Computer Architecture Final Project

Evaluating Cache-Miss Reduction Techniques

** Joshua Green and Akhil Sankar

1. Project Overview

This project investigates mechanisms for reducing cache misses. While we are specifically interested in victim caches as outlined in Norman Jouppi's **"Improving Direct-Mapped Cache Performance by the Addition of a Small Fully-Associative Cache and Prefetch Buffers"**, other mechanisms can be incorporated as well.

Two simulation frameworks are used for testing:

- **SimpleScalar (PISA)** - used for designing and implementing cache mechanisms directly into the simulator -- **ChampSim** - used for evaluating cache behaviors using traces.

2. Motivation

Traditional L1 caches use fixed placement policies (direct-mapped or set-associative). Conflict misses can degrade performance significantly, especially in workloads with poor locality or with repeated access to a small conflicting set of addresses.

Victim Caches improve this by:

- Storing recently evicted L1 lines
- Allowing "secondary" lookup before going to L2 or memory
- Recovering conflict misses at low hardware cost

This project aims to

- Implement a victim cache in SimpleScalar
- Benchmark the effects on miss rates and performance
- Compare results against baseline caches
- Use ChampSim to validate for modern out-of-order cores

3. Project Directory Layout

`env.sh` # Environment loader `scripts/` # Automation for both simulators `config/` # User-editable experiment settings `results/` # Experiment outputs `traces/` # Benchmarks and ChampSim traces `simplesim-3.0/` # SimpleScalar simulator `ChampSim/` # ChampSim simulator `docs/` # Documentation directory