

Trabalho 1 - *FACINBook*

Laboratório de Programação II
Profs. Márcio Sarroglia Pinho e Edson Ifarraguirre Moreno

04/2017

1 Introdução



Você foi contratado para fazer um software revolucionário: um gerenciador de uma rede de relacionamentos, algo nunca visto. O objetivo é explorar os conhecimentos vistos até o momento sobre a linguagem C++ e orientação a objetos, bem como estruturas de dados encadeadas (listas).

2 Funcionamento

Ao ser iniciado, o programa deve carregar os arquivos que descrevem os participantes, os interesses/atividades e a rede social.

2.1 Arquivo de participantes

O arquivo de participantes (**contatos.txt**) possui o seguinte formato:

```
Chuck Norris  
Jackie Chan  
Bruce Lee  
Tony Jaa  
Sammo Hung
```

Ou seja, uma linha contendo o nome de cada participante.

2.2 Arquivo de interesses/atividades

O arquivo de interesses/atividades (**atividades.txt**) possui o seguinte formato:

```
Cinema  
Literatura  
Pintura  
Jeet Kune Do  
Karate  
Muhay Thay  
Tang Soo Do  
Wing Chun  
Wushu
```

Ou seja, uma linha contendo o nome de cada interesse/atividade (no exemplo, uma combinação saudável de artes... e artes marciais).

2.3 Arquivo da rede social

O arquivo que descreve a rede social (**rede.txt**) possui o seguinte formato:

- Cada pessoa possui um nome (primeira linha) e um ou mais interesses.

- A seguir, segue uma lista com os contatos da pessoa (as linhas que começam com --).
- As pessoas são separadas por linhas que começam com #.

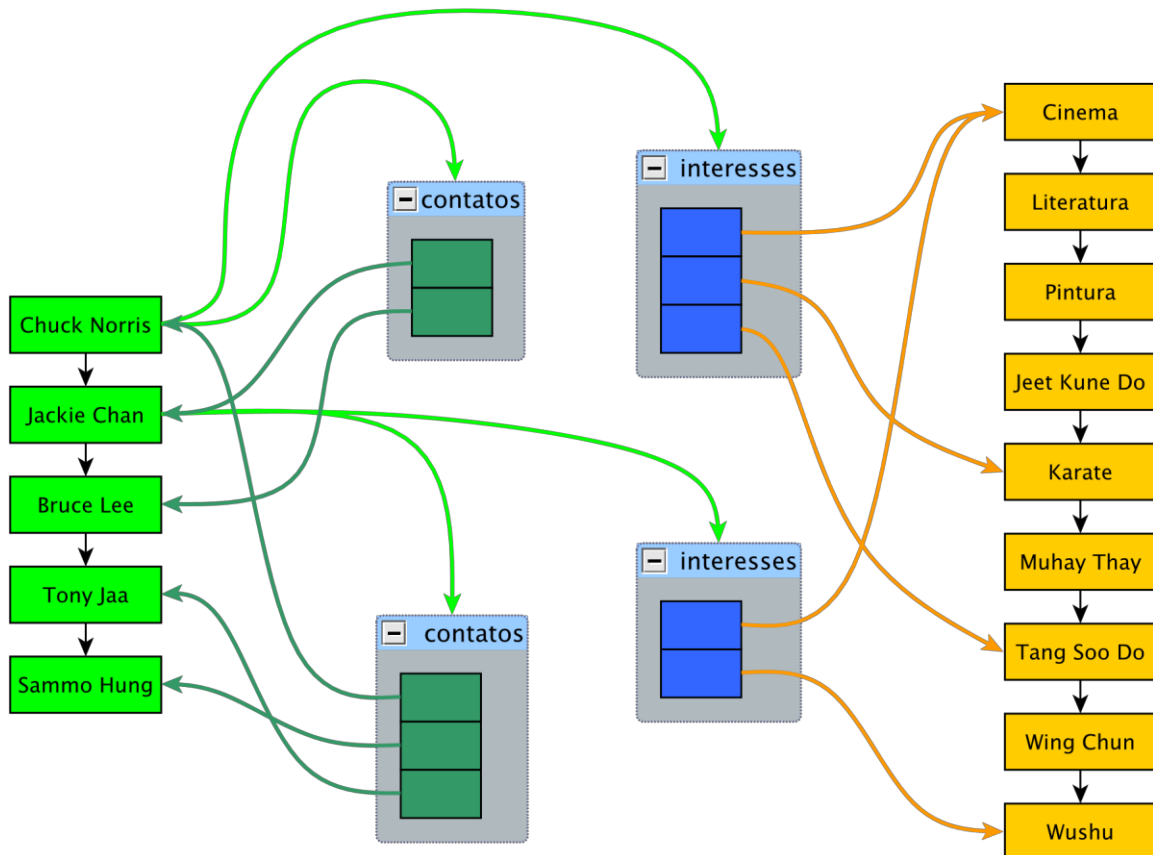
```

Chuck Norris
Tang Soo Do
Karate
Cinema
-- Jackie Chan
-- Bruce Lee
#
Jackie Chan
Wushu
Cinema
-- Chuck Norris
-- Sammo Hung
-- Tony Jaa
#
Bruce Lee
Wing Chun
Jeet Kune Do
Cinema
-- Sammo Hung --
Tony Jaa -- ... #

```

2.4 Estrutura de Dados

A partir da leitura dos arquivos, deve ser montada uma estrutura de dados encadeada em memória, como apresenta a figura abaixo:



Neste exemplo, estão destacadas as duas listas principais do sistema: **participantes** (em verde) e **interesses/atividades** (em laranja). Cada participante, por sua vez, possui também uma lista de contatos e uma lista de interesses (no exemplo, apenas de Chuck e Jackie). Essas duas listas devem armazenar ponteiros para os objetos correspondentes das listas principais. Por exemplo, Chuck tem interesse em Cinema, Karate e Tang Soo Do, e tem como contatos Jackie e Bruce Lee. Jackie, por sua vez, também tem interesse em Cinema, e Wushu.

