In this session, instead of updating the functionality of our code that translates C (infix notation) into Lisp (prefix notation), we focused on a brand new code that translates Lisp into Forth (postfix notation). We will refer to this part of the final assignment as the backend, as opposed to the frontend, which was what we were doing in the previous sessions.

What we needed to do was transform the output of the code of the previous sessions into the input of the new code, which would be transformed with something compliant with the syntax of Forth. Since it was a new code file we first needed to create the tokens for all the keywords that we would be using (setq, setf, print, princ, do, loop, if, etc.). After that, we just needed to do exactly what we did earlier, but changing the grammar so that it fits the Lisp syntax. Once that was done we only needed to adjust the output to adapt to the Forth language. The general structure of our backend code ended up being almost a mirror image of our frontend code, it even uses the exact same lexical analyzer (yylex).