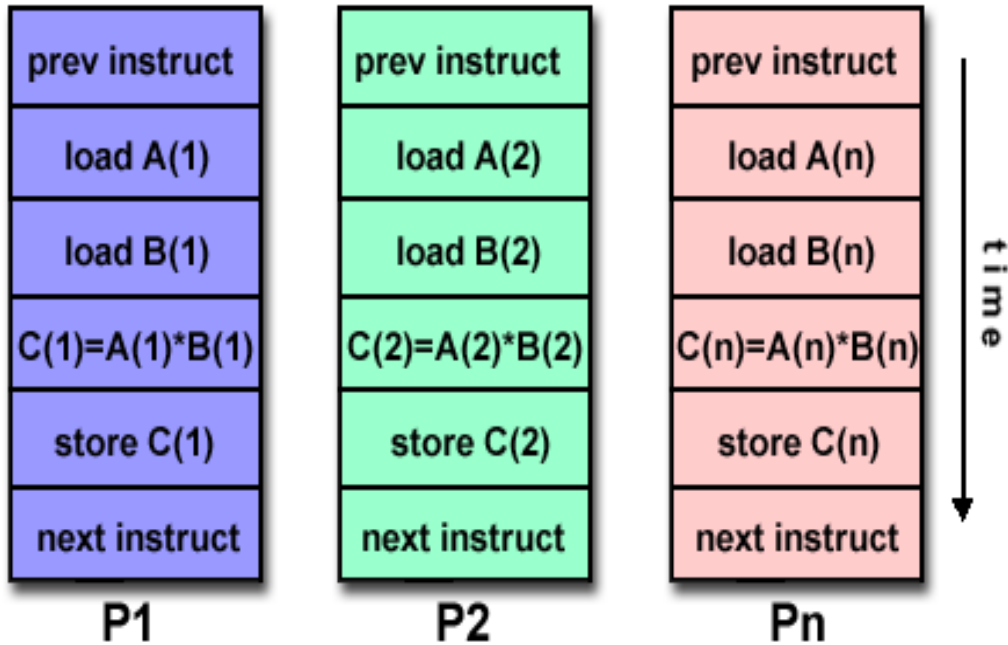
communication

computation
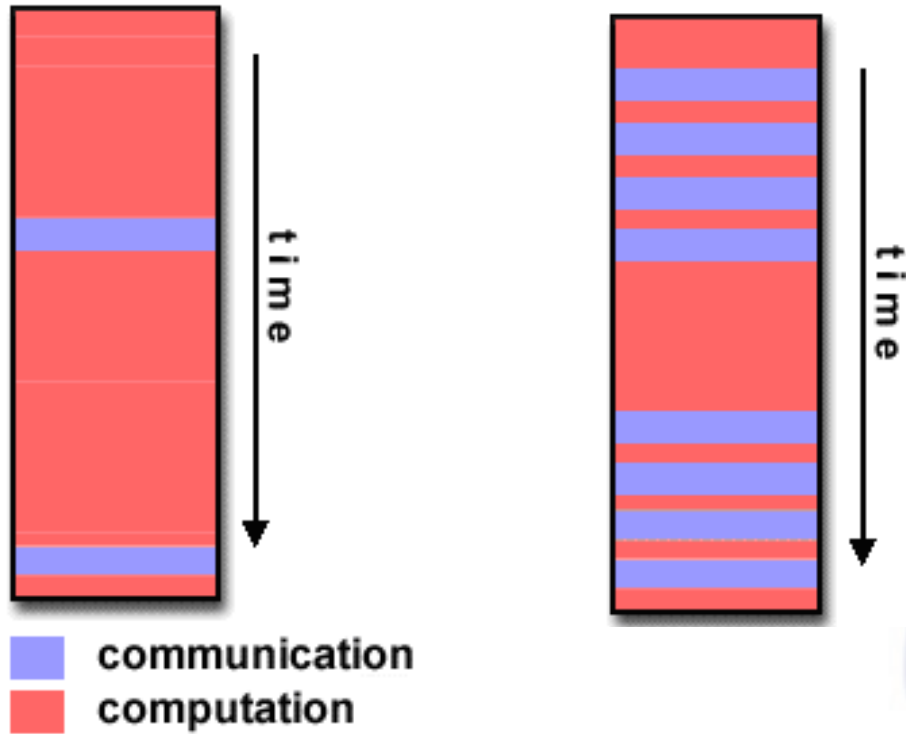
**6)** Explain the graph below and how it relates to parallelization. (10 pts)
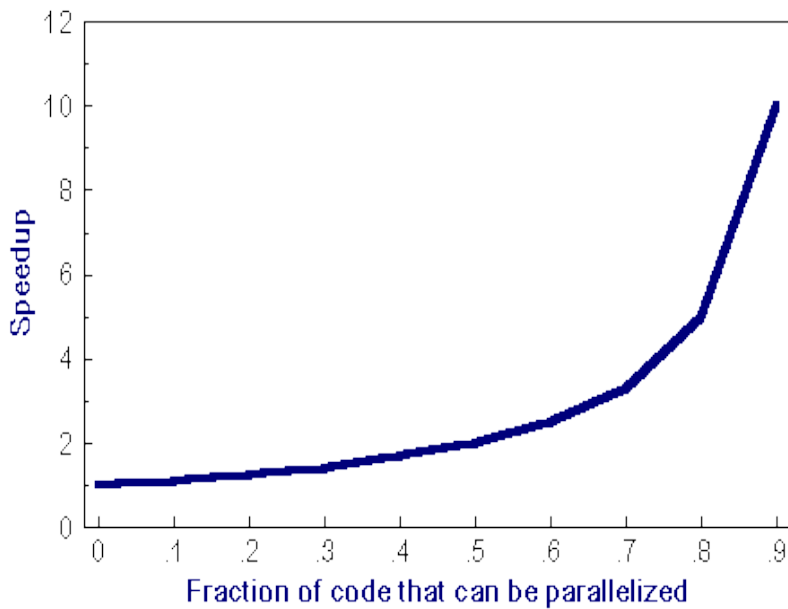
7) Explain the parallel processing concept below in relation to a CFD problem. Give a specific example of a CFD Numerical Method or a solution algorithm where this particular
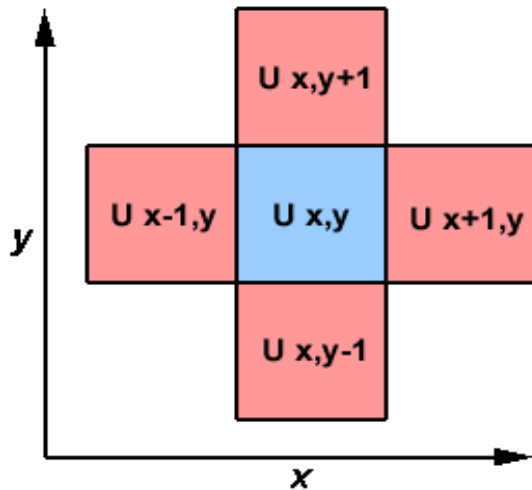
concept can be applied (10 pts)

8) Write the parallel algorithm for the 1 D Wave Equation given below? Which method of parallelization can you use for the solution of this problem? (10 points)