2.5 Menu de Opções

Uma vez carregado o arquivo, o programa oferece as seguintes opções:

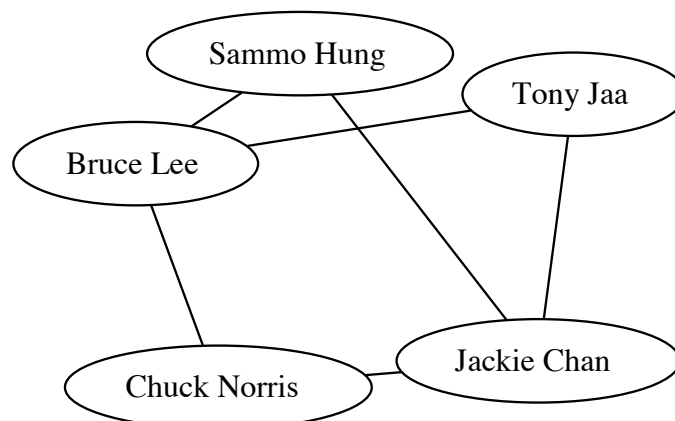
- *Login*: exibe a lista de usuários e permite escolher um (**não é necessário implementar controle de senha**, até porque a senha de Chuck Norris não é armazenável em um computador).
- *Logout*: retorna para a lista de usuários. A lista de usuários é **fixa**, não sendo possível nem incluir nem remover usuários do sistema.
- Contatos:
 - Listar: mostra os contatos do usuário logado, bem como os seus interesses/atividades.
 - Incluir: permite a inclusão de um novo contato, dentre os presentes no sistema.
 - Excluir: permite a exclusão de um contato existente na lista do usuário.
- Interesses/Atividades:
 - Listar: mostra os interesses/atividades do usuário logado.
 - Incluir: permite a inclusão de um novo interesse/atividade no usuário logado, dentre os disponíveis na lista geral.
 - Excluir: permite a exclusão de um interesse/atividade do usuário logado (**obs: não é possível remover interesses da lista geral**).
- Exportar arquivo de dados para o formato de arquivo do *graphviz* (veja abaixo).
- Finalizar: salva o arquivo *rede.txt*, e sai do programa.

A opção de exportação deve gerar um arquivo de saída denominado **rede.dot**, com o seguinte formato:

```
graph
{ overlap=false
  "Chuck Norris" -- "Jackie Chan"
  "Chuck Norris" -- "Bruce Lee"
  "Jackie Chan" -- "Sammo Hung"
  "Jackie Chan" -- "Tony Jaa"
  "Bruce Lee"      -- "Sammo Hung"
  "Bruce Lee"      -- "Tony Jaa"
}
```

Observe que não deve haver duplas ligações. Por exemplo, Jackie Chan é contato de Chuck Norris e vice-versa, e só há uma linha indicando essa ligação - apesar de haver a indicação da ligação de volta no arquivo de entrada.

Este arquivo pode ser processado pelo software *graphviz* (<http://www.graphviz.org>) que é capaz de interpretar os comandos escritos na linguagem *DOT* e produzir uma figura com a representação da rede (tecnicamente, um *grafo*):



Outra forma simples de visualizar o arquivo é através do seguinte site:
<http://sandbox.kidstrythisathome.com/erdos/index.html>

3 Requisitos

- Devem ser utilizados os conceitos vistos até agora de orientação a objetos: classes e objetos, modularização de código.
- Para os dados, deve ser utilizada uma estrutura encadeada baseada em listas. Por exemplo, pode ser empregada uma lista encadeada para as pessoas no sistema. Cada pessoa, por sua vez, possui uma lista de contatos e uma lista de atividades. **É obrigatório o uso de ponteiros.**

4 Avaliação

Leia com atenção os critérios de avaliação:

- Os trabalhos são **em duplas**. Os arquivos contendo o código-fonte (.cpp) devem ser compactados e submetidos pelo *Moodle* até a data e hora especificadas. O arquivo compactado deve ter os últimos sobrenomes de ambos os alunos, da seguinte forma: *sobrenome1_sobrenome2.zip*.
- A implementação deve seguir as orientações dadas em aula quanto a convenções C++ para nomes de identificadores e estrutura da classe (ou seja, nomes de classe começam com letra maiúscula, atributos, métodos e variáveis em minúsculas). O código deve estar indentado corretamente.
- Data de entrega no *Moodle* e apresentação: 17/05/2017 (impreterivelmente, não haverá adiamentos).
- A nota do trabalho depende da apresentação deste no laboratório, na data marcada. Trabalhos entregues mas não apresentados terão sua nota anulada automaticamente. Durante a apresentação será avaliado o domínio da resolução do problema, podendo inclusive ser possível invalidar o trabalho quando constatada a falta de conhecimento sobre o código implementado.
- **A cópia parcial ou completa do trabalho terá como consequência a atribuição de nota ZERO ao trabalho dos alunos envolvidos. A verificação de cópias é feita inclusive entre turmas.**