CS 4100, Fall 2025
Topic Proposal Guidelines

Your initial topic proposal must be uploaded to Canvas no later than **Monday, September 8th**. Please see the Project Handout for details about the project paper. Below are format guidelines for the topic proposal:

<Your Name>
CS 4100
<Date>
Topic Proposal

<Working Title>

First paragraph: describe your proposed topic.

Second paragraph: discuss which issues you intend to address and emphasize and what you intend to leave out and why.

Additional paragraphs as necessary.

Provide a list of tentative sources in APA format. You must have at least four sources and they must be reliable (peer reviewed, edited, or primary). Sources **not** allowed include (but are not limited to) blogs, Wikipedia, stackoverflow, and similar sites. Acceptable references are required.

https://library.csustan.edu/computerscience/citation

To continue in this course, you must have your topic proposal approved; I may ask you for one or more revisions along the way. Students may have similar, but not identical topics, I reserve the right to make the determination between acceptable similar and not acceptable topics. Submit your topic proposal no later than **Monday, September 8th**.

Choosing a Topic

Your choice of topic will be of considerable importance to the success of your project, so choose carefully and thoughtfully. Make it something that engages your interest and curiosity. It must also be something about which enough has been written that you can access multiple reliable sources in order to synthesize and write a deep, accurate, and balanced report. However, you must also be careful not to choose too broad a topic, which spans too much intellectual territory.

Your topic needs to deal with programming languages and computers and needs to go significantly beyond material we address in class. Although you may eventually impose your own emphases, and organization, it should not be your own opinions, but rather a fact-based reporting of some issue. Most good topics will include some complexity in terms of alternative approaches which might lead to different choices under different circumstances, or which have been handled in different ways in the past. Your paper must also have technical depth, which could include the thoughtful analysis of code. Please refer to the Scoring Rubric for details about depth, accuracy, and structure.

To summarize, your paper should be a well-researched, balanced, factual report.

Some reasonable approaches to topic selection:

1. Start with a problem and look at which languages (often special purpose) are used to solve the problem. Consider the design choices made to facilitate solving this type of problem.
Some examples might include graphics, teaching, robotics, or mobile devices.

2. Look at how an interesting aspect of programming languages is treated differently in different languages. Some examples could include, data sharing, parameter passing, typing, or scope. You would need to compare and contrast the approaches in at least two, but not more than four programming langauages.

3. Look in depth at a programming paradigm, for example logic or rule-based languages, describe the design choices and the types of problems for which it is well suited. In this case you would be comparing your choses paradigm to other paradigms.

Some topics that are **not** allowed:
1. Something that is too broad. For example, programming languages for AI. Instead narrow to a specific subfield of AI, such as planning, knowledge bases, robotics, natural language processing, etc.

2. An overview, or aspect, of a single language. To get depth you will need to compare and contrast, so you will need to discuss more than one language.