

Grammar v0.4

November 5, 2019

```
INT := 'int'
FLOAT := 'float'
ID := alpha + (alphanum)*
VOID := 'void'
( := '('
) := ')'
{ := '{'
} := '}'
[ := '['
] := ']'
, := ','
+ := '+'
++ := '++'
- := '-'
'/': := '/'
* := '*'
> := '>'
< := '<'
; := ';'
= := '='
NUM := (num)+
      | (num)* + '.' + (num)+
      | (num)+ + '.'
FOR := 'for'
IF := 'if'
RETURN := 'return'
STRING := '"' + (stuff)* + '"'
```

```

< goal > := < funcs >
< funcs > := ε
           | < func > < funcs >
< func > := INT/FLOAT ID ( < args > { < stmts > }
< args > := VOID )
           | ε )
           | INT/FLOAT ε / * ID :, INT/FLOAT ε / * ID : * )
< stmts > := ε
           | < stmt > < stmts >
< stmt > := < instr > ;
           | < forst >
           | < ifst >
           | { < stmts > }
           | < retst >
< instr > := < decl >
           | < expr >
           | ε
< decl > := INT/FLOAT ID :, ID : *
           | INT/FLOAT ID [ NUM ] :, ID [ NUM ] : *
< expr > := < expr > + / - < term >
           | < term >
           | ID / ID [ < expr > ] = < expr >
< term > := < factor >
           | < term > * / ' ' < factor >
< factor > := ID
           | ID / ID [ < expr > ] ++
           | ++ ID / ID [ < expr > ]
           | ID [ < expr > ]
           | ID ( < call > )
           | NUM
           | ( < expr > )
           | + / - factor
< call > := ε
           | < expr > :, < expr > : *
< forst > := FOR ( < instr > ; < comp > ; < expr > ) < stmt >
< ifst > := IF ( < comp > ) < stmt >
< comp > := < expr > > / < < expr >
< retst > := RETURN < expr > ;

```

1. $\langle stmt \rangle := \langle instr \rangle ;$
2. | $\langle forst \rangle$
3. | $\langle ifst \rangle$
4. | $\{ \langle stmts \rangle \}$
5. | $\langle retst \rangle$

1. $\langle instr \rangle := INT/FLOAT ID \ ;, ID :^*$
2. | $INT/FLOAT ID[NUM] \ ;, ID[NUM] :^*$
3. | $\langle expr \rangle$
4. | ϵ

1. $\langle expr \rangle := \langle term \rangle \mid \langle expr \rangle + / - \langle term \rangle$
2. | $lhs = \langle expr \rangle$

1. $\langle factor \rangle := lhs \mid ++lhs \mid lhs ++$
2. | $ID(\langle call \rangle)$
3. | NUM
4. | $NUM2$
5. | $(\langle expr \rangle)$
6. | $+ / - \langle factor \rangle$