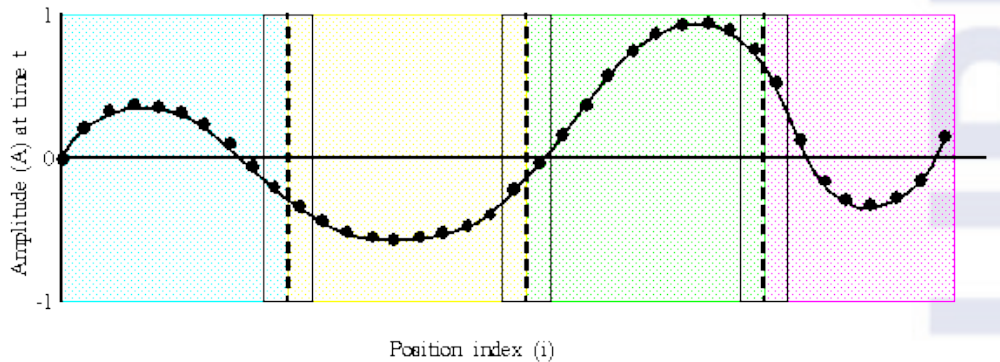

```
A(i,t+1) = (2.0 * A(i,t)) - A(i,t-1)
         + (c * (A(i-1,t) - (2.0 * A(i,t)) + A(i+1,t)))
```

9) For the heat equation and for the grid given below, devise a parallel processing scheme and write the algorithm. (10 points)

Where:

$$U_{x,y} = U_{x,y}$$

$$+ C_x * (U_{x+1,y} + U_{x-1,y} - 2 * U_{xy})$$

$$+ C_y * (U_{x,y+1} + U_{x,y-1} - 2 * U_{x,y})$$

SECTION C (10 points – All Questions are Mandatory)

10) **Explain about OpenMP in detail and give examples of its use, as to how to start it and hoe to identify it and what type of operating systems and programming languages does it work with? (10 points)**