



universidade
de aveiro

deti

universidade de aveiro
departamento de eletrónica,
telecomunicações e informática

Compiladores

Trabalho Prático

Bernardo Rodrigues | 88835

João Dias | 89236

João Vasconcelos | 89022

Tomás Batista | 89296

Vinícius Ribeiro | 82773

Índice

Introdução	3
Desenvolvimento	4
Taxa de contribuição de cada elemento	5
Conclusão	6
Bibliografia	6

Introdução

Foi-nos proposto o desenvolvimento de duas linguagens: uma para um **compilador** e outra para **ler informação estruturada**.

O grupo escolheu o desenvolvimento de uma linguagem para **análise dimensional**, que tem como objetivo principal estender o sistema de tipos de uma linguagem de programação com a possibilidade de definir dimensões distinta a expressões numéricas.

Desenvolvimento

Como trabalho inicial, começámos por modelar como queríamos que a nossa linguagem se comportasse, como iria ser definida, o que iria ser permitido e proibido, as suas limitações e as suas capacidades. Primeiramente, desenhamos um esboço da linguagem (que, inevitavelmente, sofreu alterações) e definimos as unidades com que iríamos trabalhar (Ver [linguagem.md](#) e [unidades.md](#)).

Posto este trabalho, o nosso próximo passo foi começar a modelar exemplos do uso da nossa linguagem de modo a clarificar as ideias e poder extrair quaisquer conclusões que não estivessem devidamente esclarecidas. (Ver [linguagem.md](#))

Por fim, com todos os detalhes bem definidos e os objetivos estabelecidos, demos início à concepção da nossa linguagem.

Primeiramente, foi estabelecida a gramática da nossa linguagem (Ver [Calculator.g4](#)). Posto isto recorremos ao uso de Visitors para percorrer a árvore sintática (Ver [Calculator_visitor.java](#)) e realizar assim o trabalho. Usámos como linguagem destino de geração de código a linguagem **Java**.

Preparámos um conjunto de exemplos do uso da nossa linguagem e das diferentes funcionalidades de modo a permitir melhor compreensão e uso da linguagem. Os exercícios encontram-se em anexo na mesma pasta que o relatório. (Ver [ex1.txt](#) até [ex5.txt](#))

A linguagem permite ao utilizador realizar operações de adição, subtração e multiplicação a várias unidades dimensionais (como por exemplo adicionar $1\text{ m} + 120\text{ cm} = 2.20\text{ m}$) certificando-se que as unidades são compatíveis (impossível somar metros com gramas, mas é possível multiplicar) e efetuando conversões caso necessário. Adicionalmente suporta também unidades adimensionais (realizando corretamente operações como $4\text{ m} * 2.5 = 10\text{ m}$)

Suporta também a criação de variáveis podendo estas ser definidas com o comando: 'nome_variavel' = 'valor' 'dimensão' (exemplo: $a=10\text{m}$ ou $b=3\text{cm} + 4\text{mm}$)

É possível converter uma unidade para uma grandeza diferente com o comando (3m to cm ou 'variável' to cm)

Devido à maneira como a linguagem está estruturada é possível usar unidades do sistema métrico juntamente com unidades do sistema imperial (ou uma outra unidade desde que esteja nas tabelas de conversão)

Também foram implementadas as operações de IF, WHILE e FOR que funcionam de forma semelhante às da linguagem Java.

Taxa de contribuição de cada elemento

88835 - Bernardo Rodrigues: 37.5%

89236 - João Dias: 15%

89022 - João Vasconcelos: 5%

89296 - Tomás Batista: 37.5%

82773 - Vinícius Ribeiro: 5%

Conclusão

Este trabalho prático permitiu ao grupo aprofundar conhecimentos sobre ANTLR4, criação de uma linguagem, de uma gramática, entre outros. Levou também a uma melhor percepção de como uma linguagem funciona, como são feitas as análises sintáticas e como é percorrida a árvore sintática. Em suma, foi benéfico para o grupo.

Bibliografia

- [Repositório Oficial do ANTLR4](#)
- [Tiny Language - ANTLR4](#)
- [The ANTLR Mega Tutorial](#)
- [SI - Wikipedia](#)