

```

⟨program⟩ → [⟨metas⟩] ⟨object⟩ EOF
⟨metas⟩ → ⟨meta⟩ { EOL ⟨meta⟩ } EOP
⟨comment⟩ → ⟨commentary⟩ EOL
⟨commentoptional⟩ → { ⟨comment⟩ }
⟨object⟩ → (⟨commentoptional⟩ ⟨masterbody⟩)
| ⟨bound⟩
⟨bound⟩ → ⟨commentoptional⟩ (⟨application⟩ | ((⟨method⟩ | ⟨just⟩) (⟨onameortname⟩ | ⟨aphi⟩) EOL))
⟨tbound⟩ → ⟨commentoptional⟩ (⟨tapplication⟩ | (((⟨method⟩ | ⟨just⟩) ⟨tname⟩ EOL))
⟨tsubmaster⟩ → ⟨commentoptional⟩ ⟨tmasterbody⟩
⟨submaster⟩ → ⟨commentoptional⟩ ⟨masterbody⟩
⟨masterbody⟩ → ⟨formation⟩ | ⟨atom⟩ | ((⟨hanonym⟩ ⟨onameortname⟩ EOL)
⟨tmasterbody⟩ → ⟨tformation⟩ | ⟨atom⟩ | ((⟨hanonym⟩ ⟨tname⟩ EOL)
⟨just⟩ → ⟨beginner⟩ | ⟨finisher⟩
⟨atom⟩ → ⟨voids⟩ ⟨suffix⟩ "␣" "?" ⟨testsoreol⟩
⟨formation⟩ → ⟨voids⟩ ⟨onameortname⟩ ⟨innersoreol⟩
⟨tformation⟩ → ⟨voids⟩ ⟨tname⟩ ⟨innersoreol⟩
⟨testsoreol⟩ → ⟨tests⟩ | EOL
⟨innersoreol⟩ → ⟨inners⟩ | EOL
⟨inners⟩ → EOL TAB (⟨bound⟩ | ⟨submaster⟩) { (⟨bound⟩ | ( EOL ⟨submaster⟩ ) ) } UNTAB
⟨tests⟩ → EOL TAB (⟨tbound⟩ | ⟨tsubmaster⟩) { (⟨tbound⟩ | ( EOL ⟨tsubmaster⟩ ) ) } UNTAB
⟨voids⟩ → "[" [⟨void⟩ { "␣" ⟨void⟩ } ] "]"
⟨void⟩ → ⟨name⟩ | "@"
⟨application⟩ → ((⟨happlicationextended⟩ ⟨onameortname⟩ EOL)
| ⟨vapplication⟩)
⟨tapplication⟩ → ((⟨happlicationextended⟩ ⟨tname⟩ EOL)
| ⟨tvapplication⟩)
⟨happlication⟩ → ((⟨happlicationhead⟩ ⟨happlicationtail⟩)
| ⟨happlicationreversed⟩)
⟨happlicationreversedhead⟩ → ⟨reversed⟩ "␣" ⟨happlicationreversedfirst⟩
⟨happlicationextended⟩ → ((⟨happlicationheadextended⟩ ⟨happlicationtail⟩)
| ⟨happlicationreversed⟩)
⟨onlyaphi⟩ → ⟨happlicationheadextended⟩ ⟨happlicationtail⟩ ⟨aphi⟩
⟨aphi⟩ → "␣" ">" ">" "␣" ⟨voids⟩
⟨happlicationreversed⟩ → ⟨happlicationreversedhead⟩ [⟨happlicationtail⟩]
⟨happlicationhead⟩ → ⟨hmethod⟩ | ⟨applicable⟩
⟨happlicationheadextended⟩ → ⟨vmethod⟩ | ⟨happlicationhead⟩
⟨applicable⟩ → "*" | ⟨name⟩ | "@"
⟨happlicationargunbound⟩ → "␣" ⟨happlicationarg⟩
⟨happlicationtailscoped⟩ → { ⟨happlicationargunbound⟩ } "␣" ⟨happlicationargscoped⟩
⟨happlicationtail⟩ → { ⟨happlicationargunbound⟩ ⟨as⟩ }+ | { ⟨happlicationargunbound⟩ }+
⟨happlicationreversedfirst⟩ → ⟨happlicationarg⟩
⟨happlicationarg⟩ → ⟨just⟩ | ⟨hmethod⟩ | ⟨scope⟩
⟨happlicationargscoped⟩ → ⟨voids⟩ ⟨aname⟩ ⟨innersoreol⟩
⟨vapplication⟩ → ((⟨vapplicationhead⟩ ⟨onameortname⟩ ⟨vapplicationargs⟩) | ((⟨reversed⟩ ⟨onameortname⟩ ⟨vapplicationargsreversed⟩)
⟨tvapplication⟩ → ((⟨vapplicationhead⟩ ⟨tname⟩ ⟨vapplicationargs⟩) | ((⟨reversed⟩ ⟨tname⟩ ⟨vapplicationargsreversed⟩)
