Exam

ECE 742 – Parallel Computer Architecture, Spring 2010

Instructor: Onur Mutlu

TA: Chris Fallin

April 9, 2009

NAME: _____

Problem	Points	Score
Name	2	
1	98	
2	15	
3	30	
4	35	
5	35	
6	20+20	
7	15	
Total	250 + 20	

- **READ THE QUESTIONS VERY CAREFULLY.**
- **Please write your name on *every* sheet (2 points).**
- This is a closed book exam. You are not allowed any cheat sheets.
- No electronic devices may be used.
- This exam lasts 1 hour 59 minutes.
- If you make a mess, clearly indicate your final answer. Clearly indicate what should be graded.
- For questions requiring brief answers, provide brief answers. You will be penalized for verbosity and irrelevance.
- Please show your work and reasoning. We cannot give you partial credit if you do not clearly show how you arrive at an answer.

Name: _____

Problem 1 (Short answers) [98 Points]

Please be concise, clear, and to-the-point in all your answers.

i) Amdahl's Law [6 points]

Amdahl's abstraction of a parallel program was two portions, one in which the program is completely serial (the serial portion) and the other in which the program is running completely in parallel (the parallel portion). In practice, the parallel portion doesn't achieve perfect speedup. Why is this so? Give three reasons, no more than 10 words each:

1. _____
2. _____
3. _____

ii) Coupling Multiple Processors [5 points]

What key characteristic differentiates loosely coupled and tightly coupled multiprocessors? Explain (in less than 10 words).

iii) ROCK Runahead [6 points]

What is the purpose of runahead execution (or hardware scouting) employed in the Sun ROCK processor? In less than 10 words:

What does runahead execution achieve that multiple threads run on separate hardware contexts cannot achieve? In less than 10 words:

Name: _____

iv) Combining Operations [8 points]

What is the idea of "combining operations" in the network? Why could it be useful? Explain in no more than 40 words.

What kind of network topology is particularly suitable for combing operations? Give an example. Explain in no more than 20 words.

v) Network Contention [10 points]

Assume we have a 2-dimensional mesh network. When multiple packets arriving from different input ports need to go to the same output port, we say the output port is being contended for. There are three general ways to handle this contention. First is to _____ some of the packets, second is to _____ some of the packets, third is to _____ some of the packets.

Assuming contention is extremely rare, which option would you choose? Explain why, clearly, in less than 20 words.

vi) Locks (6 points)

Explain two reasons why a lock may not be necessary during dynamic execution even though it is required statically.

1. _____

2. _____

vii) Dataflow (6 points)

What does a dataflow implementation provide that is difficult to provide any other way? Explain in less than 20 words.

Name: _____

viii) Dependencies in Speculative Parallelization [6 points]

A speculative parallelization mechanism breaks a long sequential program into multiple tasks, e.g. as we saw in TLS, Multiscalar, Dynamic Multithreading. Tasks can be dependent on each other via registers or memory locations. What is the **key difference** between inter-task register and memory dependences in terms of how they are handled?

ix) Multiscalar [7 points]

A multiscalar processor can reduce the branch misprediction penalty compared to a superscalar processor with a large sequential instruction window. TRUE or FALSE? Circle one.

Explain your reasoning in less than 30 words.

x) Cache Coherence [6 points]

What is the advantage of the Illinois protocol (i.e., the addition of the Exclusive state) over the MSI cache coherence protocol? Be brief (20 words at most).

xi) Parallel Performance Analysis [10 points]

We measure the performance of a parallel application with 4 threads on 4 cores, finding its execution time is 125 seconds. Then, we measure the performance of the same application with 32 threads on 32 cores, finding its execution time is 155 seconds. Assuming there were no software changes involved, describe how this is possible. (Assume each core can execute only one thread)

We then measure the execution time of the application with a single thread on a single core machine. Can the execution time we find be 25 seconds? How? Describe your reasoning.

Name: _____

Later, we measure the execution time of the same application with 4 threads on a 4-way multithreaded processor. We find the execution time of the application to be 300 seconds. Is there anything interesting about the result? Explain why this could be the case.

xii) Routing (6 points)

a. What is the key difference between an oblivious routing algorithm and an adaptive routing algorithm? Explain in 10 words or less.

b. Remember that Valiant's algorithm performs routing in two steps. First, it routes the packet to a randomly chosen intermediate node via dimension order routing. Next, the packet is routed to the destination node from the intermediate node, again via dimension order routing.

Is this algorithm oblivious or adaptive? Circle one.

Is this algorithm minimal or non-minimal? Circle one.

c. Give an example of a non-minimal adaptive algorithm, in less than 10 words.

xiii) Dataflow Strikes Back (8 points)

Two "traditional control flow" constructs that are not possible to handle in a static dataflow architecture are _____ and _____. Explain why, in less than 20 words:

Dynamic dataflow architectures solve this problem by using _____. Explain how, in less than 20 words.

