

# Project 2

## Project description

You will need to implement a trace-driven cache simulator, and use it to evaluate the performance of different cache architecture features. The project is described in detail in sim.pdf.

## File description

In the attachment, you will find the following project files:

- cache.h - cache simulator definitions
- cache.c - cache simulator (put all your edits here)
- main.h - simulation driver definitions
- main.c - simulation driver
- Public tests
  - public-block.trace - test cache block sizes
  - public-assoc.trace - test cache associativity
  - public-write.trace - test cache write policy
  - public-instr.trace - test instruction cache
  - spice10.trace - 1st 10 accesses in spice.trace
  - spice100.trace - 1st 100 accesses in spice.trace
  - spice1000.trace - 1st 1000 accesses in spice.trace
- Expected outputs for public tests
  - public-block1.out
  - public-assoc1.out
  - public-write1.out
  - public-instr1.out
  - spice10.out
  - spice100.out
  - spice1000.out
- Makefile - makefile to create simulator
- runPublic - csh script to run public tests
- sim.pdf - detailed description of project
- tags.txt - example index & tag values for spice100.trace

In addition, there are three large application traces:

- spice.trace - circuit simulator
- cc.trace - C compiler
- tex.trace - Tex document processor

## Requirements

- Write a simple report about how you implement the cache simulator and answer the questions in sim.pdf
- Compress your project and submit it to [XueZaiZheDa](#) before the deadline. And hand in the paper report in class on 2025.1.2. Use your student ID as filename. Files you should submit:
  - code/ directory
  - Report

## Note

- It's ok if you can't implement a fully functional simulator, make your best effort and do what you can do.