Concurrent and Parallel Systems Mock Exam

December 2014

1. Concurrent and Parallel Systems Design and Theory

   (a) Discuss what a task dependency graph is and how it can be used in parallel application design.

   [4]

   (b) For the following table produce the task dependency graph and identify places of possible parallel speedup.

Task	**Computation Time**	**Depends on**
1	5 minutes	-
2	10 minutes	1
3	10 minutes	2
4	20 minutes	3
5	5 minutes	2
6	5 minutes	2
7	10 minutes	2
8	20 minutes	2
9	10 minutes	4, 8
10	30 minutes	5, 6, 7, 9

   [8]

   (c) Using the answer from part (b) calculate the serial time of the original application and the parallel time, speedup, and efficiency of your designed solution assuming that you have 5 processors.

   [4]

   (d) How can we determine the scalability of a program based on the number of processors and problem size? What is meant by strongly scalable and weakly scalable?

   [4]

Total for Question 1: 20

2. Multithreaded Applications and Single CPU Parallelism

   (a) Discuss the general lifecycle of a thread.

   [4]

   (b) Discuss the two main types of scheduling in a multithreaded system, including their advantages and disadvantages.

   [6]

   (c) Describe how mutexes and futures operate to support multithreaded applications, highlighting their different uses.

   [5]

   (d) What is meant by the term *context switch* in multithreaded applications and discuss why they should be avoided as much as possible.

   [5]

   Total for Question 2: 20

3. Distributed Parallelism.

   (a) Discuss what is meant by the terms *CPU-bound* and *I/O-bound* in the context of distributed parallelism.

   [4]

   (b) MPI is a framework that provides communication between multiple machines to support parallel problem solving. It supports a number of communication models to achieve this. Describe what is meant by *map-reduce* and *scatter-gather* in MPI and how they can be used to solve particular parallel problems.

   [6]

   (c) Discuss the advantages and disadvantages when working with distributed parallelism.

   [6]

   (d) Discuss what it meant by the acronym *SPMD* and how this relates to a framework such as MPI.

   [4]

   Total for Question 3: 20

4. General Purpose GPU Programming

   (a) Discuss the different concerns for GPGPU programming in comparison to CPU parallelism when considering memory.

   [4]

   (b) Discuss the limitations of using the GPU for parallel processing.

   [4]

   (c) Provide a diagram that illustrates how the GPU is designed from the processor and memory point of view.

   [6]

   (d) Describe, with illustrations, how to perform vector addition on the GPU, given arbitrarily (but equal) length vectors. Include the general algorithm required by the GPU kernel.

   [6]

Total for Question 4: 20

Question	Points	Score
1	20	
2	20	
3	20	
4	20	
Total:	80	