

# Final Project Checklist

CS334 Fall 2022

- (a) \_\_\_\_\_ You have created a 5-10 minute video presentation, and included this video with your implementation. Note: large video files cannot be added to GitHub; please include a link to Google Drive/YouTube/etc. instead.
- (b) \_\_\_\_\_ Your project has a (silly) name.
- (c) \_\_\_\_\_ Your project has a specification.
- (d) \_\_\_\_\_ Your parser is in a file called `Parser.fs`.
- (e) \_\_\_\_\_ Your AST is in a file called `AST.fs`.
- (f) \_\_\_\_\_ Your interpreter / evaluator is in a file called `Evaluator.fs`.
- (g) \_\_\_\_\_ Your `main` function is in a file called `Program.fs`.
- (h) \_\_\_\_\_ Your project compiles (this is **very** important).
- (i) \_\_\_\_\_ Your project runs (this is **very** important).
- (j) \_\_\_\_\_ Your language “does something.” It prints out a computed result, it generates a file, etc.
- (k) \_\_\_\_\_ **You are sure to tell the user** (me) what the expected result should be!
- (l) \_\_\_\_\_ Your implementation has a test suite and it runs.
- (m) \_\_\_\_\_ There is at least one test written for the parser.
- (n) \_\_\_\_\_ There is at least one test written for the interpreter.
- (o) \_\_\_\_\_ Your project can be run at the project level by calling `dotnet run <input>` or at the solution level by calling `dotnet run --project <whatever.fsproj> <input>`.
- (p) \_\_\_\_\_ Your specification document has a title.
- (q) \_\_\_\_\_ Your name and your partner’s name is written at the top of the spec.
- (r) \_\_\_\_\_ Your specification has an **Introduction** section consisting of **2+** paragraphs.
- (s) \_\_\_\_\_ Your specification has a **Design Principles** section consisting of **1+** paragraphs.
- (t) \_\_\_\_\_ Your specification has an **Examples** section consisting of **3+** short examples.
- (u) \_\_\_\_\_ Each of your examples is provided (and ideally, each example is in a separate file) so that an interested third party (me or one of your classmates) can run them. Instructions to run the examples is provided.
- (v) \_\_\_\_\_ Your specification has a **Language Concepts** section consisting of **1+** paragraphs.

- (w) \_\_\_\_\_ Your specification has a **Formal Syntax** section consisting of **as much BNF is needed** to completely describe your language's syntax.
- (x) \_\_\_\_\_ Your specification has a **Semantics** section consisting of **one short description per language element** (where an element is normally an AST node), completely describing your language.
- (y) \_\_\_\_\_ Your specification provides enough detail that an interested third-party (like me or one of your classmates) can write a new program in your language.
- (z) \_\_\_\_\_ If syntactically valid but meaningless programs are possible, make sure that the program detects these cases and shuts down cleanly (i.e., the user does not see an exception).
- ( $\alpha$ ) \_\_\_\_\_ You committed your specification, in a folder called `docs`, to a branch called `final-submission`.
- ( $\beta$ ) \_\_\_\_\_ You committed your implementation, in a folder called `code`, to a branch called `final-submission`.
- ( $\gamma$ ) \_\_\_\_\_ Your specification has a **Remaining Work** section that describes further enhancements, if any, you would like to see. If your prior draft had features that you did not get to by the final submission, briefly explain why you were not able to implement them.
- ( $\delta$ ) \_\_\_\_\_ If you would like me to transfer ownership of your repository to you, please provide the name of a Github user or organization that will take over ownership. Put this username in a file called `TRANSFER.txt`.
- ( $\epsilon$ ) \_\_\_\_\_ If you intend to make your repository public, put a `LICENSE.txt` copyright statement in your repository, at the root level, so that people know under what conditions you plan to let them use your code. For example, provide a copy of the GNU Public License, or BSD License, etc.