

Gramática da linguagem IQui

Linguagens usadas: Java + Delphi

Gramática da linguagem Iqui desenvolvida para a disciplina de compiladores.
Código disponibilizado em <https://Github.com/Iudek>

<escopo> ::= begin <declaração de variável>* <comandos>* end

<comandos> ::= <ExpressãoAtribuição> |
<ExpressãoCondicional> | <declaração de variável>

<declaração de variável > ::= <variável>+ : <tipo><instrução vazia>

<tipo> ::= int | char | double | boolean

<operador> ::= <operadorLogico> | <operadorAritmedico> |
<operadorRelacional>

<operadorLogico> ::= && | ||

<operadorAritmético> ::= + | - | * | / | %

<operadorRelacional> ::= > | < | >= | <= | == | !=

<expressões> ::= <expressão> (<operador> <expressão>)*
<instrução vazia>* | <negação>

<expressão> ::= <variável>

<negação> ::= !<expressão>

< instrução vazia > ::= ;

<incremento> ::= ++

<decremento> ::= --

<incrementar variavel> ::= <variavel><incremento><instrução vazia>
 <decrementar variavel> ::= <variavel><decremento><instrução vazia>
 <ExpressãoAtribuição> ::= <variável>=<expressões><instrução vazia>
 <ExpressãoCondicional> ::= if (<expressões>) begin <comandos>+ end (else begin <comandos>+ end)?
 <Entrada> ::= Scanner.in.readLine() <instrução vazia>
 <Saida> ::= Sprintf("<mensagem>")<instrução vazia>
 <variável> ::= <letra> (<letra>|<dígito>)*
 <letra> ::= a..z|A..Z
 <dígito> ::= 0..9
 <char> ::= <letra>*|<dígito> *
 <boolean> ::= "true"|"false"
 <double> ::= <dígito> (<dígito>)*
 <int> ::= dígito(dígito)*
 <mensagem> ::= <letra>|<dígito>(<letra>|<dígito>)*
 <comentario> ::= */(<mensagem>)/

Expressões regulares

tipo ::= [i][n][t]\$|[c][h][a][r]\$|[d][o][u][b][l][e]\$|[b][o][o][l][e][a][n]\$
 variável ::= ^([a-z]|[A-Z])+[0-9]*
 Mensagem ::= ^\"(\\w|\\d|[_])*\"

incremento ::= $^{+}[+]\$$

decremento ::= $^{-}[-]\$$

operadorLogico ::= $^{&}[\&]\$|^{[]}[]\$$

operadorAritmético ::= $^{+}[\$]|^{-}[\$]|^{*}[\$]|^{/}[\$]|^{ \% }[\$]$

operadorRelacional ::= $^{>}[\$]|^{<}[\$]|^{>}[=]\$|^{<}[=]\$|^{=[=]}\$|^{![=]}\$$

negação ::= $^{!}[\$]$

instrução vazia ::= $^{;}[\$]$

char ::= $^{[:\text{alpha:}]}\$|^{[0-9]}+$

boolean ::= $^{[t][r][u][e]}\$|^{[f][a][l][s][e]}\$$

double ::= $^{[0-9]}+^{[.]}[0-9]+$

int ::= $^{[0-9]}+$

conjunto de elementos terminais

$T = \{0..9, A..Z, a..z, +, -, *, /, \%, =, ==, !=, !, ++, --, \&\&, ||, ;, >, >=, <, <=, *, \backslash *, (,)\}$