SINTAXE DA LINGUAGEM U

Revisão C

<bloco> ::=

 { <lista de declarações> <lista de comandos> }

<caracteres> ::=

 *qualquer seqüência de caracteres, exceto* \*/

<caracteres visíveis> ::=

 *qualquer seqüência de caracteres visíveis*

<chamada de função> ::=

 <nome> ((<lista de expressões>|))

<comando> ::=

 <comando simples> |

 <comando condicional> |

 <comando iterativo> |

 { <lista de comandos> } |

 

<comando condicional> ::=

 if <expressão> then <comando> |

 if <expressão> then <comando> else <comando>

<comando iterativo> ::=

 while <expressão> do <comando> |

 repeat <comando> until <expressão> |

 for <nome de variável> := <expressão> to <expressão> (step <expressão>|) do <comando>

<comando simples> ::=

 <nome de variável> := <expressão> |

 <chamada de função> |

 return <expressão>

<comentário> ::=

 // <caracteres visíveis> <eol> |

 /\* <caracteres> \*/

<declaração> ::=

 <declaração de variável> |

 <declaração de função>

<declaração de função> ::=

 (<tipo simples>|void) <nome> ((<lista de parâmetros>|)) <bloco>

<declaração de variável> ::=

 <tipo> <lista de nomes>

<dígito> ::=

 0 | 1 | 2 | ... | 9

<eol> ::=

 *caracter de fim de linha*

<expressão> ::=

 <expressão simples> (<operador relacional> <expressão simples>|)

<expressão simples> ::=

 <expressão simples> <operador aditivo> <termo> |

 <termo>

<fator> ::=

 <nome de variável> |

 <chamada de função> |

 <número> |

 ( <expressão> )

<letra> ::=

 a | b | c | ... | z

<lista de comandos> ::=

 <lista de comandos> ; <comando> |

 <comando>

<lista de declarações> ::=

 <declaração> ; <lista de declarações> |

 

<lista de expressões> ::=

 <lista de expressões> , <expressão> |

 <expressão>

<lista de nomes> ::=

 <lista de nomes> , <nome> |

 <nome>

<lista de parâmetros> ::=

 <lista de parâmetros> , <parâmetro> |

 <parâmetro>

<nome> ::=

 <nome> <letra> |

 <nome> <dígito> |

 <letra>

<nome de variável> ::=

 <nome> (| [<expressão>] | [<expressão>,<expressão>])

<número> ::=

 <número inteiro> ( | . | . <número inteiro>) |

 . <número inteiro>

<número inteiro> ::=

 <número inteiro> <dígito> |

 <dígito>

<operador aditivo> ::=

 + | - | or

<operador multiplicativo> ::=

 \* | / | and

<operador relacional> ::=

 > | < | >= | <= | <> | =

<parâmetro> ::=

 <tipo simples> <nome>

<programa> ::=

 main <bloco>

<termo> ::=

 <termo> <operador multiplicativo> <fator> |

 <fator>

<tipo> ::=

 <tipo simples> |

 <tipo agregado>

<tipo agregado> ::=

 [(<número inteiro>,< número inteiro> | <número inteiro>)] <tipo simples>

<tipo simples> ::=

 int |

 bool |

 float