

# Projecto 1 de Laboratórios de Informática 3 2009/10

## Linguagem C

### 1 Introdução

Este documento apresenta as linhas condutoras do enunciado do primeiro projecto de LI3. O projecto deve ser efectuado na linguagem C (com recurso ao `gcc`) e utiliza os conhecimentos adquiridos nas Unidades Curriculares de Programação Imperativa e Algoritmos e Complexidade. Tem como objectivos que os alunos exercitem as boas práticas e técnicas que adquiriram nessas UC's num caso real.

Valoriza-se na avaliação do trabalho que a construção do projecto obedeça a alguns princípios importantes, tais como:

- a adequação das estruturas de dados utilizadas à representação da informação que o problema aborda;
- a reutilização de algoritmos e definições aprendidas nas UC's de PI e AC;
- a escrita de código genérico, modular e reutilizável, que possa ser utilizado noutros projectos futuros;
- a utilização de mecanismos de serialização dos dados, com o intuito de salvaguarda do estado das aplicações.

O enunciado do projecto aborda um conjunto de requisitos, cuja concretização em C obriga à instanciação de várias estruturas de dados que foram estudadas previamente. Cada um dos grupos deve escolher as estruturas que lhe parecem melhores e justificar no relatório, a apresentar na defesa do projecto, as razões e vantagens das escolhas efectuadas.

Como bibliografia sugerida, recomenda-se a consulta às páginas web das disciplinas de Programação Imperativa e Algoritmos e Complexidade.

## 2 A Rede Social *Sempre Ligados*

As redes sociais são um fenómeno sociológico e tecnológico. De uma forma muito simples podemos dizer que cada utilizador possui um perfil próprio, com a instanciação de vários atributos, e identifica também um conjunto de outros utilizadores com os quais está de alguma forma relacionado. Existem várias redes sociais que são exemplos do que se acaba de descrever, como por exemplo, o Facebook, o Twitter, Orkut, o LinkedIn, entre outros. No caso vertente do Linked In os utilizadores inscrevem-se na rede com o objectivo de criarem uma teia de relacionamentos de índole profissional. Muitos dos utilizadores procuram nesta rede novas oportunidades a nível profissional ao fazerem parte de uma teia relacionada pelos mesmos interesses. Uma das actividades mais frequentes no Linked In consiste na procura de contactos que permitam ajudar a promover a carreira profissional de um utilizador. Tipicamente o objectivo é contactar alguém e utilizar a rede social para conseguir alguém que nos apresente (ou recomende).

Assuma que pretendemos criar a nossa rede social, baseada no Linked In, que se designará por *Sempre Ligados*. A proposta de valor da *Sempre Ligados* assenta nas seguintes premissas:

- as relações entre os utilizadores podem ser de três tipos: *Profissional*, *Família* e *Amigos*
- é possível adicionar novos utilizadores e registar a quem é que estão ligados de acordo com os diferentes tipos de relações
- é objectivo dos promotores da rede social que esta possa atingir pelo menos os 10000 utilizadores, sendo que o programa deve estar preparado para tal.

Cada utilizador conhece os utilizadores com que está directamente relacionado. Na altura de criar uma ligação a outro utilizador é necessário escolher o tipo de relacionamento (Profissional, Família e Amigos) e atribuir um peso à distância entre os dois utilizadores. Regra geral a distância entre dois utilizadores é de 1, embora nada impeça que o custo da relação seja superior. Imagine-se que o utilizador  $X$  conhece o utilizador  $Y$ , mas a relação de confiança entre ambos não é significativa. Nesse caso o custo pode ser superior a 1, na medida em que utilizar  $Y$ , para determinado propósito, implica que este está a prestar, de forma algo renitente, um *favor*.

Para cada utilizador é guardada a seguinte informação: nome, cidade onde vive, estado civil, número de contribuinte, email.

As tarefas de manutenção da estrutura de dados da rede social (tipicamente um grafo) que contém os utilizadores (nós) e respectivas ligações (arestas), incluem as seguintes funcionalidades:

- inserção de um utilizador;
- remoção de um utilizador;
- inserir ligação tipada entre dois utilizadores;
- remover ligação entre dois utilizadores <sup>1</sup>;
- alterar as propriedades da ligação entre dois utilizadores.

Uma vez explicada a estrutura de dados que a rede social utiliza para gerir os diversos utilizadores e respectivas ligações, importa detalhar as funcionalidades disponibilizadas pela rede social:

- determinar todos os utilizadores a uma distância  $d$  de um outro utilizador
- determinar todos os utilizadores a uma distância  $d$  de um outro utilizador, de acordo com um determinado tipo de relação
- caminho mais curto entre dois utilizadores, de acordo com um tipo de relação.
- determinar qual é a distância máxima entre quaisquer dois utilizadores da rede social. Importa também dizer quem são os utilizadores que estão mais afastados entre si.

Por forma a facilitar a utilização do programa por parte dos utilizadores, deve ser possível pesquisar a base de dados de utilizadores através do nome e através do número de contribuinte. A definição da estrutura de dados deve prever esta necessidade.

Os utilizadores comunicam entre si através do envio de mensagens directamente para o perfil do utilizador destino. Cada utilizador guarda as mensagens indexadas pelo nome do utilizador que as originou. O programa guarda as últimas 20 mensagens recebidas de cada utilizador. A rede social tem na sua página web a indicação de quem é o utilizador mais popular, isto é, o utilizador que recebe mais mensagens.

---

<sup>1</sup>Por uma questão de eficiência na manipulação da estrutura de dados podem optar por marcar como "apagada", em vez de remover fisicamente.

## 2.1 Salvaguarda do estado da aplicação

O programa deve permitir que em qualquer momento se possa guardar em ficheiro a informação existente em memória sobre os utilizadores, ligações entre eles e mensagens trocadas. A gravação deve ser feita da forma mais optimizada possível e deve permitir que o estado seja recuperado novamente.

## 3 Relatório

O relatório a entregar deve permitir esclarecer:

- a razão pela qual se escolheu determinada estrutura de dados;
- a capacidade de modularidade que a solução encontrada apresenta;
- qual a complexidade (em termos de ocupação de memória) das representações escolhidas e a complexidade (em termos de tempo) das funções implementadas.

## 4 Cronograma

A entrega do projecto far-se-á de forma faseada, nas seguintes *milestones*:

1. entrega do código com as estruturas de dados e respectivas operações de manutenção.

**Data Limite:** 26 Março

2. entrega do código da interface com o utilizador e um relatório (manual de utilização) com os diversos *screenshots*.

**Data Limite:** 9 Abril

3. entrega final de código, executável e relatório de projecto.

**Data Limite:** 23 Abril

As cotações associadas às determinados milestones são:

- milestone 1: 4 valores
- milestone 2: 2 valores
- milestone 3: 14 valores (relatório: 2 valores)

Devem aproveitar as aulas TP e os períodos de atendimento para esclarecimento de dúvidas.