

”

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.

Exemplo de testes de execução:

```
-----
1: Arroz ao forno
-----
2: Arroz maggi
-----
3: Tutu de feijão
-----
4: Omelete expressa
-----
5: Omelete de forno
-----
6: Baked potatoes ao requeijão
-----
7: Purê de batata
-----
8: Macarrão à fiorentina
-----
9: Macarrão ao alho e óleo
-----
10: Frango com laranja
-----
11: Frango ensopado
-----
12: Bife de marinheiro
-----
13: Bife grelhado
-----
14: Creme de Maracujá
-----
15: Mousse de chocolate
-----
Estas são as receitas disponíveis de momento!

Indique a opção que pretende:
1 - Consultar uma receita
2 - Escolha diária do chef em número de três receitas
0 - Sair
```

Imagem 1: Interface inicial do programa

```
Indique o número da receita que pretende visualizar: 7
-----
Receita 7: Purê de batata
-----
Ingredientes                               Quantidade
-----
1: batata                                  0.5 kg
-----
2: manteiga                               2 colheres (sopa)
-----
3: leite                                  0.5 xicara (chá)
-----
4: fondor                                  0.5 colher
-----
5:
-----
6:
-----
7:
-----
8:
-----
9:
-----
10:
-----
11:
-----

Modo de Preparação:
Descasque as batatas e cozinhe-as em água fervente até que estejam macias. Escorra-as e passe-as pelo espremedor enquanto estiverem quentes. Coloque a batata espremida em uma panela, junte a manteiga e o leite bem quente e bata vigorosamente com uma colher, até deixar a mistura bem lisa e cremosa. Tempere com o Fondor Maggi e continue mexendo em fogo baixo por mais 5 minutos.

Pressione uma tecla para Continuar
```

Imagem 2: Exemplo do output de uma receita