

Trabalho de Estruturas de Dados II

Rede social de "amizade verdadeira"

Anderson Borges
9045195

Marina Yoza
10746964

Stéfane Tame
10829970

28 de junho de 2020

1 Introdução

Neste trabalho foi desenvolvido um sistema capaz que armazenar informações de usuários numa espécie de rede social onde fosse possível verificar graus de compatibilidade entre eles, isto é, se há similaridades em relação a preferências, idade e local onde moram. A partir desse "cálculo de compatibilidade", foram implementadas funções auxiliares que sugerissem amizades, namoros e também informasse quais amizades fossem menos compatíveis.

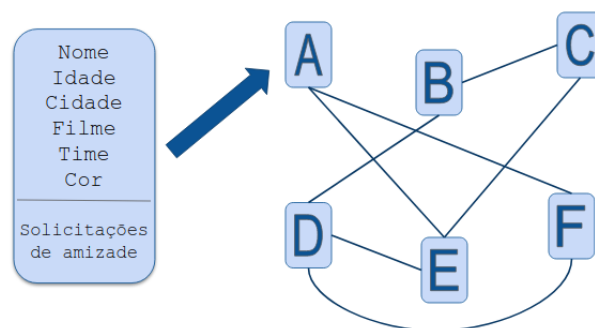


Figura 1: Ilustração do grafo implementado

2 Implementação e demonstração de execução do sistema

O sistema proposto foi implementado a partir de um arquivo com os perfis de usuários da rede social e, assim, foi gerado um grafo a partir destas informações. Optamos pela implementação de um grafo com lista de adjacências, pela praticidade de manuseio e para evitar a matriz esparsa. Além disso, foram incluídas no código as funções básicas de grafo, que foram: criar um grafo, verificação se existe aresta, inserir uma aresta, remover aresta, verificação se a lista de adjacências está vazia, conseguir o primeiro nó e o próximo da lista de adjacências e, por fim, liberar memória utilizada para o grafo.

Inicialmente, o programa lê o arquivo de texto (*dados.txt*) com dados dos perfis dos usuários da rede social. Cada perfil é salvo num vértice do grafo, onde futuramente as arestas representarão as amizades entre usuários. Veja na Figura 1, onde o usuário A é amigo de E e F, C é amigo de B e E, e assim por diante.

2.1 Função de *Log in*

Ao executar o programa, o usuário se depara com a opção de acessar a rede social com seu perfil já existente, ou se necessário, criar um novo perfil, a ser inserido no arquivo de dados e no grafo. Esta segunda opção é uma função extra implementada. Veja o resultado nas Figuras 2a e 2b.

```
❯ clang-7 -pthread -lm -o main main.c tad.c
❯ ./main
Digite seu nome para acesso a seus dados
Guilherme Sousa
Este nome nao foi encontrado.
Deseja criar um perfil?
1.sim 2.nao
1
Digite sua idade: 20
Digite sua cidade: Governador Valadares
Digite seu filme predileto: As Branqueelas
Digite seu time predileto: Nenhum
Digite sua cor predileta: Rosa
```

(a) Novo perfil

```
Caro(a) Guilherme Sousa, suas opcoes sao:
1) Fazer um convite de amizade
2) Ver solicitacoes de amizade para aceitar ou recusar
3) Ver sugestoes de amigos verdadeiros
4) Consultar amigos que nao sao amigos verdadeiros
5) Encontrar namorado ideal
6) Ver litas de amigos
7) Trocar de perfil
8) Encerrar programa

O que deseja fazer?
Sua resposta: 
```

(b) Menu

Figura 2: Execução da função de *Log in*

2.2 Função de fazer convite de amizade

Os usuários têm a opção de estabelecer relações de amizade com outros usuários, para isso, um deles envia a solicitação de amizade e o outro deve aceitar ou não este convite, como no *Facebook*, por exemplo.

A implementação dessa função se deu a partir da inclusão de uma lista de convites em cada "*nó vértice*", ou seja, cada vértice do grafo armazena, além dos dados do perfil que ele representa, os convites que amizade que este usuário ainda não respondeu.

Nesta função, o usuário "*logado*" dá o nome da pessoa de interesse e o programa verifica se este nome está em algum perfil, se não, o usuário pode tentar outro nome. Caso o nome seja encontrado, é verificado se os dois usuários já são amigos, através da função *existe_aresta*, e se este convite de amizade já foi feito (por ambas as partes). Então, se realmente for o caso de criar uma nova solicitação, a função a insere na lista de solicitações do perfil buscado.

```

Caro(a) Guilherme Sousa, suas opcoes sao:
1) Fazer um convite de amizade
2) Ver solicitacoes de amizade para aceitar ou recusar
3) Ver sugestoes de amigos verdadeiros
4) Consultar amigos que nao sao amigos verdadeiros
5) Encontrar namorado ideal
6) Ver litas de amigos
7) Trocar de perfil
8) Encerrar programa

O que deseja fazer?
Sua resposta: 1

Digite o nome da pessoa que deseja mandar um convite de amizade
: Marina Yoza
Solicitacao enviada!

