

# INF441 - Roteiro para Aula Prática

## Análise Semântica - parte II

Baixe o arquivo

[https://drive.google.com/file/d/1IVNEufrZ7paitb6G3E\\_lok4r-Y1ehUHh/view?usp=sharing](https://drive.google.com/file/d/1IVNEufrZ7paitb6G3E_lok4r-Y1ehUHh/view?usp=sharing)

Em seguida, descompacte o conteúdo em uma pasta qualquer, em um computador que tenha Eclipse + ANTLR4 instalado.

Abra o IDE Eclipse e importe o projeto descompactado. Antes de iniciar este roteiro, verifique se o IDE Eclipse está devidamente configurado, seguindo as instruções em:

<https://drive.google.com/file/d/1TWBsuBXlzZmSvHniZQ80POzoCVSaPbhP/view?usp=sharing>

Em seguida, siga as instruções do roteiro para atividades relacionadas ao uso e símbolos durante a análise semântica de programas MicroJava.

### Instruções Iniciais

Abra o projeto Eclipse mencionado acima. Neste roteiro, vamos implementar a segunda parte da análise semântica de programas MicroJava, processando o uso de símbolos dentro

. O *listener SemanticListener* está parcialmente implementado e as instruções deste roteiro irão guiá-lo para completar a implementação da segunda parte da análise semântica. Essas instruções estão dentro do próprio arquivo *SemanticListener.java*, em forma de comentários. Em alguns pontos, está inserido um comentário

```
//TODO
```

que significa “to do” ( a ser feito). Esse comentário ressalta pontos em que você deverá inserir novo código Java para implementar partes da análise semântica.

No programa piloto, você pode encontrar o comando

```
symbolTable.setDebug(true);
```

que liga o “debug” da tabela de símbolos. Quando ligado, toda inserção de símbolo ou empilhamento/desempilhamento de ambientes na tabela é reportada na saída padrão. Pode ser útil manter esse debug ligado enquanto as atividades desse roteiro são desenvolvidas, para facilitar o acompanhamento.

### Instruções para entrega

Entregue o arquivo **SemanticListener.java** construído e o arquivo **teste01.txt** com testes significativos.