

# Trabalho Prático 2

Paradigmas da Programação 1 (LESI) / Programação Funcional (LMCC)

2005/2006

## Introdução

**Constituição dos grupos de trabalho** Os grupos deverão ser formados por três elementos. Preferencialmente deverão ser mantidos os mesmos grupos formados para o primeiro trabalho prático. Idealmente os elementos de um grupo deverão frequentar o mesmo turno teórico-prático.

**Data de entrega** O trabalho será entregue durante as aulas da semana de 12 de Dezembro a 16 de Dezembro.

**CrITÉrios de avaliação** Na avaliação dos trabalhos serão valorizadas os seguintes elementos:

- Definições de tipos de dados.
- Utilização adequada de classes.
- Aplicação criteriosa de funções de ordem superior.
- Estruturação do código em módulos.
- Compilação e execução correcta na linha de comando.

## Enunciado

Pretende-se a implementação de um sistema para a gestão operacional de um restaurante, que forneça as seguintes funcionalidades:

- O menu do restaurante é gerido como uma lista de pratos/bebidas e os respectivos preços.
- Deverá ser possível adicionar e remover elementos do menu.
- O stock do restaurante é gerido como uma lista de ingredientes que regista, para cada um deles, uma determinada quantidade disponível.
- Deverá ser possível adicionar e remover quantidades de ingredientes do stock.
- O restaurante tem um número pré-definido de mesas. Para cada mesa deve ser possível:
  - Abrir uma nova conta.
  - Adicionar um elemento do menu à conta.
  - Anular um elemento da conta.
  - Fechar a conta, imprimindo uma factura.
- A aplicação deverá assegurar a coerência entre as vendas e o stock:
  - Deverá manter uma relação entre os elementos do menu e as quantidades de ingredientes necessárias à sua manufactura.

- A venda de um prato ou bebida deverá implicar a dedução das quantidades adequadas do stock, e apenas deverá ser possível se o stock o permitir.
- O cancelamento de uma venda deverá permitir, opcionalmente, voltar a adicionar as quantidades adequadas ao stock (caso não tenham sido inutilizadas).

Notas de implementação:

- A implementação em Haskell de um sistema com estas características passa por definir e manipular valores de um tipo de dados complexo que reflecta uma visão global de toda a informação (não estática) gerida pela aplicação.
- Este tipo de dados terá de incluir os dados relativos a menu (possivelmente com os ingredientes de cada alimento), stock e mesas.
- O *estado* da aplicação em cada instante será mantido num único valor deste tipo que será actualizado ao longo do tempo.
- Por outro lado, a tabela de informação nutricional que armazena os dados relativos a cada ingrediente poderá ser implementada como informação estática, por exemplo definida como uma constante.

O programa a desenvolver deverá contemplar os seguintes componentes:

1. Implemente uma função que lê o estado inicial da aplicação a partir de um ficheiro.
2. Implemente uma função que permita guardar o estado actual da aplicação num ficheiro.
3. Implemente uma interface interactiva com o utilizador.
4. Integre as funções que desenvolveu para o primeiro trabalho prático na sua aplicação, permitindo fornecer ao cliente:
  - (a) uma estimativa das calorias que ingeriu em cada alimento, e no total da sua refeição.
  - (b) uma versão alternativa do menu que contenha apenas os alimentos com baixo teor de lípidos.
5. Compile o programa Haskell que escreveu.