

6.5900 (6.823)

Computer System Architecture

AI Use Policy

<http://csg.csail.mit.edu/6.5900/>

Using AI assistance (e.g., Claude Code, Codex, etc.) is allowed within specific guidelines when completing most of the labs. Specifically, AI assistance will be allowed for labs 1, 2, and 4. AI assistance will be completely **prohibited** for lab 3. This includes chat assistants (claude/codex web-gui), coding agents (claude/codex CLI), and AI autocomplete in your editor (Copilot).

The goal of this AI use policy is to enable productive use of these tools, reflective of modern practice, while ensuring that the intellectual work behind the lab remains your own. For example, you will be able to use AI to code specific solutions, run experiments, etc. However, using AI to solve the entire lab with no or little input from you is prohibited. Similarly, you cannot use AI tools to write the lab report for you. Going beyond these restrictions will be treated as a violation of the academic honesty policy.

To ensure that you understand and own your lab solutions, quizzes may include personalized questions about your lab designs. These can draw upon your code and report, so overusing AI in labs will adversely impact not only your lab scores, but your quiz performance as well.

In many ways, our policy is close to the policy for using human assistance:

- Just like you can come to office hours and ask a TA about the lecture material, you are welcome to ask an AI the same kinds of questions. You may also look up definitions and facts, like you'd use a textbook or a search engine. Questions about what to build for the lab are a different matter, since that's the part we'd like you to work out yourself. If you're unsure what a lab question is asking, please ask on Piazza or come to office hours.
- Just like you are not allowed to ask an expert friend to do your homework for you, you should also not ask an expert AI.
- If you use any AI on a lab, you need to explain to us how you used it (see below how to do that).

Allowed Uses of AI Assistants (What is OK)

1. **Having the AI write code from a detailed plan that you devised.** You decide what the code does, how it works, what its inputs and outputs are, and how it handles special cases, and you state all of this in your prompt. The AI only turns your plan into code. The plan must be specific enough that the AI has no design decisions left to make. If the AI makes any design decision, your plan is not detailed enough, and that use **IS NOT** allowed in the context of this class. Compare these two prompts:

- **Disallowed:** “Write a script that plots my cache results”
- **Allowed:** “Write a python script, plot.py, that reads results.csv (columns: benchmark, logBlockSize, logNumRows, associativity, reads, writes, readHits, writeHits) and produces one figure per benchmark. Put total cache size on the x-axis on a log2 scale, miss rate on the y-axis, and one line per associativity, where miss rate is $1 - (\text{readHits} + \text{writeHits}) /$

(reads + writes). Save each figure as <benchmark>.png in a figs/ directory, and also write a combined figure averaging over all benchmarks. Label the axes, add a legend for the associativity values, and use the same y-axis range across all benchmarks so I can compare them side by side. If a (benchmark, configuration) pair is missing from the CSV, leave a gap in the line rather than interpolating, and print the missing pairs at the end so I know what still needs to run. Don't add any analysis or commentary to the plots; I'll interpret them in the report myself."

The first prompt leaves every decision to the AI. The second is the kind of prompt we'd like to see in your AI statement file (more on this later): You've decided what the tool does, and the AI saves you typing. While in this case, the script is not part of the design, we expect you to do the same for the actual design coding assignments.

2. Asking conceptual questions about the course material. You may ask an AI to explain concepts from lecture and to look up definitions and facts, the same way you would ask a TA at office hours or consult a textbook. This covers understanding the material. It does not cover what to build for the lab or what a lab question is asking. Questions of the following kind are allowed:

- "How does tree-PLRU work?"
- "What's the difference between a write-back and a write-through cache?"

3. Debugging a specific problem that you are stuck on. You may give an AI an error message or a description of incorrect behavior, together with the relevant code, and ask it to explain the cause and propose a fix. You must name the specific problem. Asking the AI to make your design work better or perform better is not debugging. You are expected to understand any fix you apply. Prompts of the following kind are allowed:

- "Each file compiles fine on its own but linking fails with multiple definition of 'hash_addr'. I only wrote that function once, in utils.h. Could you provide me with a proposed solution and explain what the problem is?"
- "The compiler says expected ';' in example_file.cpp, but that line looks correct and I haven't touched that file. What could cause this? Could you provide me with a proposed solution and explain what the problem is."

In general, tasks that require a lot of time but not much thought are a great way to use these tools.

Disallowed Uses of AI Assistants (What is NOT OK)

The following are *prohibited*:

- **DO NOT** share the lab handout with an AI in any form. That includes uploading the PDF, pasting the questions, or running an agent in a directory that contains them.
- **DO NOT** ask an AI assistant to write the report for you. In many cases, questions that you will answer in the report will be more important and more reflective of your understanding of the concepts intended to be exercised by the lab than the code.
- **DO NOT** use agents to run and monitor more simulations than the ones you would normally run if you did not have agents. In many of the labs, a significant portion of achieving the correct solution will be to run simulations to verify your current solution and to see the impact of your final solution on the benchmarks. This is very

important because we have a limited number of machines dedicated to the class and an overwhelming number of experiments would saturate compute resources, causing trouble for all students trying to run simulations.

DO NOT use prompts that ask the AI to make design decisions, interpret results, or complete the lab for you, such as:

- “Here are my hit rates for each configuration. What trends do you see, and why do they happen?”
- “Implement a branch predictor that achieves 99.9% accuracy on this data. Use this starter code. Make no mistakes”
- “I have done this so far but doesn’t work very well, fix it. Rerun experiments if necessary and write the report explaining the code and results.”
- “I am attaching the pdf with the questions and the starter code for this lab that I have to do for my computer architecture class. Implement any necessary code changes, run all experiments that the pdf asks us to do and finally write a report explaining the code implementation, reporting the results of the experiments, and answering any questions the report has. Make no mistakes.”

If in doubt, ask the staff

If you have any uncertainty or doubt about how to use an AI, please ask us on Piazza or come to Office Hours.

What to include in your lab if you used AI

Using AI is optional, and you are welcome to complete every lab without it.

*Every submission must include **ai-statement.md**, and its first line must state whether you used AI on the lab. If you did not use AI, that line is the whole file.*

If you used AI, **ai-statement.md** must also state:

- the tools you used, and the model you used most with each tool;
- what you used each one for;
- for every piece of AI-written code longer than a few lines, the full prompt containing your plan, followed by any clarification prompts you sent afterwards or a description of them;
- the files and functions that contain AI-written code.

If the AI wrote more than a few lines of code for you, your lab directory must also contain **ai-logs/**, a directory with the complete log of every such conversation and agent session, as .txt or in the format the tool exports. If you do not know how to export logs from your tool, ask on Piazza or at office hours.

It would be useful if you keep track of this information and add to it as you go and commit different stages of **ai-statement.md** as part of your commit history. You may also reference specific chat logs committed as part of the **ai-statement.md**. Beyond showing your work, doing

this will also enable the course staff to better understand how you are using these tools, so that we can improve future iteration of these labs.