# CSE341 HW3 REPORT

Flex + Bison:

Lexer: gpp\_lexer.l

Sends tokens while also binding them, important part is VALUEI VALUEF and IDENTIFIER token bindings, which is done using helper functions in gpp\_helper

Bison: gpp\_interpreter.y

Defined some token values sent by lexer, defined concrete grammar, applies shift reduce parser to match given input while also doing semantic bindings.

Headers:

gpp\_helper.h:

Gives definitions for lexer and parser to use:

Structs for linked lists, fractions, reference environment tables

Functions for said linked lists, fractions.

gpp\_helper.c:

It has implementations for arithmetic fraction operations, fraction creation, table operations(linked lists).

Lisp:

Main Evaluator:

The gpp-eval function evaluates expressions defined in gpp language.

It tests, checks variable tables, recursively handles expressions and do uses necessary functions

Repl:

gpp-repl continuously reads input from the user, sends it to evaluator, and prints the result. It handles errors gracefully and allows the user to exit the interpreter by typing exit.

Helpers:

split-sequence:

Split a sequence with a delimiter. Used for parsing the input.

Fraction helpers:

Arithmetic and logical: add-fractions, subtract-fractions, multiply-fractions, divide-fractions, and compare-fractions,

Functional: parse-fraction, fraction-to-string