

Objective-00

Análise Léxica

Cezar Augustus Signori

João Paulo Pizani

José João Junior



Ferramentas utilizadas

- ANTLR
 - Através da IDE ANTLRWorks
- Código gerado: Java



Categorias de tokens

- Símbolos e operadores
- Palavras reservadas
- Identificadores
- Literais
 - Numéricas
 - Inteiras
 - Ponto flutuante
 - Caractere
 - String



Símbolos e operadores

DQUOTE: ' "'	LSBCKT: '['	RSBCKT: ']'	COMMA: ','
MINUS: '-'	SEMICOL: ':'	LPAR: '('	RPAR: ')'
ATTR: '='	LBCKT: '{'	RBCKT: '}'	TIMES: '*'
AND: '&'	BITAND: '&&'	DIV: '/'	MOD: '%'
LT: '<'	GT: '>'	LTEQ: '<='	GTEQ: '>='
OR: ' '	BITOR: ' '	NOT: '!'	BITNOT: '~'
INC: '++'	DEC: '--'	EOL: ';'	PLUS: '+'



Palavras reservadas

IMPORT: '#import'

PROPERTY: '@property'

RETAIN: 'retain'

ASSIGN: 'assign'

COPY: 'copy'

READONLY: 'readonly'

STATIC: 'static'

WHILE_: 'while'

DO_: 'do'

SWITCH_: 'switch'

FOR_: 'for'

IN_: 'in'

IF_: 'if'

ELSE_: 'else'

IFACE: '@interface'

END: '@end'

PRIVATE: '@private'

IMPL: '@implementation'

CASE_: 'case'

DEFAULT_: 'default'

- Não há o comando 'break'



Identificadores

```
* ENGLISH_LETTER: ('a'..'z') | ('A'..'Z');  
* ID_SYMBOL: '_' | '?';  
* DIGIT: '0'..'9';
```

```
ID: ENGLISH_LETTER (ENGLISH_LETTER | DIGIT | ID_SYMBOL)*;
```

```
* FILE_NAME: (ENGLISH_LETTER | DIGIT | '_' | '.')+;  
PATH: '/'? FILE_NAME ('/' FILE_NAME)*;
```



Literais numéricas

* BIN_DIGITS: ('0'|'1')+

* OCT_DIGIT: ('0'..'7')

* OCT_DIGITS: OCT_DIGIT+

* DEC_DIGITS: DIGIT+

* HEX_DIGIT: ('0'..'9') | ('A'..'F')

* HEX_DIGITS: HEX_DIGIT+

* EXPONENT: 'e' ('-'|'+')? DEC_DIGITS

* NONDEC_LITERAL:

'#' ('b' BIN_DIGITS|'o' OCT_DIGITS|'x' HEX_DIGITS)

INTEGER_LITERAL: DEC_DIGITS | NONDEC_LITERAL;

FLOAT_LITERAL: DEC_DIGITS '.' DEC_DIGITS EXPONENT?;



Strings e Caracteres

* ESCAPE: '\\\ ('b'|'t'|'n'|'f'|'r'|'\"'|'\''|'\\\\')

| UNICODE_ESCAPE | OCTAL_ESCAPE

* OCTAL_ESCAPE: '\\\ ('0'..'3')? OCT_DIGIT? OCT_DIGIT

* UNICODE_ESCAPE: '\\\ 'u' HEX_DIGIT HEX_DIGIT HEX_DIGIT HEX_DIGIT

STRING_LITERAL: '\"' (ESCAPE | ~('\\\\'|'\"')) * '\"'

CHAR_LITERAL: '\\\ (' | ESCAPE | ~('\''|'\\\\')) '\\\'



Exemplo de código

```
@interface Interface : Dad {
    ClassName variable;
    int var3;
    int[4] sampleArray = {12, 15, 7};
}

- (void)methodWithParam1:(float)param1 andParam2:(int)param2;

@property(retain) variable;
@property(readonly) sampleArray;

@end
```

