CSC/ECE 506: Architecture of Parallel Computers
Problem Set 1
Due Wednesday, January 20, 2015

Problems 1, 2, and 4 will be graded. There are 80 points on these problems. *Note:* You must do *all* the problems, even the non-graded ones. If you do not do some of them, half as many points as they are worth will be subtracted from your score on the graded problems.

Problem 1. *(15 points)* Calculate the [integer] number of processors needed to achieve the desired speedup. Suppose a program has 84 seconds of work that is parallelizable and 16 seconds of work that is not.

(a) How many processors are required to execute the program twice as fast?

(b) How many processors are required to execute the program 3 times as fast?

(c) How many processors are required to make the execution 5 times faster?

(d) II a speedup of 7 achievable? Why or why not?

(e) Assuming a machine has 32 processors, what is the maximum percentage of the program that could be serial to achieve a speedup of 10?

Problem 2. *(20 points)*
Consider the following C program:

```
1:  for (i = 1; i <= 1024; i++) {
2:          sum[i] = 0;
3:          for (j = 1; j <= i; j++)
4:                  sum[i] = sum[i] + i;
5:  }
```

Assume the statements 2 and 4 each take two machine cycles and ignore the overhead due to software loop control (statements 1, 3, and 5) and all other system overhead (IPC, stalling, etc.) and resource conflicts.

(a) What is the total execution time of the program on a uniprocessor system?

(b) Using a simple partitioning scheme where i-loop iterations are divided among 32 processors (i.e. processor 1 executes iterations i = 1 to 32, processor 2 executes iterations i = 33 to 64, and so on), determine the total execution time and the speedup factor compared to the results of part (a). Note that this leads to an unbalanced computational workload among the processors.

(c) Modify the partitioning scheme to facilitate a balanced parallel execution of all the computational workload over 32 processors. What are the minimum execution time and speedup factor over the uniprocessor execution resulting from the balanced parallel execution on 32 processors?

Problem 3. *(25 points)* Suppose that a program takes $n^3 + n^2$ units of time to execute sequentially.

(a) Supposing that the $O(n^{2)}$ work may be done in parallel, and that we have p processors, what is the maximum speedup?

(b) As $p \rightarrow \infty$ in part a, the highest speedup approaches what value? *Hint:* Consider L'Hôpital's rule.

(c) Suppose that the $O(n^3)$ work may be done in parallel, and that we have p processors, what is the maximum speedup?

(d) As $p \rightarrow \infty$ in part (c), the highest speedup approaches what value?

(e) Explain why there is such a large discrepancy between the answers for speedup between part (b) and part (d).

Problem 4. *(40 points)* Watch two videos (http://tinyurl.com/506q4a-video and http://tinyurl.com/506q4b-video) and take the embedded quizzes (you need to enter your e-mail address to receive credit for the quizzes). The videos are from Wen-Mei Hwu's "Introduction to Heterogeneous Parallel Computing" MOOC. They cover data-parallel programming and the CUDA GPU architecture. Notes are available at http://tinyurl.com/506q4a-slides and http://tinyurl.com/506q4b-slides. By adjusting the settings wheel in the lower right-hand corner of the video player, you can speed up or slow down the video.