

# AtomC - reguli lexicale

## Identificatori

```
ID: [a-zA-Z_] [a-zA-Z0-9_]*
```

## Cuvinte cheie

```
TYPE_CHAR: 'char'  
TYPE_DOUBLE: 'double'  
ELSE: 'else'  
IF: 'if'  
TYPE_INT: 'int'  
RETURN: 'return'  
STRUCT: 'struct'  
VOID: 'void'  
WHILE: 'while'
```

## Constante

```
INT: [0-9]+  
DOUBLE: [0-9]+ ( '.' [0-9]+ ( [eE] [+-]? [0-9]+ )? | ( '.' [0-9]+ )? [eE] [+-]? [0-9]+ )  
CHAR: [''] [^'] ['']  
STRING: [""] [^"]* [""]
```

## Delimitatori

```
COMMA: ','  
SEMICOLON: ';'   
LPAR: '('  
RPAR: ')'   
LBRACKET: '['  
RBRACKET: ']'   
LACC: '{'  
RACC: '}'   
END: '\\0' | EOF           // '\\0' - terminatorul de șir; EOF - codul C pentru End Of File
```

## Operatori

```
ADD: '+'  
SUB: '-'  
MUL: '*'  
DIV: '/'  
DOT: '.'  
AND: '&&'  
OR: '||'  
NOT: '!'  
ASSIGN: '='  
EQUAL: '=='
```

```
NOTEQ: '!='  
LESS: '<'  
LESSEQ: '<='  
GREATER: '>'  
GREATEREQ: '>='
```

## Spații - acestea nu formează atomi lexicali

```
SPACE: [ \n\r\t]           // conține și caracterul spațiu  
LINECOMMENT: '//' [^\n\r\0]*
```