Goals

Select the programming language you will use to build your compiler. Choose carefully, as you are making a semester-long commitment. You may choose from C, C++, Java, and TypeScript.

Every now and then a student will ask to use a functional language like Haskell, ML, or LISP. That’s probably okay, but ask me first.

If there’s a language that you’d like to use that I haven’t already mentioned, ask me about it. If you’re thinking about Python, let me save you some time: NO.

Development Tooling and Build Documentation

• Work out your development tooling.
• Make sure that Alan can compile and build your code.

Notes

• Set up your development environment.
• Practice your code, compile/build, deploy, debug cycle.
• Document and explain how to build and deploy your code in ridiculous detail in the readme.md document in the root of your GitHub repository.
• See the latest build environment notes on our class web site.
• Test it. A lot. If Alan cannot compile your code, you will fail.

Resources

Crafting a Compiler

• Read chapters 1 and 10.1.2

Dragon

• Read chapter 1

Other

• Overleaf: https://www.overleaf.com/
• LaTeX lab template: https://www.overleaf.com/read/njbdynktsc

Submitting

Commit your build and deploy documentation in readme.md in your private GitHub repository. Add me (Labouseur) as a collaborator and e-mail me the URL.