

## Compiladores 2025\_1

### Trabalho – P2 parte #2 (5,0)

Poderá ser feito **em quatro (aluno(o) que não comparece deve ser individual)**. Para evitar trabalhos semelhantes/iguais é necessário informar o que será desenvolvido e participar dos encontros presenciais. Como é um trabalho, então não tem 2ª chamada dessa parte #2.

Encontros presenciais para tirar **dúvidas** e **mostrar** o andamento do trabalho. Será pontuado em até **2,0**. *Quem não aparecer, perderá essa pontuação. Não basta ir à aula, tem que mostrar o andamento do trabalho.* Pode ser pelo menos uma vez na semana.

Entrega até **25/06/25** – vídeo de no máximo 7 minutos com a apresentação e execução do código desenvolvido. Ver requisitos a seguir.

### Objetivo do trabalho

Explorar e aplicar os conceitos de análise léxica, sintática e semântica vistos na disciplina Compiladores ao modificar um gerador de analisador.

### Passos para a realização do Trabalho

- 1º. Pesquisar sobre Geradores de Analisadores (léxico/sintático). Buscar de preferência na linguagem de programação que domina/gosta. (até 28/05)
- 2º. Baixar, executar e analisar um exemplo disponível na linguagem escolhida. *Normalmente é encontrado o exemplo de uma calculadora aritmética (calc). Por isso, não serão aceitos trabalhos que sejam calculadoras ou que apenas adicionem novos operadores/funções como potência, raiz quadrada etc.* (até 28/05)
- 3º. Modificar o exemplo, incluindo definições, regras e ações nos geradores de analisadores léxicos e sintáticos (ex.: FLEX/BISON, PLY, SLY, RPLY, JAVACC etc). *Alguns alunos preferem iniciar do zero.* (até 20/06)
- 4º. Preparar uma apresentação (máximo 7 min) sobre o gerador de analisador escolhido. Requisitos mínimos:
  - i. Informar o analisador de gerador escolhido (linguagem).
  - ii. Apresentar onde foram realizadas as modificações e as inclusões no código, dando ênfase nas definições e nas regras da análise léxica (criação das expressões regulares e onde elas ficam), nas regras gramaticais (criação das produções, regras de precedência, eliminação de ambiguidade, se necessário etc.) e quais são as ações/regras semânticas criadas (mostrar as ações nas produções). Apresentar código de usuário, caso tenha criado.
  - iii. Mostrar a tabela com as produções e ações semânticas (ver exemplo nos slides de Tradução Dirigida pela Sintaxe ou Geração de Código).
  - iv. Mostrar um exemplo de uma árvore de derivação e de uma árvore de derivação anotada de uma sentença que seja aceita pela sua linguagem.
  - v. Apresentar um exemplo **executando** o código modificado. Não é para printar a tela.
- 5º. Enviar pela tarefa no Teams (até 25/06): vídeo da apresentação (máximo 7 min), arquivo do código modificado (código(s) fonte(s)), sinalizando no(s) código(s) as modificações realizadas (como comentário no código).

**Obs. Importante: a P2 parte #1 –será uma avaliação individual presencial que valerá 5,0. Provavelmente acontecerá no dia 18/06. Favor, acompanhar as informações nas aulas e/ou postagens no Teams.**