

### Trabalho Prático – Prova 3 (valor 2 pontos)

Considere a definição da linguagem a seguir.

Programa	→ DeclLista
DeclLista	→ Decl DeclLista   Cmd CmdLista   $\epsilon$
Decl	→ Tipo ID ;   Tipo ID = Expressao ;
CmdLista	→ Cmd CmdLista   $\epsilon$
Cmd	→ ID = Expressao ;   if ( ExpressaoRel ) { CmdLista }   while ( ExpressaoRel ) { CmdLista }
Expressao	→ Expressao + Termo   Expressao - Termo   Termo
Termo	→ Termo * Fator   Termo / Fator   Fator
Fator	→ ( Expressao )   NUM   ID   ! Fator
ExpressaoRel	→ Expressao OpRel Expressao
OpRel	→ <   <=   >   >=   ==   !=
Tipo	→ int   float
ID	→ [a-zA-Z_][a-zA-Z0-9_]*
NUM	→ [0-9]+(\.[0-9]+)?

Crie, usando o Flex / Bison, os analisadores léxicos e sintáticos para a linguagem.

Como entrada deve ser usado um arquivo com um código na linguagem (exemplo na próxima página).

Como saída, espera-se o resultado da análise (“Finalizada com sucesso” ou mostrando os erros encontrados e citando a linha).

Use os arquivos do Teams como base.

## Exemplo de código para teste

```
int x;  
float y;  
int z = 10;  
  
x = 5;  
y = 3.14;  
  
if (x < z)  
{  
    x = x + 1;  
}  
  
while (x < 20)  
{  
    x = x + 2;  
}
```