```

(a) Sugestão enviada

```

Caro(a) Guilherme Sousa, suas opcoes sao:
1) Fazer um convite de amizade
2) Ver solicitacoes de amizade para aceitar ou recusar
3) Ver sugestoes de amigos verdadeiros
4) Consultar amigos que nao sao amigos verdadeiros
5) Encontrar namorado ideal
6) Ver litas de amigos
7) Trocar de perfil
8) Encerrar programa

O que deseja fazer?
Sua resposta: 1

Digite o nome da pessoa que deseja mandar um convite de amizade
: Alex Lopes
Este nome nao foi encontrado. Gostaria de tentar outro nome?
1)Sim
2)Nao

```

(b) Sugestão não enviada

Figura 3: Função para fazer convite de amizade

2.3 Função de verificação de solicitações

Esta função é mais simples que a anterior. Como cada usuário tem uma lista de solicitações, bastou implementar uma função que percorresse esta lista, dando ao usuário a opção de aceitar ou não a solicitação de amizade, exibindo antes o grau de compatibilidade entre eles, calculado através da função descrita na Seção 2.4. Caso o convite seja aceito, a função base *inserir_aresta* faz a ligação entre os dois vértices, estabelecendo a amizade entre os dois perfis.

```

Caro(a) Marina Yoza, suas opcoes sao:
1) Fazer um convite de amizade
2) Ver solicitacoes de amizade para aceitar ou recusar
3) Ver sugestoes de amigos verdadeiros
4) Consultar amigos que nao sao amigos verdadeiros
5) Encontrar namorado ideal
6) Ver litas de amigos
7) Trocar de perfil
8) Encerrar programa

O que deseja fazer?
Sua resposta: 2
A pessoa Guilherme Sousa te mandou um convite de amizade
Chance de compatibilidade: 36%
Deseja aceitar? (sim/nao): sim

```

Figura 4: Função para responder solicitações de amizade

2.4 Função de sugestão de amizades

Para atender a proposta geral deste trabalho, foi criada uma função auxiliar para fazer o cálculo de compatibilidade entre dois perfis. Considerando que a análise psicológica não é nossa área de competência, optamos pelo cálculo de grau de semelhança entre usuários, ou seja, dois usuários 100% compatíveis teriam todos os cinco dados idênticos e teriam mesmo sobrenome. Estabelecemos os seguintes pesos: 2 para idade, filme predileto, time e cidade, e 1 para cor favorita e sobrenome. Assim, duas pessoas da mesma cidade teriam pelo menos 20% compatibilidade.

No caso da idade, consideramos que pessoas de idades próximas também seriam compatíveis. Então, estipulamos que o acréscimo c de compatibilidade é da forma $c = 10 - 2|i_a - i_b|$, onde i_a e i_b são as idades dos dois usuários em questão. Esta escolha se baseou na ideia de que quanto mais próximas as idades, mais compatíveis são os perfis, estabelecendo um valor máximo de diferença, que no caso foi 4.

Desse modo, a partir do cálculo de compatibilidade, a função *sugestoes* percorre o grafo (os vértices dele) e retorna ao usuário uma lista dos perfis com maior compatibilidade, no caso, todos os que apresentarem o maior valor encontrado. Veja o exemplo na Figura 5.

```

Caro(a) Stefane Tame, suas opcoes sao:
1) Fazer um convite de amizade
2) Ver solicitacoes de amizade para aceitar ou recusar
3) Ver sugestoes de amigos verdadeiros
4) Consultar amigos que nao sao amigos verdadeiros
5) Encontrar namorade ideal
6) Ver litas de amigos
7) Trocar de perfil
8) Encerrar programa

O que deseja fazer?
Sua resposta: 3
Sugestoes de amizades verdadeiras:
Sofia Tame, com 50% de compatibilidade
Debora Tame, com 50% de compatibilidade
Maria Tame, com 50% de compatibilidade

