



**SERVIÇO PÚBLICO FEDERAL**  
**MINISTÉRIO DA EDUCAÇÃO E DESPORTO**  
**CENTRO FEDERAL DE EDUCAÇÃO TECNOLÓGICA DE MINAS GERAIS**

<b>Curso:</b> Informática	<b>Série/Ano:</b> 3ª	<b>Turno:</b> Diurno
<b>Disciplina:</b> Linguagem de Programação II	<b>Atividade:</b> 1º Trabalho Prático	
<b>Valor:</b> 20,0 pontos	<b>Tipo de Atividade:</b> Trabalho em Grupo	
<b>Professor:</b> Cristiano Maffort	<b>Data para Entrega:</b> até 27/04/2018	
<b>Observações:</b> <ol style="list-style-type: none"><li>1. O trabalho deverá ser entregue através do Moodle, em arquivo único (no formato zip – obrigatório!), o qual deverá possuir um arquivo readme.txt contendo o nome dos alunos.</li><li>2. Na realização deste Trabalho Prático o aluno poderá utilizar apenas a API básica da linguagem (java.util.*, java.io.*, java.lang.*).</li><li>3. Na avaliação serão consideradas características como: corretude, organização, modularidade, legibilidade, extensibilidade e robustez.</li><li>4. Será atribuída nota 0 (zero) a eventuais cópias ou quando o aluno não for capaz de demonstrar/comprovar ser o autor do trabalho por meio da arguição que poderá realizada, a critério do professor.</li></ol>		

Implemente em Java um interpretador para a linguagem *cefetiny*, a qual possui a seguinte gramática:

```
<programa>..... <lista_comandos> "end"
<lista_comandos> ..... (<comando>)>=1
<comando> ..... "print" "("<Expressao>)" ||
                    "println" ||
                    "readInt" "(" <variavel> ")" ||
                    <comando_atrib> ||
                    <comando_if> ||
                    <comando_while> ||
                    <comando_for>
<comando_atrib> ..... <variavel> "!=" <Expressao>
<comando_if> ..... "if" "("<Expressao>)" "then"
                    <lista_comandos>
                    ["else"]
                    <lista_comandos>
                    "endif"
<comando_while>..... "while" "("<Expressao>)" "do"
                    <lista_comandos>
                    "endwhile"
<comando_for>..... "for" <comando_atrib> ("to" || "downto") <Expressao> "do"
                    <lista_comandos>
                    "endfor"
<variavel> ..... <letra>(<letra> || <digito>)>=0
<letra> ..... "a" || "b" || ... || "z"
<digito>..... "0" || "1" || ... || "9"
<Expressão>..... <Expressao> ( <relop> || <addop> || <mulop> ) <Expressao> ||
                    <unop> "("<Expressao>)" ||
                    <constante> || <variável> ||
```

<addop> ..... "+" || "-" || "or"  
 <mulop> ..... "\*" || "/" || "mod" || "div" || "and"  
 <unop> ..... "sqrt" || "not"  
 <constante> ..... (<digito>)<sup>>=1</sup> || true || false || <string>  
 <string> ..... "" (<letra> || <digito>)<sup>>=0</sup>""

O interpretador deverá possuir uma interface gráfica de linha de comando, lendo o programa de um arquivo texto. Maiores informações sobre a implementação do trabalho serão dadas em sala de aula.

#### **Cronograma de implementação e pontuação**

<b>Aula</b>	<b>Comandos</b>
01	atribuição
02	print, println, readint
03	if
04	while
05	for

**Sugestão: Comece hoje!!! Você nunca terá tanto tempo...**