Read me!

I made these slides before I started to work on the quiz itself. Therefore, there’s no guarantee that every topic covered on the quiz is listed here, or that every topic listed here is covered on the quiz. These points are just things that caught my eye as I ran through the lecture notes, but if you’re comfortable with all of this material, you will probably do well. Feel free to send questions to 6.172-staff@mit.edu. The quiz will be 90 minutes, and will be closed-book, closed-notes except for your single-sided, handwritten 8.5”x11” crib sheet.
Quiz 1 Review Session

6.172 Fall 2009
Kevin Kelley
Quiz 1 Review Session

• Everything we’ve done so far
  – Nine lectures
  – Two projects

• Other skills
  – Read/write C
  – Read/write simple assembly
L1 – Matrix Multiply: A Case Study

• Why is performance important?
• How do we go about improving performance?
• What factors affect performance?
  – Algorithmic decisions
  – Data structures
  – Cache and branch performance
L2 – Basic Performance Engineering

• Tradeoffs in data structures
  – Space for time
  – Time for space
  – Space and time
L2 – Basic Performance Engineering

• Loop modifications
  – Lift loop invariants
  – Unroll loops
  – Use sentinel tests
  – Fuse loops
L2 – Basic Performance Engineering

• Logic transformations
  – Reorder tests
  – Precompute logic functions

• Procedure transformations
  – Inline small function calls
  – Eliminate tail recursion

• Other
  – Eliminate common subexpressions
  – Pair function calls
  – Implicit parallelism
L3 – Performance Analysis

• Statistical sampling
• Performance counters
  – CPI
  – Branch misprediction
  – L1 icache misses
  – L1 dcache misses
• Dynamic instrumentation
L4 – From C to Assembler

• General machine model
• Stack, heap, etc.
• General mechanics for a function call
• Registers vs. memory
  – How registers affect cache behavior
L5 – Computer Architecture and Performance Engineering

• Pipelining and hazards
  – Structural hazards
  – Data hazards
  – Control hazards

• WAR vs. RAW hazards
  – Register renaming (both compiler and HW-level)

• SIMD instructions

• Branch prediction and stalling
L6 – Memory Systems

• Memory system performance
  – Temporal locality
  – Spatial locality
  – Levels of the hierarchy

• Types of misses
  – Cold miss
  – Capacity miss
  – Conflict miss
  – True sharing miss
  – False sharing miss
L6 – Memory Systems

• Blocking
• Memory access patterns
• Analysis of cache behavior
• Cache coherency
• Write-back caches
L7/8 – Algorithms and Data Structures for Caches

• Ideal cache model
  – Cache size M; block size B
  – Fully associative with optimal replacement
  – Can be relaxed to LRU replacement policy

• Tall-cache assumption

• Voodoo tuning parameters

• Divide-and-conquer algorithms

• Recursion trees
L7/8 – Algorithms and Data Structures for Caches

- Cache-oblivious algorithms
- Binary vs. k-ary merging
- Funnel-sort
L9 – Dynamic Storage Allocation

• Stack allocation
• Heap allocation
  – Fixed-size allocation
  – Binned free lists
• Issues
  – External fragmentation
  – Internal fragmentation
  – Coalescing
L9 – Dynamic Storage Allocation

• Garbage collection
  – Reference counting
  – Copying garbage collection
    • Forwarding pointers
  – Mark and sweep, generational, etc.
Questions?

• Basic Performance Engineering
• From C to Assembly
• Performance Analysis
• Computer Architecture
• Memory Systems
• Algorithms and Data Structures for Caches
• Dynamic Storage Allocation