

## Assignment #1

Due: 3/4/2012

### 1. An Application of Parallel/Distributed/Grid/Cloud Computing

- Document an application of parallel/distributed/grid/cloud computing on solving challenging problems in the past two years (2010, 2011) . Your document should minimally include the following sections:
  - Problem description
  - Difficulties (or why using the technology selected)
  - Computing facility
    - No. of processors, amount of memory, ...
  - Computational resources used
    - How long did it take to solve the problem
    - Amount of memory consumed, amount of disk space used, etc.
  - Reference
  - Results (e.g. animations, images)

2

### 1. An Application of Parallel/Distributed/Grid/Cloud Computing

- Hint: the following URLs may help
  - <http://www.top500.org>
  - <http://www.psc.edu/>
  - <http://www.sdsc.edu/>
  - <http://www.osc.edu/>
  - <http://www.ncsa.uiuc.edu/>
  - Or other supercomputer centers in the world
- An example is given at <http://yo-1.ct.ntust.edu.tw/Courses/PA/01-Example/>

3

### 2. Programming warmup

- a) Please review/preview the following programming concepts in C/C++:
- a) **Pointer**
    - <http://cplusplus.com/doc/tutorial/pointers/>
  - b) **Preprocessor directives**
    - <http://wwwcplusplus.com/doc/tutorial/preprocessor/>
  - c) **Dynamic 1D and 2D array allocation, use, and deallocation**
    - <http://randu.org/tutorials/c/dynamic.php>
    - <http://www.learncpp.com/cpp-tutorial/65-multidimensional-arrays/>
    - <http://www.parashift.com/c++-faq-lite/freestore-mgmt.html#faq-16.16>

### 2. Programming warmup

- b) For each of the three topics (pointer, pre-processor directive, and 1D/2D memory management), give at least one example source code. For each of the source code you hand in, please give proper comments showing your understanding on the topic.

### 3. Reading Assignment

- Chapter 2, "Thinking Parallel", of "Intel Threading Building Blocks: Outfitting C++ for Multi-core Processor Parallelism".
- After reading, you should be able to understand:
  - Difference between: 1) data parallelism, 2) task parallelism, and 3) pipelining.
  - Definition of "embarrassingly parallel"?
  - Speedup and scaling
  - Amdahl's Law vs. Gustafson's observations
  - The difference between a thread and a process?
  - Synchronization
  - *Threads: thread-safe, mutual exclusion and locks, deadlock, race condition.*
  - *Abstraction, Patterns*
- **You will be asked these questions later in the course ...**

6