

UNIVERSIDADE DE SÃO PAULO
INSTITUTO DE CIÊNCIAS MATEMÁTICAS E DE COMPUTAÇÃO
BACHAREL EM CIÊNCIAS DE COMPUTAÇÃO

MODELAGEM COMPUTACIONAL EM GRAFO: Relatório do Projeto Final

ALBERTO CAMPOS NEVES nº 108.190.33
GABRIEL VAN LOON BODÊ DA COSTA DOURADO FUENTES ROJAS nº 103.916.07
MATHIAS FERNANDES DUARTE COELHO nº 107.343.52

SÃO CARLOS - SP

1. INTRODUÇÃO AO TEMA DO PROJETO

O tema escolhido pelo grupo foi o *Tema 3: rede social de “amizade verdadeira”* cujo objetivo era modelar as relações entre os usuários de uma rede social de forma que fosse possível dizer se dois usuários possuem interesses comuns ou não.

Foi desenvolvido um sistema em C que funcionasse como modelo base para implementação do cálculo de compatibilidade entre usuários. O sistema atua com base em um **sistema de pontos de afinidade**, tal que quanto mais uma pessoa possui gostos similares ao seu, maior é a pontuação entre vocês.

O sistema possui as seguintes funcionalidades:

❑ Tela de login

- ❑ **Fazer login:** o usuário entra na sua conta por meio do login cadastrado;
- ❑ **Criar uma conta:** o usuário cria uma nova conta no sistema para fazer login;
- ❑ **Listar contas cadastradas:** função utilitária que lista todos os logins registrados no sistema para que o usuário possa selecionar um e entrar;
- ❑ **Sair do sistema.**

❑ Tela do Usuário

- ❑ **Ver lista de amigos:** exibe a lista de amigos do usuário ordenada daqueles com maior para menor afinidade. Amigos cuja pontuação de afinidade seja inferior à um limiar estabelecido são exibidos com pontuação em vermelho;
- ❑ **Ver sugestões de amizade:** exibe os usuários do sistema ordenados por ordem decrescente de pontuação de afinidade. Assim como na lista, usuários com baixa afinidade são listados em vermelho;
- ❑ **Enviar solicitação de amizade:** o usuário envia uma solicitação de amizade para algum usuário do sistema;
- ❑ **Excluir amizade:** exclui algum usuário da sua lista de amigos. O usuário também é excluído na lista de amigos de quem removeu;

- ❑ **Encontrar parceiro ideal:** baseado no seu gênero e orientação sexual, o sistema busca e retorna a pessoa solteira com maior pontuação no sistema e que também tenha atração pelo seu gênero;
- ❑ **Sair da tela do usuário.**

2. DECISÕES DE IMPLEMENTAÇÃO

A primeira decisão do projeto foi de como armazenar os dados dos usuários e também seus relacionamentos, haja vista que, por se tratar de uma rede social, seria ineficiente recadastrar todos os usuários 1 a 1 toda vez que o programa fosse executado.

Utilizamos, portanto, os seguintes arquivos (e bibliotecas para manipulá-los):

- ❑ **Usuários.bin:** Arquivo que serve como index de usuário. Possui um registro de cabeçalho com a quantidade de usuários salvos e cada registro de dados possui o seguinte formato:
 - ❑ **idUsuario:** número de identificação do usuário.
 - ❑ **login:** login único que serve para entrar no sistema.
- ❑ **Dados.bin:** Arquivo que salva os dados pessoais do usuário, sendo eles o nome, idade, gênero, orientação sexual e se o usuário está ou não namorando atualmente.
- ❑ **Solicitações.bin:** Arquivo que registra solicitações de amizade pendentes para o usuário aceitar ou recusar. Há um limite de até 100 solicitações pendentes.

Para modelar o relacionamento entre os usuários, o grupo decidiu limitar o número de amigos que cada usuário poderia ter (sendo o máximo de 100 amigos). Dessa forma seria possível definir uma estrutura de dados de tamanho fixo para salvar no arquivo **Relacionamentos.bin**.

