CS593: Principles in Computer Architecture

Spring, 2026

Instructor: Kazem Taram
E-mail: kazem@purdue.edu
Web: https://www.cs.purdue.edu/homes/mtaram/
Class Hours: TBD
Class Room: TBD

Course Description

This course will cover fundamental concepts in computer architecture. Topics include instruction set architecture, pipelining, pipeline hazards, bypassing, dynamic scheduling, branch prediction, superscalar issue, memory-hierarchy design, advanced cache architectures, prefetching and other techniques to explore instruction level parallelism and thread level parallelism. Throughout the course we will discuss challenges in designing high-performance processors (e.g., power, energy, performance, and security). In particular, we will delve deep into the recent processor security issues. We will also have case studies as to how modern microprocessors are designed, we will look at recent Intel, AMD, and Apple processors. We will also briefly cover design principles and examples of domain-specific architectures including accelerators for machine learning applications.

Learning Objectives

- Undrestand the challenges involved in designing high performance processors
- Learn basic techniques for improving performance in modern processors and their tradeoffs
- Learn how to evaluate architectural solutions (performance, security, and power)
- Acquire in-depth knowledge of microarchitectural structures that can be used in other domains, e.g., writing highly-optimized high-performance software

Prerequisites

Prerequisites: Undergraduate Computer Architecture (CS250 or equivalent).

Reading Materials

Course textbook

– Hennessy & Patterson, "Computer Architecture: A Quantitative Approach", Sixth Edition, Morgan Kaufmann (requited)

Other background reading:

– We'll also be reading some research papers as well as referencing some books from the outstanding series Synthesis Lectures on Computer Architecture, which can be found here: Synthesis Lectures on Computer Architecture. Those books will be free to access while you are on the Purdue domain.

– Hennessy & Patterson, "Computer Organization and Design: The Hardware/Software Interface" gives a lower-level treatment of the material (more from the design standpoint). This represents the assumed background/prerequisite for this class.

High-Level Course Outline

- Introduction
- Computer System Performance
- Instruction Set Architecture
- Pipelining
- Instruction-Level Parallelism
- Data Level Parallelism
- Thread level parallelism
- Domain Specific Accelerators
- Security

Assessments (Tentative)

- Homework Assignments: 40%
- Midterm-1: 25%
- Final: 25%
- Paper Presentations: 5%
- Online Quizzes/Paper Summaries (for readings): 5%

Course Format

- **Lectures:** 60-80% of the class will be lectures by the instructor. Lectures will be in-person. Students are expected to attend most of the lectures.
- **Student Presentations:** Students (individually or in groups, depending on the size of the class) are expected to present and lead the discussion of papers from the research literature to class. The papers will be selected by the instructor. A good presentation will include the necessary background and motivation, will explain overall merit and technical details, and provide answers to questions from the audience. While presenting a paper, a critical perspective will be encouraged and students do not need to necessarily defend the paper. All students are expected to read the paper before each class and participate in the class discussion.
- **Homework Assignment:** We will have four programming assignments/labs to help you fully understand course materials. The following is a tentative coverage of these assignments:
  - Pipelining and out of order execution
  - Branch predictor (competition)
  - Caching and advanced memory optimization techniques
  - Side-channel security lab
- **Reading Quizzes:** There will be either online quizzes or paper summaries due the start of each class to further encourage students to read the papers.