Parallel Programming for High-Performance Applications
22 January 2015
Department of Computer Science, Faculty of Sciences

The exam has 9 questions. Your answers should be to the point: address the questions and omit information that is not asked for. The grading system is shown after the last question.

1. (a) What is Moore's law (from 1975)?
   (b) Someone claims that Moore's law currently no longer holds because of the "power wall". Comment on whether this claim is true or false.

2. What are the diameter and bisection width of a 3-dimensional mesh (lattice) topology that has 64 nodes in total?

3. What would change if Floyd's All pairs Shortest Paths (ASP) algorithm would be parallelized by distributing the rows *cyclically* instead of blockwise over the different machines? Would the communication pattern and/or performance change?

4. (a) What is a *reduction operation* in MPI? With which kinds of operators can it be used?
   (b) What is a *communicator* in MPI? Why does MPI need this construct?

5. Automatic parallelization of sequential programs is extremely difficult. Languages that try to do (more or less) automatic parallelization therefore make compromises, such as

   - They make restrictions on the source program
   - They restrict the kind of parallelism that can be used
   - They use a semi-automatic approach and let the programmer still do part of the work.

   Discuss which compromises or restrictions HPF(High Performance Fortran) makes.

6. The parallel Barnes-Hut N-body simulation algorithm can exploit the fact that the simulated system (e.g. a galaxy) changes only gradually in each timestep (e.g., stars don't move from one end of the galaxy into another end of the galaxy in one timestep). Explain how the Barnes-Hut algorithm exploits this knowledge to reduce the communication overhead and load imbalance of the parallel program.

7. CPU A has 4 scalar cores and hyperthreading (2x). CPU B has 4 vector cores, each able to perform 4-lane SIMD. Their clock frequencies, memory hierarchies and memory bandwidths are the same. The system uses a compiler that does an optimal job at parallelizing and vectorizing simple loops like the ones shown below.

   (a) What is the peak performance ratio between CPU A and CPU B?

   (b) Which one of the A and B processors will be faster for the following loop? Why?

```
int tmp = 0;
for (i=0; i<100; i++)
  tmp = tmp + i*i;
```

   (c) Which one of the A and B processors will be faster for the following loop? Why?

```
int tmp = 0;
for (i=0; i<100; i++)
  if (i % 2 == 0)
    tmp = tmp + i*i;
  else
    tmp = i+7;
```

   (d) Which one of the A and B processors will be faster for the following loop (the array is simply an array of integer numbers)? Why?

```
int tmp = 0;
for (i=0; i<100; i++)
  if (array[i] % 2 == 0)
    tmp = tmp + i*i;
  else
    tmp = i+7;
```

8. (a) Give three examples of programming practices (i.e., programming constructs or operations) that negatively affect the GPU performance. Explain why they negatively affect performance.

   (b) Consider an application that adds two arrays on the GPU and writes the result to a third array. The arrays are 2GB each. The host takes 1s to compute the vector addition, the GPU takes 40ms. What should the PCI/e bus bandwidth be (i.e., the bandwidth of the connection between the CPU and the GPU) such that the GPU version obtains better wall clock performance?

9. ACME is a company that has just bought a new multi-core processor with 8 cores. Each core is capable of 4-way SIMD operations, and operates at 1GHz. The company has a weather prediction sequential application, Cloudy, which computes the weather prediction in the form of 4 floating point numbers per cell. Cloudy uses 1 core and runs in 10s for a grid of 1024 x 1024 cells. The kernel to be parallelized takes 80% of the sequential execution time, and has an Arithmetic Intensity (AI) of 4. Bill does an internship at ACME and he has to parallelize Cloudy. Help Bill answer some of his performance questions:

   (a) What is the peak performance of the processor?

   (b) Assuming Bill can use all 8 cores, but he doesn't know how to apply vectorization, how many processors should he use to get the kernel execution time below 0.3s ? How about getting the full execution time below 0.3s?

   (c) Knowing the bandwidth of the processor is 6GB/s, should Bill apply vectorization to his code (in the hope of performance improvement)? If yes, what is the expected peak performance improvement?

Points

1a	1b	2	3	4a	4b	5	6	7a	7b	7c	7d	8a	8b	9a	9b	9c
5	5	10	10	5	5	10	10	2	3	3	2	5	5	3	3	4

Total: 90 (+ 10 = 100)