THE C-- LANGUAGE

RESERVED WORDS

```
print
if
else
while
```

PUNCTUATION AND OPERATORS

`(`	`+`	`=`	`<`
`)`	`-`	`==`	`>`
`{`	`*`	`!=`	`<=`
`}`	`/`	`&&`	`>=`
`,`	`%`	`||`	

OTHER LEXICAL RULES

Each number consists of one or more digits, and denotes a non-negative integer.

Each identifier consists of one or more letters that do not form a reserved word. Reserved words and identifiers are case sensitive. That is, `if` denotes a reserved word, but `If` and `iF` and `IF` are each distinct identifiers.

Whitespace characters include blanks, tabs, line feeds, and carriage returns.

STATEMENTS

A program is a sequence of statements. Each statement is one of the following:

statement type	
assignment	identifier `=` expression
print	`print` expression1 , expression2 ... , expressionN
selection	`if` `(` expression `)` statement1 `else` statement2
iteration	`while` `(` expression `)` statement
compound	`{` statement1 statement2 ... statementN `}`

EXPRESSIONS

Binary operators have the same meanings, precedence, and associativity as in the C language. Parentheses force an evaluation order. There are no unary operators.

TYPES

Each identifier denotes a variable that has integer type and global scope.
Arithmetic operators (`+`, `-`, `*`, `/`, `%`) return integer values.
Relational and logical operators (`==`, `!=`, `<`, `>`, `<=`,`>=`, `&&`, `||`) also return integer values (`1` for *true*, `0` for *false*).
The conditional expressions in `if` and `while` statements must evaluate to `1` or `0`.

SEMANTICS

Semantics for `C--` are generally the same as for `C`, except for the new `print` statement.

```
print expression1 , expression2 ... , expressionN
```

displays the values of the *N* expressions on one line, separated by blanks, and followed by a newline.