As relações de amizade foram pensadas como um grafo utilizando lista de adjacências, pois dessa forma conseguimos modelar um registro do arquivo relacionamento.bin da seguinte forma:

| | | | | | | |
|-----------|-----------|----------|--------|----------|--------|-----|
| idUsuario | nroAmigos | idAmigo1 | pontos | idAmigo2 | pontos | ... |
|-----------|-----------|----------|--------|----------|--------|-----|

Na prática, o modelo se aproxima ao que foi visto em aula, sendo idUsuario o número do vértice e no arquivo está salvo, de maneira encadeada e ordenada, todas as conexões que o usuário possui.

Tratando agora da parte de como calcular a afinidade entre usuários, o grupo decidiu que a melhor opção seria deixar fixado quais seriam os tópicos que o usuário poderia selecionar ao cadastrar uma conta. Isso se deve ao fato de que seria muito caro em questão de tempo e complexidade ter que controlar o cadastro de infinitas opções de gostos ou temas.

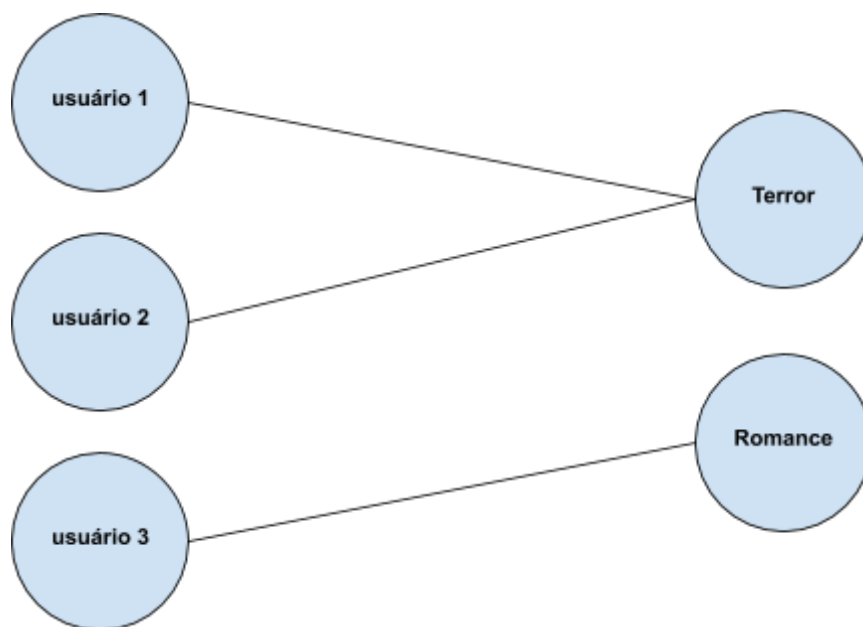
No sistema atual, o usuário decide seus gostos ao criar seu cadastro e selecionar uma opção para cada um dos seguintes temas:

- ❑ **Gosto musical:** sertanejo, rock, gospel, mpb, pop, reggae, samba, pagode, funk, eletronica, outro gênero;
- ❑ **Cidade em que vive:** Analandia, Araraquara, Brotas, Campinas, Ibaté, Limeira, Rio Claro, São Carlos ou São Paulo;
- ❑ **Gênero de filme favorito:** Ação, Animação, Aventura, Comédia, Comédia Romântica, Documentário, Drama, Ficção Científica, Faroeste, Musical, Romance, Suspense, Terror, outro gênero;
- ❑ **Gênero de livro favorito:** Ciência, Filosofia, Ficção, História, Poesia, Auto Ajuda, Romance, Novela, outro gênero;
- ❑ **Tribo / Grupo social:** Nerd ou Geek, Hippie, Gótico, Hipster, Apreciador de cultura coreana (K-pop), Apreciador de cultura japonesa (Animes, etc), Fitness, outro grupo;

As características acima não são, pensando em uma aplicação real, suficientes para conseguir definir se duas pessoas tem, de fato, os mesmos gostos, uma vez que seria possível enquadrar outros milhares de tópicos como times de

futebol, hobbies e lazers e muitos outros aspectos. Porém, como o objetivo era focar na parte de implementar um jeito de modelar essas semelhanças achamos que essa quantidade de informação era suficiente.

