Grammar

Implemented: (sort of)

$$
\begin{aligned}
[\mathit{prog}] &\to \quad [\mathit{stmt}]^* \\
[\mathit{scope}] &\to \quad \{[\mathit{stmt}]^*\} \\
[\mathit{stmt}] &\to \begin{cases}
[\mathit{type}]\ \langle \mathit{ident}\rangle = [\mathit{expr}]; & \textbf{variable assignment} \\
\langle \mathit{ident}\rangle = [\mathit{expr}]; & \textbf{variable reassignment} \\
\mathit{ret}\ [\mathit{expr}]; & \textbf{exit/return} \\
\mathit{if}[\mathit{expr}][\mathit{scope}] \\
\mathit{while}[\mathit{expr}][\mathit{scope}] \\
[\mathit{scope}]
\end{cases} \\
[\mathit{expr}] &\to \begin{cases}
[\mathit{term}] \\
[\mathit{binexpr}]
\end{cases} \\
[\mathit{binexpr}] &\to \begin{cases}
[\mathit{expr}] + [\mathit{expr}] & \mathit{prec} = 0 \\
[\mathit{expr}] - [\mathit{expr}] & \mathit{prec} = 0 \\
[\mathit{expr}] * [\mathit{expr}] & \mathit{prec} = 1 \\
[\mathit{expr}]\ /\ [\mathit{expr}] & \mathit{prec} = 1
\end{cases} \\
[\mathit{term}] &\to \begin{cases}
\langle \mathit{int\_lit}\rangle \\
\langle \mathit{ident}\rangle \\
\_[\mathit{term}] & \textbf{negative expression (underscore)} \\
([\mathit{expr}])
\end{cases} \\
[\mathit{type}] &\to \begin{cases}
\mathit{int}
\end{cases}
\end{aligned}
$$

Todo:

$$
\begin{aligned}
[prog] &\to [stmt]^* \\
[stmt] &\to \begin{cases} ret\ [expr]; \\ [type]\ ident = [expr]; \\ if([expr])[stmt] \\ [scope] \end{cases} \\
[scope] &\to \{[stmt]^*\} \\
[expr] &\to \begin{cases} [binexpr] \\ [term] \end{cases} \\
[binexpr] &\to \begin{cases} [expr] * [expr] & prec = 1 \\ [expr]\ /\ [expr] & prec = 1 \\ [expr] + [expr] & prec = 0 \\ [expr] - [expr] & prec = 0 \end{cases} \\
[term] &\to \begin{cases} int\_lit \\ ident \\ ([Expr]) \\ TRUE \\ FALSE \\ 'c' \end{cases} \\
[type] &\to \begin{cases} int \\ bool \\ char \\ float \end{cases}
\end{aligned}
$$