

”

**E-fólio A** | Folha de resolução para E-fólio

**UNIDADE CURRICULAR:** Linguagens de Programação

**CÓDIGO:** 21077

**DOCENTE:** Ricardo Batista

**A preencher pelo estudante**

**NOME:** Hernâni Filipe Resendes Coelho

**N.º DE ESTUDANTE:** 1800045

**CURSO:** Engenharia Informática

**DATA DE ENTREGA:** 24/04/2022

## **TRABALHO / RESOLUÇÃO:**

Este trabalho foi desenvolvido utilizando a linguagem funcional OCAML, e tem como objetivo apresentar ao utilizador uma interface que disponibiliza um conjunto de receitas culinárias. Ao utilizador estará disponível duas funcionalidades: 1- Consultar detalhes de uma receita; 2- Consultar a escolha diária do chefe em número de três receitas. As receitas disponíveis são carregadas para o programa a partir de um ficheiro, disponibilizado no formato .csv.

A organização da estrutura de ficheiros do programa está organizada de acordo com as indicações, a partir do diretório base:

- efolioA/
  - receitas.ml (código fonte)
  - receitas.cmi (ficheiro de compilação)
  - receitas.cmx (ficheiro de compilação)
  - receitas.o (ficheiro de compilação)
  - receitas (executável)
  - readme.txt (instruções de compilação e execução)
  - dados/
    - receitasV1.csv (dados de entrada)

### **Organização do código:**

O código está organizado por funções, de modo a permitir uma melhor legibilidade. As principais escolhas que tomei para implementação do programa foram:

- Definição de uma estrutura de dados (record) para armazenamento da informação de cada receita;
- Leitura do ficheiro e posterior preenchimento dos registos;
- Adição de cada registo preenchido a uma lista do seu tipo;

O programa inicia a execução na função “main” passando de seguida à função “menu” onde primeiramente é executado a leitura do ficheiro, carregado para a memória, através de uma lista de registos a informação disponível. A função

“lerFicheiro” verifica se o ficheiro existe e só de seguida o tenta ler, abrindo o canal de leitura e lê de linha a linha através da função recursiva “lerLinha”. Esta função faz um match pattern onde caso exista algo processa a linha, quebrando através do delimitador a informação e adiciona a mesma num registo que posteriormente é adicionado à lista de registos. A recursão é executada até que seja atingido o fim do ficheiro (caso base da recursão) onde é fechado o canal de leitura e procede-se à inversão da lista, já completa, para que fique na ordem correta. Após isto, temos então o ficheiro processado e a informação toda separada de cada receita por atributos.

De seguida, através de um ciclo com variável controlo é apresentado ao utilizador a interface do menu, apresentando as receitas e as opções disponíveis. A opção 1, permite que o utilizador consulte a informação completa de uma receita. O utilizador indica qual das receitas pretende ver (através do nome ou número) sendo esta indicação guardada para passar posteriormente à função “obterRegisto” a qual executa uma procura na lista de registos retornando o registo correspondente. Após isto, a informação da receita é então apresentada na tela. Na opção 2, o utilizador indica quais as receitas (através do nome ou número) que pretende que sejam compiladas para serem a escolha diária do chefe. Esta informação é posteriormente processada na função “compilarRegistos” que retorna uma lista de registos contendo apenas as receitas indicadas anteriormente pelo utilizador, permitindo que posteriormente se possa proceder à compilação conjunta das receitas. Como não tive mais disponibilidade, pelo facto de ter mais efólios em paralelo, não consegui finalizar a funcionalidade 2 tal como queria. Pelo que, o que apresento não vá totalmente ao encontro do que se pede, imprimir a compilação das quantidades dos ingredientes das três receitas. A minha ideia para implementação completa da funcionalidade, passaria por imprimir os ingredientes diferentes nas três receitas e os que fossem iguais, seriam processados. Isto é, no campo das quantidades, iria quebrar a string pelo delimitador e proceder à identificação da quantidade numérica na string utilizando uma expressão regex. Identificada a quantidade, seria então convertido para valor numérico para proceder à soma. Posteriormente seria novamente concatenado e imprimido para a tela. Deste modo teríamos a agregação de ingredientes iguais nas receitas escolhidas.

Em relação à apresentação na tela dos resultados, escolhi apresentar a informação num formato de tabela, no entanto devido às palavras com caracteres acentuados, não consegui em tempo útil solucionar o problema da formatação da largura das colunas, pois na presença deste tipo de caracteres a há um desalinhamento da largura.