⟨vapplicationhead⟩ → ⟨applicable⟩ | ⟨hmethod⟩ | ⟨vmethod⟩ | ⟨compactarray⟩
⟨compactarray⟩ → "$" | "QQ" | ⟨name⟩ { "." ⟨name⟩ } "␣" "*" [⟨int⟩]
⟨vapplicationargs⟩ → EOL TAB ⟨vapplicationargsspecific⟩ UNTAB
⟨vapplicationargsreversed⟩ → EOL TAB ⟨vapplicationargunbound⟩ [⟨vapplicationargsspecific⟩] UNTAB
⟨vapplicationargsspecific⟩ → { ⟨vapplicationargbound⟩ }+ | { ⟨vapplicationargunbound⟩ }+
⟨vapplicationargbound⟩ → ((⟨vapplicationargboundcurrent⟩ EOL)
| ⟨vapplicationargboundnext⟩)
⟨vapplicationargboundcurrent⟩ → (( "(" ⟨happlicationextended⟩ ")" ) | ⟨just⟩ | ⟨method⟩ ⟨as⟩ [⟨oname⟩] ) | ((⟨commentoptional⟩ "(" ⟨hanonym⟩ ")" ⟨as⟩ [⟨fname⟩])
⟨vapplicationargboundnext⟩ → ((⟨commentoptional⟩ ⟨voids⟩ ⟨as⟩ [⟨fname⟩] ⟨innersoreol⟩) | ((⟨vapplicationhead⟩ ⟨as⟩ [⟨oname⟩] ⟨vapplicationargs⟩) | ((⟨reversed⟩ ⟨as⟩ [⟨oname⟩] ⟨vapplicationargsreversed⟩)
⟨vapplicationargunbound⟩ → ((⟨onlyaphi⟩ | ((⟨vapplicationargunboundcurrent⟩ [⟨aphi⟩]) EOL)
| ⟨vapplicationargunboundnext⟩)
⟨vapplicationargunboundcurrent⟩ → ((⟨happlicationextended⟩ | ⟨just⟩ | ⟨method⟩ [⟨oname⟩]) | ((⟨commentoptional⟩ ⟨hanonym⟩ [⟨fname⟩])
| ⟨vapplicationargunboundnext⟩)
⟨vapplicationargunboundnext⟩ → ⟨formationnamed⟩ | ((⟨vapplicationhead⟩ [⟨oname⟩ | ⟨aphi⟩] ⟨vapplicationargs⟩) | ((⟨reversed⟩ [⟨oname⟩] ⟨vapplicationargsreversed⟩) | ((⟨happlicationhead⟩ | ⟨happlicationreversedhead⟩ ⟨happlicationtailscoped⟩)
⟨formationnamed⟩ → ⟨commentoptional⟩ ⟨voids⟩ [⟨fname⟩] ⟨innersoreol⟩
⟨hformation⟩ → ⟨voids⟩ { ⟨hanonyminner⟩ }+
⟨hanonym⟩ → ⟨hformation⟩ | ((⟨hmethod⟩ | ⟨happlication⟩ | ⟨hformation⟩ | ⟨just⟩ { ⟨onlyphitail⟩ } )+
⟨onlyphitail⟩ → ⟨arrow⟩ ⟨voids⟩
⟨hanonyminner⟩ → "␣" "(" (⟨hmethod⟩ | ⟨happlication⟩ | ⟨hanonym⟩ | ⟨just⟩) ⟨oname⟩ ")"
⟨method⟩ → ⟨hmethod⟩ | ⟨vmethod⟩
⟨hmethod⟩ → ((⟨just⟩ | ⟨scope⟩) { ⟨methodtail⟩ } )+
⟨vmethod⟩ → ((⟨vmethodheadvapplication⟩ | (( ⟨just⟩ | ⟨hanonym⟩) [⟨oname⟩] EOL | ⟨formationnamed⟩) ⟨methodtail⟩ { ⟨vmethodheadapplicationtail⟩ ⟨methodtail⟩ }
⟨vmethodheadapplicationtail⟩ → ([⟨oname⟩] ⟨vapplicationargs⟩ | EOL | ((⟨happlicationtail⟩ [⟨oname⟩] EOL)
| ⟨vmethodheadvapplication⟩)
⟨vmethodheadvapplication⟩ → ((⟨applicable⟩ | ⟨hmethod⟩ [⟨oname⟩] ⟨vapplicationargs⟩) | ((⟨reversed⟩ [⟨oname⟩] ⟨vapplicationargsreversed⟩)
| ⟨methodtail⟩)
⟨methodtail⟩ → "." (⟨finisher⟩ | ("␣" ⟨int⟩))
⟨beginner⟩ → "*" | "Q" | "QQ" | "$" | ⟨data⟩
⟨finisher⟩ → ⟨name⟩ | "@" | "^"
⟨reversed⟩ → ((⟨finisher⟩ | ("␣" ⟨int⟩)) "."
