

INE5622 – Introdução a Compiladores

Implementação de Compilador 1 (IC1)

Analizador léxico e sintático

Os trabalhos práticos desta disciplina constituem na elaboração de uma linguagem de programação imperativa e na construção de um interpretador para ela. A execução deles está dividida em duas partes (IC1 e IC2), onde os principais entregáveis, em sequência, são: (IC1) a gramática ANTLR4 usada para gerar o analisador léxico e sintático em Python; e (IC2) o interpretador completo da linguagem com análise léxica, sintática e semântica e geração de código.

Para o contexto dessa entrega, aborda-se a primeira etapa de trabalhos (IC1).

1 Elaboração da linguagem de programação

As **características mínimas** que a linguagem precisa ter para a primeira entrega são:

- 4 tipos, sendo `int` e `float` obrigatórios¹;
- Instância e uso de classes;
- Definição e chamada de funções;
- Estruturas de controle `if-then-else` e `switch-case`.
- Laços `for` e `while`;
- Operações `*`, `/`, `+`, `-`, `>`, `<`, `<=`, `>=` e `==` seguindo essa ordem de precedência²;
- Possibilidade de definir a precedência de operações usando parênteses.

Além das características mínimas, a linguagem proposta deve ter ao menos uma outra característica selecionada pela equipe. Cada equipe deve selecionar uma característica própria.

¹A linguagem pode ter tipos de diferentes precisões, porém eles são contados como um.

²Operações da esquerda tem maior precedência.

2 Entrega

O trabalho deve ser feito em **equipes de até 4 pessoas** e entregue pelo MO-ODLE por **apenas um membro da equipe**.

A primeira entrega constitui em um relatório, um analisador léxico e sintático e exemplos de código.

- O relatório deve ser entregue em **pdf** e ter:
 - O nome de todos os membros da equipe;
 - A contribuição de cada membro no trabalho até a entrega;
 - Documentação da linguagem;
 - Exemplos de código.
 - * A documentação e os exemplos de código devem demonstrar/enfatizar que a linguagem proposta cumpre as características mínimas e possui ao menos uma característica adicional;
- O analisador léxico e sintático deve escrito apenas em Python e ser gerado a partir de uma gramática **.g4** compatível com **ANTLR**³;
- Os exemplos de código devem contemplar todas construções possíveis da linguagem.
- Adicionalmente, dois exemplos de código com erros sintáticos⁴.

3 Avaliação

O trabalho valerá nota de 0 a 10, entendendo que o cumprimento de cada item depende da sua descrição no relatório e da implementação da gramática no arquivo **g4**. A linguagem deve conter todas as **características mínimas** para poder ser avaliada. Caso essas características forem cumpridas, a pontuação obedecerá a seguinte distribuição. Caso não forem cumpridas as **características mínimas**, o trabalho receberá nota zero (0).

- Implementação e adaptação do código-fonte aos tipos: até 2 pontos;
- Implementação das estruturas de controle **switch-case** e **for**: até 2 pontos;
- Implementação da criação e uso de classes: até 1 ponto;
- Característica adicional: até 2 pontos;
- Relatório: até 2 pontos;
- Exemplos de código-fonte: até 1 ponto.

³O analisador não deve ter nenhuma dependência além de `antlr4-python3-runtime 4.9.1`

⁴Deixe explícito no nome do arquivo que se trata de um exemplo de código com erro.

4 Links úteis

- Getting Started with ANTLR v4
<https://github.com/antlr/antlr4/blob/master/doc/getting-started.md>
- Linguagem de exemplo usada na disciplina
<https://gitlab.com/evandro-crr/cmm>