Name: _____

xiv) Cache Coherence Again (8 points)

A designer has built a directory-based cache coherent system using the MESI invalidation protocol. Under certain workloads, however, the system performs very poorly, and after a simulation run and closer examination, the designer finds that there is a constant stream of invalidation requests between four of the nodes for one particular cache block. What is this phenomenon called? (no more than 5 words)

Where and how is the problem best solved?

Name: _____

Problem 2 (Asymmetry) [15 Points]

An enthusiastic software engineer wrote a parallel program which spends a large fraction of its execution time in Amdahl's serial bottleneck (serial portion) when executed on a symmetric multi-core processor with 16 small, in-order-execution cores, each running at 1GHz. To improve the performance of the program by speeding up the serial portion, the engineer uses two different asymmetric multi-core processors, both of which are area equivalent to the 16-core symmetric machine.

First, she runs the program on a dynamically asymmetric multi-core processor that executes serial sections on one of the cores at 2GHz. She finds that the program speeds up by 1%.

Next, she runs the program on a statically asymmetric multi-core processor that has 12 small cores and one large, out-of-order-execution core. Each core executes at 1GHz. She finds the program speeds up by 10%.

What could be the reason for the higher speedup on the statically asymmetric multi-core processor? Explain in enough detail. (Assume there are no faults and the program runs correctly)

Name: _____

Problem 3 (Simultaneous Multithreading) [30 Points]

Two designers for a microprocessor company found out that long latency load instructions were causing performance problems in the new 2-way SMT processor they architected. Explain why this could be a problem.

To reduce the performance problem, the designers came up with a creative solution, after days of experimenting and sleepless nights (and mounds of coffee, of course). Their idea was to flush the subsequently fetched instructions of a thread immediately after a load instruction is found to be a data cache miss. Fueled by the "Eureka moment" excitement of their new idea, they spent a few more sleepless nights and trillions of simulation cycles (not to mention the more mounds of coffee, of course) to test this idea on a suite of 1000 2-way multiprogrammed workloads. They found the idea improved performance (weighted speedup, harmonic speedup) in 700 workloads, but it degraded performance in 300, compared to the baseline case of SMT where no flushing is performed after a load miss.

Explain why and when the engineers' idea could be a good idea. (for performance; give as many reasons as makes sense)

Explain why and when the engineers' idea could be a bad idea. (for performance; give as many reasons as makes sense)

Name: _____

What could be the characteristics of the workloads that lost performance with the new idea? (assume each workload consists of two single-threaded applications)

You are a colleague of the engineers, who has listened to the discussions about alternative ways of solving the problem. Suggest one way of improving the proposed idea (feel free to use the ideas we have discussed and read about in class). Describe your technique.

What are the advantages and disadvantages of your technique? Is your technique strictly better than the engineers' proposal?

Name: _____

Problem 4 (Higher Performance Locking) [35 Points]

In lecture, we have discussed at least two approaches to improving the performance of lock-based synchronization: Accelerated Critical Sections (ACS) and Speculative Lock Elision (SLE). Compare and contrast these two following the questions below:

i. What is the basic idea of ACS? (less than 20 words)

ii. What is the basic idea of SLE? (less than 20 words)

iii. Which approach, if any, requires the programmer to modify their programs?

iv. Give at least two reasons why ACS can provide better performance than SLE. Be concise, but specific, describing any conditions that lead to this result.

Name: _____

v. Give at least two reasons why SLE can provide better performance than ACS. Be concise, but specific, describe any conditions that lead to this result.

vi. Assume each instance of a heavily contended, long critical section updates the same element in a shared data structure. Which approach, if any, would you expect to perform the best? Why?

vii. Assume each instance of a heavily contended, long critical section updates disjoint elements in a shared data structure. Which approach, if any, would you expect to perform the best? Why?

Name: _____

Problem 5 (Memory Scheduling) [35 Points]

In class, we examined three memory scheduling techniques that can be employed when DRAM is shared between multiple threads: the commonly used FR-FCFS (or row-hit first) scheduler which aims to optimize for DRAM throughput, the stall-time fair memory scheduler (STFM) and the parallelism aware batch scheduler (PAR-BS). In this question, you will compare and contrast the three approaches.

i. How does the FR-FCFS (row-hit first) scheduling algorithm work? No more than 20 words.

ii. What are the disadvantages of FR-FCFS in a multi-core system?

iii. Explain the key idea of STFM algorithm. No more than 20 words.

iv. Explain the key idea of the PAR-BS algorithm. No more than 20 words.

Name: _____

v. What are the advantages of STFM over PAR-BS?

vi. What are the advantages of PAR-BS over STFM?

vii. You find that the FR-FCFS algorithm provides higher DRAM throughput (data bytes transferred across DRAM bus per second) than the STFM algorithm, but STFM provides higher system throughput (total number of instructions retired in the entire system over a period of time) than FR-FCFS. Explain how this is possible.