```

Figura 5: Função faz sugestões de amizade para Stéfane Tame

```

Caro(a) Guilherme Sousa, suas opcoes sao:
1) Fazer um convite de amizade
2) Ver solicitacoes de amizade para aceitar ou recusar
3) Ver sugestoes de amigos verdadeiros
4) Consultar amigos que nao sao amigos verdadeiros
5) Encontrar namorade ideal
6) Ver litas de amigos
7) Trocar de perfil
8) Encerrar programa

O que deseja fazer?
Sua resposta: 4
Lista de amigos que nao sao amigos verdadeiros:
Stefane Tame, com 16% de compatibilidade

Deseja excluir sua amizade com Stefane Tame?
1.sim 2.nao

```

Figura 6: Função aponta amigo menos compatível de Guilherme Sousa

2.5 Função que encontra amigos incompatíveis

Novamente usando a função auxiliar de cálculo de compatibilidade, esta opção percorre a lista de adjacências do usuário e verifica a cada amigo o grau de compatibilidade. Ao fim, mostra para o usuário a lista de amigos que são os menos compatíveis e dá a opção de remover a amizade ou não. Veja o exemplo na Figura 6.

2.6 Função que sugere perfil para namoro

É evidente a importância da função de cálculo de compatibilidade para a rede social criada, especialmente porque ela está diretamente ligada à proposta inicial do sistema. Neste caso, o usuário tem a opção de buscar um perfil que possivelmente seria ideal para namoro, novamente considerando que compatibilidade vem da semelhança de perfil.

Como na função descrita na Seção 2.4, nesta função o grafo é percorrido, calculando a compatibilidade entre o usuário "logado" e os demais perfis. Dessa vez, entretanto, não é apresentada ao usuário uma lista de possíveis parceiros, mas aquele que teve maior compatibilidade, desconsiderando aqueles que têm mesmo sobrenome, uma vez que isso pode indicar pertencimento à mesma família. Note na Figura 7 que o namorado sugerido não está na lista de amigos sugeridos, pois todos os mais compatíveis com o perfil de Stéfane Tame eram também da família Tame.

```

Caro(a) Stefane Tame, suas opcoes sao:
1) Fazer um convite de amizade
2) Ver solicitacoes de amizade para aceitar ou recusar
3) Ver sugestoes de amigos verdadeiros
4) Consultar amigos que nao sao amigos verdadeiros
5) Encontrar namorado ideal
6) Ver litas de amigos
7) Trocar de perfil
8) Encerrar programa

O que deseja fazer?
Sua resposta: 5
Namorado ideal eh Davi Braz, com 36% de compatibilidade

```

Figura 7: Função sugere namorado

```

Caro(a) Guilherme Sousa, suas opcoes sao:
1) Fazer um convite de amizade
2) Ver solicitacoes de amizade para aceitar ou recusar
3) Ver sugestoes de amigos verdadeiros
4) Consultar amigos que nao sao amigos verdadeiros
5) Encontrar namorado ideal
6) Ver litas de amigos
7) Trocar de perfil
8) Encerrar programa

O que deseja fazer?
Sua resposta: 6
Lista de amigos:
Marina Yoza
Stefane Tame
Alex Lopes

```

(a) Lista de amigos de Stéfane Tame

```

Caro(a) Joaquim Coelho, suas opcoes sao:
1) Fazer um convite de amizade
2) Ver solicitacoes de amizade para aceitar ou recusar
3) Ver sugestoes de amigos verdadeiros
4) Consultar amigos que nao sao amigos verdadeiros
5) Encontrar namorado ideal
6) Ver litas de amigos
7) Trocar de perfil
8) Encerrar programa

O que deseja fazer?
Sua resposta: 6
Nao ha amigos na sua lista

```

(b) Caso o usuário não tenha amigos

Figura 8: Função imprime lista de amigos do usuário *logado*

2.7 Função que exhibe a lista de amigos

Uma função extra implementada neste trabalho é a *lista_amigos*, que imprime a lista de amigos do usuário "*logado*", veja nas Figuras 8a e 8b. É uma ideia simples, mas para sua execução foi necessária a criação de mais duas funções auxiliares, uma para verificar se o usuário não tem amigos e outra para acessar o primeiro usuário na lista de adjacências do perfil "*logado*". Assim, caso o usuário não tenha amigos, ele recebe uma mensagem o informando disso. Caso contrário, o programa imprime a lista de vértices (nomes dos usuários) que têm aresta com o vértice em questão.

3 Compilador usado e como executar o sistema

Este trabalho foi feito no *repl.it*, plataforma que possibilita o acesso ao código por diferentes máquinas, além permitir edição simultânea sem necessidade de versionamento.

Para utilizar o programa, os dois arquivos *.c* devem ser compilados individualmente antes de gerarem um arquivo executável, para isso, basta digitar os seguintes comandos no terminal:

```

gcc -c TAD.c
gcc -c main.c
gcc TAD.o main.o -o main -lm
./main

```