| ⟨aname⟩)
⟨aname⟩ → ">" ">" [ "!" ]
⟨fname⟩ → ⟨oname⟩ | ("␣" ⟨aname⟩)
⟨onameortname⟩ → ⟨oname⟩ | ⟨tname⟩
⟨oname⟩ → ⟨suffix⟩ [ "!" ]
⟨tname⟩ → ⟨tarrow⟩ ("@" | ⟨name⟩)
⟨tarrow⟩ → "␣" "+" ">" "␣"
⟨suffix⟩ → ⟨arrow⟩ ("@" | ⟨name⟩) [ "!" ]
⟨arrow⟩ → "␣" ">" "␣"
⟨scope⟩ → "(" (⟨happlication⟩ | ⟨hanonym⟩) ")"
⟨as⟩ → ":" (⟨name⟩ | ⟨int⟩)
⟨data⟩ → ⟨bytes⟩ | ⟨text⟩ | ⟨string⟩ | ⟨int⟩ | ⟨float⟩ | ⟨hex⟩
⟨commentary⟩ → "#" [ [ ^\r\n ] { [ ^\r\n\t ] }
| ⟨meta⟩)
⟨meta⟩ → "+" ⟨name⟩ { "␣" { [ ^\t\r\n ] } }+
⟨indent⟩ → "␣" "␣"
⟨linebreak⟩ → [ "\r" "\n"
| ⟨byte⟩)
⟨byte⟩ → [ 0-9A-F ] [ 0-9A-F ]
⟨empty_bytes⟩ → "_" "_"
⟨line_bytes⟩ → ⟨byte⟩ { "_" ⟨byte⟩ }+
⟨bytes⟩ → ⟨empty_bytes⟩ | ((⟨byte⟩ "_" | ((⟨line_bytes⟩ { "_" EOL ⟨line_bytes⟩ } )
| ⟨escape_sequence⟩)
⟨escape_sequence⟩ → "\{" [ [ btfnfr " '\{ } ] | ([ [ 0-3 ] [ 0-7 ] [ 0-7 ] | ( { "u" }+ ⟨byte⟩ ⟨byte⟩ ) )
| ⟨string⟩)
⟨string⟩ → DQ { [ [ ^ "\{ } \r\n ] | ⟨escape_sequence⟩ ) } DQ
⟨int⟩ → [ ("+" | "-") ] ("0" | ([ "0" ] [ 1-9 ] { [ 0-9 ] } ) )
| ⟨exponent⟩)
⟨exponent⟩ → ("e" | "E") [ ("+" | "-") ] { [ 0-9 ] }+
⟨float⟩ → [ ("+" | "-") ] { [ 0-9 ] }+ "." { [ 0-9 ] }+ [ ⟨exponent⟩ ]
| ⟨hex⟩)
⟨hex⟩ → "0x" { [ 0-9a-fA-F ] }+
⟨name⟩ → [ a-z ] { [ ^ _ . ! ' : ; ! ? ] [ - ] { \r\n\t\xF335 } }
| ⟨text⟩)
⟨text⟩ → DQ DQ DQ [ "\r" "\n" { ( [ ^ \{ } ] | ⟨escape_sequence⟩ ) } } DQ DQ DQ

```