Name: _____

Problem 6 (Design Question - Smart resources, dumb resources) [20+20 Points]

In class, we have discussed solutions to reducing inter-thread interference in shared multi-core resources by designing control mechanisms in resources (e.g. caches, on-chip network, and memory controllers) that aim to reduce inter-thread interference. The goal of these mechanisms was to improve fairness and performance – let's call this approach the "smart resources" approach. An alternative, which we briefly discussed, is to keep resources simple and "dumb" (as they are in existing system) without such control mechanisms, but control the rate at which each core can inject in the resources. Let's call this latter approach "source throttling" as we did in the lecture.

What are the advantages of "smart resources" over "source throttling"?

What are the advantages of "source throttling" over "smart resources"?

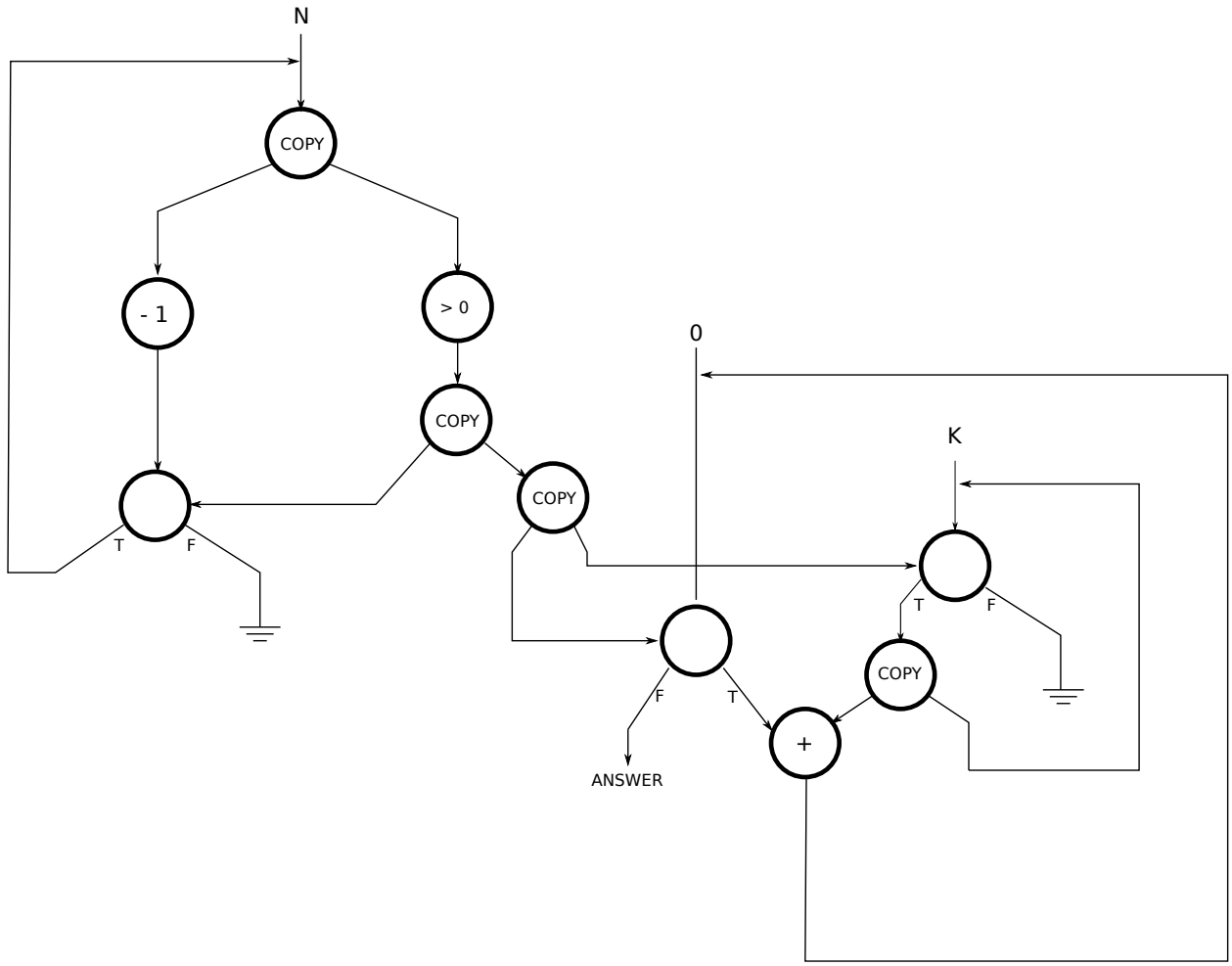
BONUS (20 points):

Can both approaches be combined to provide higher performance than either approach alone? If so, describe how you would go about combining them. Explain your ideas clearly.

Name: _____

Problem 7 (Dataflow) [15 Points]

The following data flow graph requires three non-negative integer input values 0 (zero), N, k as shown. It produces a value "Answer", also shown.



What function does this data flow graph perform (in less than ten words, please)?