|  |  |
| --- | --- |
| Program | (FuncDeclaration | GlobalDeclaration)+ |
| GlobalDeclaration | ‘var’ Label ‘=’ Expression ‘;’ |
| FuncDeclaration | ‘function’ Label ‘(‘ [‘var’ Label [‘, var’ Label]\* ] ‘)’ CodeBlock |
| CodeBlock | ‘{‘ [Code]\* ‘}’ |
| Code | Statement | Control |
| Control | IF | LOOP |
| LOOP | ‘while’ ‘(‘ Condition ‘)’ CodeBlock |
| IF | ‘if’ ‘(‘ Condition ‘)’ CodeBlock [ELSEIF]\* [ELSE] |
| ELSEIF | ‘else if’ ‘(‘ Condition ‘)’ CodeBlock |
| ELSE | ‘else’ CodeBlock |
| Condition | Expression ConditionOp Expression |
| ConditionOp | == | != | < | > | <= | >= |
| Statement | (VarDeclaration | Assignment | FuncInvoke) ‘;’ |
| FuncInvoke | Label ‘(‘ [Expression [, Expression]\* ‘)’ |
| VarDeclaration | ‘var’ Label ‘=’ Expression |
| Assignment | Label ‘=’ Expression |
| Expression | Or |
| Or | Xor [ ‘|’ Or] |
| Xor | And [ ‘^’ Xor] |
| And | Shift [ ‘&’ And] |
| Shift | MathExpression [ (‘<<’ | ‘>>’ | ‘>>>’) Shift] |
| MathExpression | [‘-‘] Term [ (‘+’|’-‘) Term ]\* |
| Term | Factor [‘\*’ Term] |
| Factor | [‘~’] (‘(‘ Expression ‘)’ | Constant | Variable | FuncInvoke) |
| Variable | Label |
| Label | [A-Za-z][A-Za-z0-9]\* |
| Constant | ‘0x’ [A-F0-9]+ | [0-9]+ |

TODO: figure out a better way to handle leading negative sign on MathExpression