

## Trabalho – Parte 3: Análise Semântica e Geração de Código Intermediário

### Objetivos:

#### 1) Implementar o analisador semântico da linguagem proposta.

- As questões semânticas devem ser especificadas de acordo com as características da linguagem, proposta na etapa 1.

#### 2) Implementar um gerador de código intermediário para o compilador:

- Como linguagem intermediária usar:
  - Código de 3 endereços (Livro Dragão);
  - Árvore Sintática Abstrata - AST (Livro Dragão); ou
  - LLVM IR (<https://llvm.org/docs/LangRef.html>)

No caso das AST, o resultado deverá ser apresentado de forma gráfica. Sugestão: gerar arquivo na linguagem DOT (<https://graphviz.org/doc/info/lang.html>) e visualizar o gráfico no website Viz.js (<http://viz-js.com/>) ou outro visualizador compatível.

### O que deve ser entregue:

- Documento com a especificação dos aspectos semânticos da linguagem;
- Código-fonte do compilador completo;
- Código-fonte de entrada na linguagem proposta. O código deve conter todas as primitivas definidas na linguagem. Todas as questões semânticas devem estar presentes no código.
- Os arquivos devem ser enviados em um arquivo compactado para o e-mail: [guilherme.galante@unioeste.br](mailto:guilherme.galante@unioeste.br)

### Avaliação (total 100 pontos):

- Documento dos aspectos semânticos (10 pontos)
- Implementação do analisador semântico (40 pontos)
- Implementação do gerador de código (40 pontos)
- Apresentação (10 pontos)

**Data para Entrega:** 26/08 – 14:59.

**Apresentações:** a partir do dia 26/08 – horários de aula