Para a modelagem do **Grafo.bin**, definimos utilizar para modelar as semelhanças entre os usuários um **grafo bipartido**, pois dessa forma, quando um usuário define as suas características o sistema insere um novo vértice de usuário e as seguintes arestas: (idUserio, idMusica), (idUserio,idCidade), (idUserio, idFilme), (idUserio,idLivro) e (idUserio,idTribo).



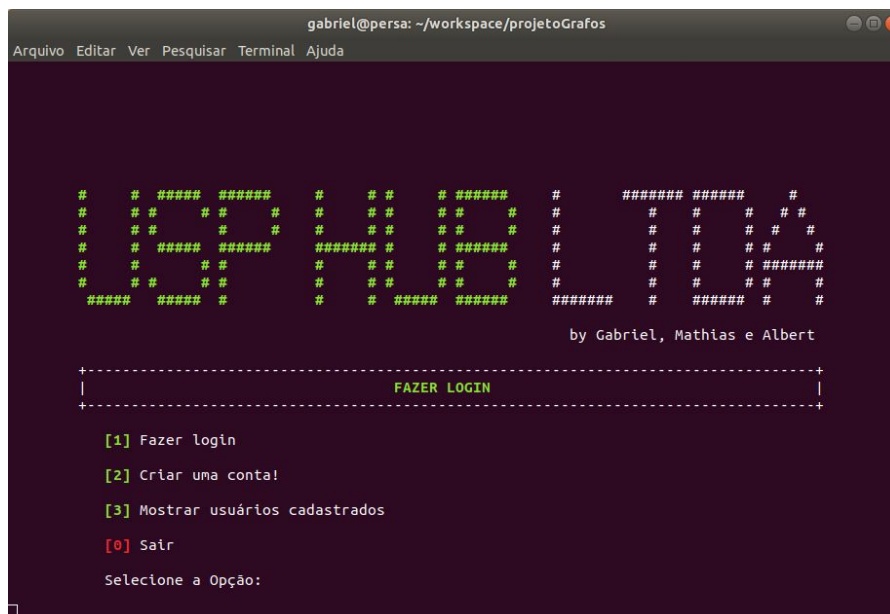
Exemplo de grafo bipartido. Os usuários (à esquerda) se conectam à uma das opções para cada tema.

Dentro do programa utilizamos uma matriz de adjacências do tipo usuários x opções (matriz[idUsuario][idOpcao]) para salvar todas as conexões. Escolhemos essa abordagem por se tratar de um sistema de pequeno porte, pois caso fosse necessário inserir uma grande quantidade de usuários se possível utilizar uma lista de adjacências estruturada da seguinte forma:

(idOpção) -> {(usuario1), (usuario2), (usuarioX)}

Dessa forma seria possível manter a performance da etapa de calcular os pontos de afinidade sem consumir memória com arestas vazias.

