```
Winzig     → <'program' Name Consts Types Dclns SubProgs Body
Name>

Consts     → <'consts' Const+>
           → 'consts'

Const      → <'const' Name ConstValue>

ConstValue → '<integer>'
           → '<char>'

Types      → <'types' Type+>
           → 'types'

Type       → <'type' Name LitList>

LitList    → <'lit' Name+>

SubProgs   → <'subprogs' Fcn*>

Fcn        → <'fcn' Name Params Name Consts Types Dclns Body Name>

Params     → <'params' Dcln+>

Dclns      → <'dclns' Dcln+>
           → 'dclns'

Dcln       → <'var' Name+ Name>

Body       → <'block' Statement+>

Statement  → <'output' OutExp+>
           → <'if' Expression Statement Statement?>
           → <'while' Expression Statement>
           → <'repeat' Statement+ Expression>
           → <'for' ForStat ForExp ForStat Statement>
           → <'loop' Statement+>
           → <'case' Expression Caseclauses OtherwiseClause>
           → <'read' Name+>
           → 'exit'
           → <'return' Expression>
           → '<null>'

OutExp     → <'integer' Expression>
           → <'string' StringNode>

StringNode → '<string>'
```

```
Caseclause → <'case_clause' CaseExpression+ Statement>

CaseExpression → <'..' ConstValue ConstValue>

OtherwiseClause → <'otherwise' Statement>

Assignment → <'assign' Name Expression>
           → <'swap' Name Name>

ForStat    → '<null>'

ForExp     → 'true'

Expression → <'<=' Term Term>
           → <'<' Term Term>
           → <'>=' Term Term>
           → <'>' Term Term>
           → <'=' Term Term>
           → <'<>' Term Term>

Term       → <'+' Term Factor>
           → <'-' Term Factor>
           → <'or' Term Factor>

Factor     → <'*' Factor Primary>
           → <'/' Factor Primary>
           → <'and' Factor Primary>
           → <'mod' Factor Primary>

Primary    → <'-' Primary>
           → <'not' Primary>
           → 'eof'
           → '<integer>'
           → '<char>'
           → <'call' Name Expression+>
           → <'succ' Expression>
           → <'pred' Expression>
           → <'chr' Expression>
           → <'ord' Expression>

Name       → '<identifier>'
```