

Compiladores (Recurso)

Linguagem *FigLan*

Departamento de Electrónica, Telecomunicações e Informática
Universidade de Aveiro

Julho de 2025

Objectivos

O objectivo geral deste trabalho é o desenvolvimento de uma linguagem simples de programação compilada – i.e. que crie programas equivalentes ao programa a compilar numa linguagem de programação genérica (Java, C++, Python, ...) – que faça a manipulação de figuras geométricas.

Nesta linguagem, as figuras são definidas como tipos de dados. Sendo um trabalho de recurso, e como tal muito limitado no tempo de desenvolvimento, e incapaz de substituir completamente o trabalho de grupo que decorreu durante as aulas, vão apenas ser exigidos, e avaliados, requisitos mínimos.

Esta linguagem vai ter tipos de figuras elementares (`point`, `line`, `circle`), assim como tipos numéricos (`integer`, `real`) e de texto (`text`).

Características mínimas

Os exemplos `example1.fl`, `example2.fl` e `example3.fl` indicam algum código fonte que tem de ser aceite (e devidamente compilado) pela linguagem a desenvolver.

A linguagem deve implementar:

- Instruções para definir tipos de dados das seguintes figuras elementares: ponto (`point`), linha (`line`) e círculo (`circle`). À imagem dos tipos de dados não primitivos da linguagem Java, a declaração de um tipo de dados figura apenas permite que se referencie uma figura (`object`), sendo para tal necessário criar essa figura (instrução `new`). De forma similar, a existência de um `object`, não implica o seu desenho. O desenho de figuras é feito (ou desfeito) por instruções separadas.
- Os tipos de dados inteiro, real e texto.
- Aceitar expressões aritméticas standard para os tipos de dados numéricos (incluindo a divisão inteira, com um operador para o quociente e outro para o resto). Aceita a operação de concatenação de texto (operador da soma).

- Instrução de escrita no *standard output* (com e sem mudança de linha no fim).
- Instrução de leitura de texto a partir do *standard input*.
- Operadores de conversão entre tipos de dados (por exemplo, `text(10)` para converter para texto; ou `integer("10")` para converter para inteiro).
- Instrução que mostra (desenha) figuras numa janela gráfica (pode partir do código base disponível em `java-test`).
- Instrução que esconde figuras da janela gráfica.
- Instrução de pausa (em milissegundos).
- Instrução de iteração (loop) num intervalo de inteiros.
- Verificação semântica do sistema de tipos.
- A geração de código deve utilizar a biblioteca `StringTemplate`, sendo valorizada uma geração de código de expressões com triplo-endereço (ver secções 2 e 3 do Tema 4 (Síntese) das teórico-práticas).

Desafio

Consegue, usando a linguagem que criou, desenhar a cara animada de uma pessoa a piscar os olhos várias vezes? (Usando círculos e linhas)