

# Linguagens Formais e Autómatos / Compiladores

## Linguagem *bdex*

Departamento de Electrónica, Telecomunicações e Informática  
Universidade de Aveiro

maio de 2021

---

### Objetivos

O objectivo geral deste trabalho é o desenvolvimento de uma linguagem de programação compilada – i.e. que crie programas numa linguagem de programação genérica (Java, C++, Python, ...) – que junte algumas características de bases de dados com funcionalidades de manipulação de tabelas como as que encontramos em aplicações como o EXCEL.

Este objectivo geral pode ser decomposto em 4 componentes:

- Definição da linguagem principal 'bdex' que permita definir programas para manipular tabelas.
- Construção de um compilador que permite transformar uma descrição nessa linguagem num programa na linguagem genérica escolhida.
- Definição de uma linguagem secundária que permita definir tabelas em texto literal.
- Dotação da linguagem principal de meios para incorporar tabelas definidas pela linguagem secundária.

### Características da solução

Apresentam-se a seguir um conjunto de características que a solução desenvolvida pode ou deve contemplar. Essas características estão classificadas a 3 níveis:

- obrigatória – característica que a solução tem obrigatoriamente que implementar;
- desejável – característica não obrigatória, mas fortemente desejável que seja implementada pela solução;
- opcional – característica adicional apenas considerada para avaliação se as obrigatórias e as desejáveis tiverem sido contempladas na solução.

## Características obrigatórias

- Deve suportar nativamente o conceito de tabela:
  - Uma tabela é composta por campos (representados por colunas), cada um com um nome distinto.
  - Deve ser possível acrescentar, ou remover, linhas de valores a uma tabela.
  - Os valores da tabela devem poder ser do tipo numérico (número inteiro ou real) ou do tipo texto.
- Deve suportar pelo menos o seguinte conjunto de operações sobre tabelas (quando aplicável, implementadas na gramática de forma recursiva):
  - Extrair tabela a partir de outra tabela por secção de colunas e/ou linhas da tabela base usando como critério de selecção a igualdade de valores (colunas com um determinado nome, e/ou linhas com determinados valores numa coluna).
  - Junção de tabelas. Esta operação deve gerar uma nova tabela com as colunas e linhas da primeira tabela em primeiro lugar, seguida pelas colunas e linhas da segunda tabela. Os valores não preenchidas devem ficar com os valores por omissão de cada tipo de dados (zero nos numéricos e texto vazio nos do tipo texto).
- Conter uma linguagem secundária de leitura de tabelas em ficheiros de texto. Esta linguagem deve ser simples e diferente da linguagem principal. Podem considerar ficheiros CSV como requisito mínimo. Sobre o uso desta linguagem na linguagem principal:
  - A linguagem principal deve ter instruções para leitura de tabelas recorrendo a esta linguagem secundária (que, neste sentido, se comportará como um interpretador).
  - Deve também haver uma instrução de escrita que armazena uma tabela num formato aceite pela linguagem secundária.
- Deve haver uma separação clara entre a definição de tabelas (seja ela qual for) e a sua escrita no terminal.
- Deve permitir a definição e uso de variáveis (que possam conter valores numéricos, texto ou tabelas).
- Deve possuir mecanismo de leitura, que permita a interação de um utilizador com o programa (pelo menos, para ler valores numéricos ou de texto).
- Deve possuir mecanismo de escrita, que permita a escrita no terminal de tabelas, valores numéricos e texto.
- Deve conter pelo menos uma instrução iterativa que permita percorrer os valores de uma tabela. No mínimo, deve ser possível aceder, um a um, aos valores de uma tabela com apenas uma coluna.

## Características desejáveis

- Implementar a noção de tuplo que possa contar uma linha de uma tabela (que, por exemplo, pode ser usado, na iteração de uma tabela).
- Permitir a definição de expressões booleanas (predicados) contendo, pelo menos relações de ordem e operadores booleanos (conjunção, disjunção, etc.).
- Dentro duma tabela, permitir a definição de colunas através de fórmulas envolvendo operações aritméticas (e eventualmente booleanas) operando sobre outras colunas (identificadas pelo nome). Nestas colunas, todas as linhas estão automaticamente preenchidas com o resultado da aplicação da fórmula (ao contrário do EXCEL, onde é necessário copiar a fórmula para cada célula).
- Generalizar a linguagem secundária por forma a permitir a leitura da funcionalidade anterior (isto é ter uma representação textual para colunas definidas por fórmulas).
- Permitir a extracção de uma tabela a partir de outra usando predicados.
- Permitir a definição de uma tabela aplicando expressões aritméticas a colunas de outras tabelas.
- Incluir a instrução condicional (operando sobre expressões booleanas).
- Incluir uma operação booleana de existência sobre o conteúdo de uma tabela. Esta operação deve poder aplicar um predicado sobre os valores da tabela (por exemplo, verificar se existe pelo menos um valor numa coluna que façam com que um predicado seja verdadeiro).
- Generalização da operação de junção de tabelas por forma a permitir a junção de tabelas usando colunas como critério de junção (do género da operação `join` do SQL).

## Características opcionais

- Fazer com que a linguagem contenha um sistema de tipos. Devem existir os seguintes tipos de dados: tabela, número real, número inteiro, texto, booleano, tuplo.
- Implementar funções.
- Permitir a definição de tabelas dependentes de tabelas existentes (como em bases de dados). Os valores destas tabelas estariam sempre dependentes dos valores das tabelas base (isto é, quaisquer alterações nas tabelas base, seriam automaticamente propagadas às tabelas dependentes).

- Acrescentar propriedades a campos (colunas) de tabelas. Estas propriedades funcionariam como invariantes para valores das tabelas (por exemplo, para impor limites nos valores de uma coluna, garantindo que uma coluna de percentagens não tem valores fora do intervalo  $[0 : 100]$ ).
- Permitir operações com funcionalidade que aproximem a linguagem das funcionalidades existentes em linguagens de bases de dados (tipo SQL).

## Desafios

Consegue, usando a linguagem que criou, implementar uma base de dados para uma colecção de álbuns de música ou livros numa biblioteca?