

CDA5155/CDA4150 Exam 1 Study Topics

Covers Chapter 1, Appendix B, Chapter 2, Appendix A, and Assignment 2

1. Chapter 1
 - 1.1. Classes of Computers
 - 1.2. Trends
 - 1.2.1. Implementation Technology
 - 1.2.2. Bandwidth over Latency
 - 1.2.3. Need for Energy Efficiency
 - 1.3. Measures
 - 1.3.1. Comparing Performance
 - 1.3.2. Clock Speed
 - 1.3.3. Methods for Evaluation
 - 1.4. Principles
 - 1.4.1. Locality of Reference
 - 1.4.2. Amdahl's Law
 - 1.4.3. CPU Time
2. Appendix B
 - 2.1. Memory Hierarchy Principles
 - 2.1.1. Terms
 - 2.1.2. Equations
 - 2.1.2.1. Miss Rate
 - 2.1.2.2. Hit Rate
 - 2.1.2.3. Average Memory Access Time
 - 2.1.3. Evaluation Methods
 - 2.2. Caches
 - 2.2.1. Organizations
 - 2.2.1.1. direct mapped
 - 2.2.1.2. fully associative
 - 2.2.1.3. set associative
 - 2.2.2. Physical Address Fields
 - 2.2.3. Replacement Policies
 - 2.2.3.1. random
 - 2.2.3.2. LRU
 - 2.2.3.3. FIFO
 - 2.2.4. Write Policies
 - 2.2.4.1. write through/no write allocate
 - 2.2.4.2. write back/write allocate
 - 2.2.5. Unified Caches vs Separate Caches
 - 2.2.6. Categorizing Cache Misses
 - 2.2.6.1. compulsory misses
 - 2.2.6.2. capacity misses
 - 2.2.6.3. conflict misses
 - 2.2.7. Reducing Cache Miss Rate
 - 2.2.7.1. larger block size
 - 2.2.7.2. larger cache size

- 2.2.7.3. higher associativity
 - 2.2.8. Reducing Cache Miss Penalty
 - 2.2.8.1. multi-level caches
 - 2.2.8.1.1. calculating AMAT with multi-level caches
 - 2.2.8.1.2. local and global miss rates
 - 2.2.8.2. victim caches
 - 2.2.8.3. giving priority to read misses over write misses
 - 2.2.9. Reducing Cache Hit Time
 - 2.2.9.1. virtually addressed caches
 - 2.2.9.2. virtually indexed and physically tagged caches
 - 2.3. Overlays
 - 2.4. Virtual Memory
 - 2.4.1. virtual memory terms
 - 2.4.2. virtual memory process organization
 - 2.4.3. page tables
 - 2.4.4. translation lookaside buffers
 - 2.4.5. virtual to physical address translation process
 - 2.4.6. virtual memory support for multiprogramming
3. Chapter 2
- 3.1. processor and memory access time gap
 - 3.2. Memory Technologies
 - 3.2.1. SRAM
 - 3.2.2. DRAM
 - 3.2.2.1. DRAM optimizations
 - 3.2.2.2. high bandwidth memory (HBM)
 - 3.2.3. flash memory
 - 3.2.4. enhancing dependability
 - 3.2.5. protection via virtual memory
 - 3.3. Advanced Cache Optimizations
 - 3.3.1. Reducing Cache Hit Time
 - 3.3.1.1. small and simple first-level caches
 - 3.3.1.2. way prediction
 - 3.3.2. Increasing Cache Bandwidth
 - 3.3.2.1. pipelined cache access
 - 3.3.2.2. multibanked cache
 - 3.3.2.3. nonblocking data caches
 - 3.3.3. Reducing the Cache Miss Penalty
 - 3.3.3.1. critical word first and early restart
 - 3.3.3.2. merging write buffer
 - 3.3.4. Reducing the Cache Miss Rate through Compiler Optimizations
 - 3.3.4.1. code positioning
 - 3.3.4.2. array merging
 - 3.3.4.3. loop interchange
 - 3.3.4.4. blocking
 - 3.3.5. Reducing the Miss Penalty or Miss Rate via Parallelism
 - 3.3.5.1. hardware prefetching
 - 3.3.5.2. compiler controlled prefetching

- 3.3.5.3. using HBM as an L4 cache
- 3.4. Cross Cutting Issues
 - 3.4.1. I/O and the memory hierarchy
 - 3.4.2. stale data
 - 3.4.2.1. problems caused by stale data
 - 3.4.2.2. solutions to dealing with stale data
 - 3.4.3. avoiding the memory wall
- 4. Appendix A
 - 4.1. Classes of Instruction Sets
 - 4.1.1. Stack (zero address)
 - 4.1.2. Accumulator (one address)
 - 4.1.3. General-Purpose (two and three address)
 - 4.2. Addressing Issues
 - 4.2.1. Byte Addressable
 - 4.2.2. Size of a Word
 - 4.2.3. Byte Order (little endian or big endian)
 - 4.2.4. Alignment Requirements
 - 4.2.5. Extension of Bytes and Halfwords
 - 4.3. Special Addressing Modes and Operations
 - 4.4. Transfers of Control
 - 4.4.1. Types of Transfers of Control
 - 4.4.2. Evaluating Branch Conditions
 - 4.5. Compiler Interaction
 - 4.5.1. Calling Conventions
 - 4.5.2. Compiler Optimizations
 - 4.6. ISA Design
 - 4.6.1. Instruction Encoding Tradeoffs and Properties
 - 4.6.2. Computer Architecture Periods
 - 4.7. RISC-V Instructions