

Auto speed “parser tree”

Start

<Program> => <ImportSection>| <NamespaceSection> |<FunctionList>

Import & Namespace

<ImportSection> => <ImportDecl> |<ImportSection> | ϵ

<ImportDecl> => "#oil" "<" <Identifier> ">"

<NamespaceSection> => "key" <Identifier>

Function Definitions

<FunctionList> => <Function>| <FunctionList> | ϵ

<Function> => "engine" <Identifier> "(" <ParamList> ")" <Block>
| "ignite" "(" <ParamList> ")" <Block>

<ParamList> => <Param> { ";" <Param> } | ϵ

<Param> => <Type>| <Identifier>

Blocks & Statements

<Block> => "{" <StatementList> "}"

<StatementList> => <Statement>| <StatementList> | ϵ

<Statement> => <DeclarationStmt>

| <AssignmentStmt>

```
| <ConditionalStmt>  
| <LoopStmt>  
| <OutputStmt>  
| <InputStmt>  
| <FunctionCallStmt>  
| <ReturnStmt>
```

Variable Declaration & Assignment

```
<DeclarationStmt> => <Type>| <Identifier> "=" <Expression> ";"  
<AssignmentStmt> => <Identifier> |"="| <Expression> |";"  
<ConditionalStmt> = >"track" "(" <Expression> ")" <Block> [ "pitstop" <Block> ]  
<LoopStmt> ::= "looplac" "(" <Expression> ")" <Block>
```

I/O & Function Calls

```
<OutputStmt> => "announce" <Expression> ";"  
<InputStmt> => "listen" <Identifier> ";"  
<FunctionCallStmt> => <Identifier> "(" <ArgList> ")" ";"  
<ReturnStmt> => "finishline" <Expression> ";"
```

Arguments

```
<ArgList> => <Expression> { "," <Expression> } | ε
```

Expressions

```
<Expression> => <Term> { ("+" | "-" | "==" | "!=" | "<" | "<=" | ">" | ">=") <Term> }  
<Term> => <Factor> { ("*" | "/" | "%") <Factor> }  
<Factor> => <Number>  
| <String>
```

```
| <Identifier>
| "(" <Expression> ")"
| <FunctionCallExpr>
| <BooleanLiteral>

<FunctionCallExpr> => <Identifier> "(" <ArgList> ")"

<BooleanLiteral> => "true" | "false"
```

Data Types

```
<Type> => "gear" | "track" | "exhaust" | "turbo" | "flag"
```

Tokens / Lexical Units

```
<Identifier> => letter { letter | digit | "_" }
```

```
<Number> => digit { digit } [ "." digit { digit } ]
```

```
<String> => "" { character } ""
```