



Disciplina: LINGUAGENS, AUTÔMATOS E COMPUTAÇÃO

Unidade de Aprendizagem: UA2 | LINGUAGENS LIVRES DE CONTEXTO



Importante!

O grupo deve listar os nomes de **TODOS** os participantes. Caso o nome de algum participante do grupo não seja listado, esse estudante não receberá nota.

Grupo de Trabalho

Estudante 1: Brenda Pereira

Estudante 2: Bruno Bolzan

Estudante 3: Felipe Freitas

Estudante 4: Luiza Heller

UA2 | Avaliação de Aprendizagem

Proposta | Construir, de forma cooperativa, uma especificação de gramática livre do contexto para uma linguagem de programação cuja descrição da estrutura foi descrita no módulo de aprendizagem. O sistema a ser utilizado para a descrição da gramática será o ANTLR (disponível gratuitamente em <https://www.antlr.org/>). E os testes de funcionamento da gramática deverão ser realizados através da versão on-line do sistema, nomeado ANTLR Lab (disponível gratuitamente em <http://lab.antlr.org/>).

Registre neste espaço a produção do grupo de trabalho ▼



Código de teste:

Observação importante: o material disponibilizado no Moodle diz que cada comando deve finalizar em ‘;’, porém o código disponibilizado ignora esta restrição, que só deveria ser ignorada por blocos de comandos, definidos como sendo delimitados por ‘begin’ e ‘end’, por isso, alterações foram feitas para incluir o terminador e permitir avaliação adequada.

progr exemplo

begin

clear x;

move 100 to x;

if (x)

then

begin

decr x;

move x to y;

end

else

incr x;

;

while (y)

do decr y;

;

end



```
/* Nome da Gramática */
grammar BareBones;

/* Regra Inicial */
program: 'progr' VARIABEL block EOF;

/* Blocos de comandos são expressões entre begin e end */
block: 'begin' expression* 'end';

/* As expressões disponíveis na linguagem são: */
expression: block
    | clear
    | incr
    | decr
    | move
    | if_then_else
    | while
    ;

/* Sintaxe de cada expressão: (sempre terminando em ; ) */
clear: 'clear' VARIABEL TERMINATOR;
incr: 'incr' VARIABEL TERMINATOR;
decr: 'decr' VARIABEL TERMINATOR;
move: 'move' (VARIABEL | LITERAL) 'to' VARIABEL TERMINATOR;
if_then_else: 'if' '(' (VARIABEL)' 'then' expression+ 'else' expression+ TERMINATOR;
while: 'while' '(' (VARIABEL)' 'do' expression+ TERMINATOR;

/* Definições importantes */

/* Ignorar espaços em branco e afins */
BRANCO: [ \t\r\n]+ -> skip;

/* Números não podem ter 0 à esquerda */
LITERAL: '0' | [1-9][0-9]*;

/* Variáveis não podem começar com mas podem conter _ */
VARIABEL: [a-z][a-z_]*;

/* Definição do final de cada comando */
TERMINATOR: ';;';
```