Por fim, no final na busca o sistema constrói um Rank de usuários que será utilizado pelas funcionalidades do sistema.



```
gabriel@persa: ~/workspace/projetoGrafos
Arquivo Editar Ver Pesquisar Terminal Ajuda

# # ##### ##### # # # # ##### # ##### #
# # # # # # # # # # # # # # #
# # # # # # # # # # # # # # #
# # ##### ##### ##### # # # #
# # # # # # # # # # # # # #
##### ##### # # # ##### ##### ##### # #

by Gabriel, Mathias e Albert

+-----+
| CRIAR CADASTRO - DADOS PESSOAIS |
+-----+

Nome: Bianca Tavares Teste

Idade: 21

Gênero Masculino [m] ou Feminino [f]: f

Orientação Hetero[e], Homossexual[o] ou Bissexual[b]: b

Estou namorando (s/n): n
```

```
gabriel@persa: ~/workspace/projetoGrafos
Arquivo Editar Ver Pesquisar Terminal Ajuda

# # # # # # # # # # # # # #
##### ##### # # # ##### ##### ##### # #

by Gabriel, Mathias e Albert

+-----+
| CRIAR CADASTRO - ESTILO MUSICAL |
+-----+

[a] Sertenejo
[b] Rock
[c] Gospel
[d] MPB
[e] Pop
[f] Reggae
[g] Samba
[h] Pagode
[i] Funk
[j] Eletronica
[k] Outro

Selecione a Opção: 
```



```
gabriel@persa: ~/workspace/projetoGrafos
Arquivo Editar Ver Pesquisar Terminal Ajuda

# # ##### ##### # # # # ##### # ##### #
# # # # # # # # # # # # # # #
# # # # # # # # # # # # # # #
# # ##### ##### ##### # # # # #
# # # # # # # # # # # # # # #
# # # # # # # # # # # # # # #
##### ##### # # # ##### ##### ##### #

by Gabriel, Mathias e Albert

+-----+
| DASHBOARD DO USUARIO |
+-----+

[1] Ver lista de amigos
[2] Ver sugestões de amizades
[3] Enviar solicitação de amizade
[4] Excluir amizade
[5] Encontrar parceiro(a) ideal
[0] Sair

Selecione a Opção: 
```

```
gabriel@persa: ~/workspace/projetoGrafos
Arquivo Editar Ver Pesquisar Terminal Ajuda

# # ##### ##### # # # # ##### # ##### #
# # # # # # # # # # # # # # #
# # # # # # # # # # # # # # #
# # ##### ##### ##### # # # # #
# # # # # # # # # # # # # # #
# # # # # # # # # # # # # # #
# # # # # # # # # # # # # # #
##### ##### # # # ##### ##### ##### #

by Gabriel, Mathias e Albert

+-----+
| RANK DE PESSOAS COM MAIOR AFINIDADE COM SEU PERFIL |
+-----+

@kashin (200 pontos de afinidade)
@math (50 pontos de afinidade)
@albert (50 pontos de afinidade)
@gigi (0 pontos de afinidade)

pressione ENTER para voltar...

```

```
gabriel@persa: ~/workspace/projetoGrafos
Arquivo Editar Ver Pesquisar Terminal Ajuda

# # ##### ##### # # # # ##### # ##### ##### #
# # # # # # # # # # # # # # # # # #
# # # # # # # # # # # # # # # # #
# # ##### ##### ##### # # ##### #
# # # # # # # # # # # # # # # # #
##### ##### # # # ##### ##### ##### # ##### #

by Gabriel, Mathias e Albert

+-----+
| LOVE FINDER - ENCONTRE SEU PARCEIRO IDEAL |
+-----+

@bianca (250 pontos de afinidade)

Nome: Bianca Tavares Teste.

Idade: 21 anos.

pressione ENTER para voltar...
```

```
gabriel@persa: ~/workspace/projetoGrafos
Arquivo Editar Ver Pesquisar Terminal Ajuda

# # ##### ##### # # # # ##### # ##### ##### #
# # # # # # # # # # # # # # # # # #
# # # # # # # # # # # # # # # # #
# # ##### ##### ##### # # ##### #
# # # # # # # # # # # # # # # # #
##### ##### # # # ##### ##### ##### # ##### #

by Gabriel, Mathias e Albert

+-----+
| MINHA LISTA DE AMIGOS |
+-----+

Você possui 2 amigos.

@kashin (200 pontos de afinidade)

@math (50 pontos de afinidade)

pressione ENTER para voltar...
```