

TEG

Gilmário B. Santos

*[gilmario.santos@udesc.br](mailto:gilmario.santos@udesc.br)*

*<http://www.joinville.udesc.br/portal/pagina/gilmario>*

# TEG

## Ementa:

Conceitos e definições de grafos. Representação computacional. Conexividade, Isomorfismo, Planaridade e Coloração.

Ordenação topológica. Grafos Hamiltonianos, Eulerianos e Árvores. Buscas em Grafos. Caminho Mínimo. Árvore geradora.

Fluxos em Redes. Introdução ao estudo de estruturas combinatórias.

# TEG

Avaliação: mínimo 3 atividades na forma de trabalho prático envolvendo a implementação autoral com a entrega do código fonte desenvolvido e relatório e/ou nota obtida em uma prova escrita.

Implementações práticas:

Em “C” (e talvez em Python)

Moodle: repositório do material e atividades do curso.

# TEG

Para as atividades na forma de trabalhos práticos:

- Relatório em formato PDF;
- Codificação em linguagem C (exceto se for especificado em contrário);
- Implementação em código compilável no Linux;
- Código, relatório etc são entregues exclusivamente via Moodle, compactados e dentro dos prazos;
- Nomeie o arquivo compactado como *T\_#\_iniciaisDosAutores*
  - Onde “#” é o número da tarefa. Exemplo: T\_1\_JS\_PF.zip para a submissão da solução da tarefa n. 1, cujos autores são José Silva e Pedro Francisco.
- Se for o caso, haverá 0,25 de desconto para cada 6 horas de atraso do *upload*;

# TEG

Para as atividades na forma de trabalhos práticos:

Codificação em linguagem e compilável no Linux:

Utilize um C padrão sem bibliotecas específicas para SO Windows ou indique as bibliotecas equivalentes (às específicas) que devem ser usadas na compilação no SO Linux;

Compile usando “`gcc -Wall -o ./saida programa.c`” e elimine cada warning que ocorrer.

Relatório: sempre em PDF, a função do relatório é:

Identificar a equipe de execução e os papéis de cada membro (se todo mundo fez tudo, então qualquer membro tem que ter a capacidade de responder a qualquer pergunta);

Apresentar um resumo descritivo de atividades realizadas bem como informar os dados e resultados coletados com elas. Sua estrutura apresenta: título, introdução, referências, desenvolvimento, conclusão e, em alguns casos, sugestões.

Maiores

detalhes:

<https://mundoeducacao.uol.com.br/redacao/o-relatorio.htm>

# TEG – descontos sobre tarefas atrasadas

- A) No Moodle: "Data de entrega": entrega sem desconto de nota;
- B) No Moodle: "Data limite" (firm deadline): entrega em atraso e com desconto de nota;
- C) Para as entregas atrasadas haverá desconto proporcional ao atraso, conforme é explicado a seguir:

$$0 \leq d \leq 10;$$

A nota final "N1" será  $N1=10-d$ ;

"d" proporcional à quantidade de horas de atraso "h", segundo a relação  $d=\text{round}(h/12)$ .

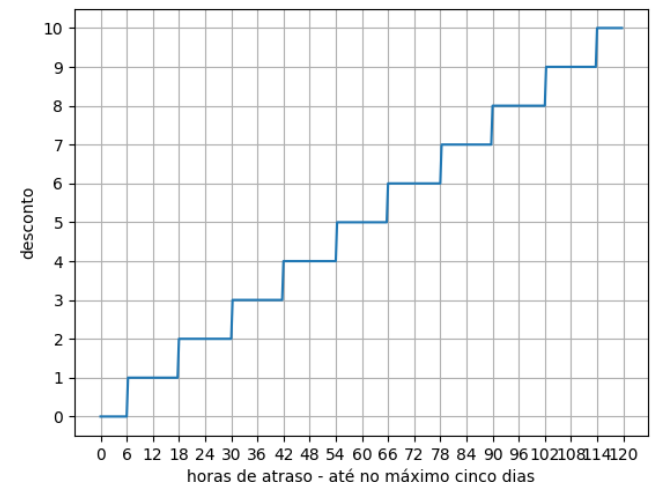
Exemplos de descontos:

Subiu a solução 3 horas depois da data de entrega: atraso=3 horas,  $d=\text{round}(3/12)$

Subiu a solução com atraso=20 horas,  $d=\text{round}(20/12)=\text{round}(1,6666)=2,0$

Subiu a solução às 23:59:59 do último dia da data limite (quinto dia), atraso=120 horas,  $d=\text{round}(120/12)=10$

D) Lembrem-se que, além da correção das soluções, também se aplica o critério comparativo entre soluções na determinação da nota da avaliação. Bem como a dupla pode ser chamada para uma entrevista que corrobore a solução apresentada.



# TEG

## Bibliografia

### Básica

LUCCHESI, C. L. et alli. Aspectos Teóricos da Computação, Parte C: Teoria dos Grafos, projeto Euclides, 1979.

SANTOS, J. P. O. et alli. Introdução à Análise Combinatória. UNICAMP; 1995.

SZWARCFITER, J. L. Grafos e Algoritmos Computacionais. Campus, 1986.

GERSTING, Judith L. Fundamentos Matemáticos para a Ciência da Computação. Rio de Janeiro. 3a Ed. Editora.

### Complementar:

1.) CORMEN, T. Introduction to Algorithms, third edition, MIT press, 2009

2.) ROSEN, K. Discrete Mathematics and its applications, seventh edition, McGraw Hill, 2011.

3.) WEST, Douglas, B. Introduction to Graph Theory, second edition, Pearson, 2001.

4.) BONDY, J.A., MURTY, U.S.R., Graph Theory with applications , Springer, 1984.

5.) SEDGEWICK, R. Algorithms in C - part 5 - Graph Algorithms, third edition, 2002, Addison-Wesley.

6.) GOLDBARG, M., GOLDBARG E., Grafos: Conceitos, algoritmos e aplicações. Editora Elsevier, 2012.

7.) BONDY, J.A., MURTY, U.S.R., Graph Theory with applications , Springer, 1984

8.) FEOFILOFF, P., KOHAYAKAWA, Y., WAKABAYASHI, Y., uma introdução sucinta à teoria dos grafos. 2011.  
([www.ime.usp.br/~pf/teoriadosgrafos](http://www.ime.usp.br/~pf/teoriadosgrafos))

9.) DIESTEL, R. Graph Theory, second edition, springer, 2000

10.) FURTADO, A. L. Teoria de grafos. Rio de janeiro. Editora LTC. 1973.

11.) WILSON, R.J. Introduction to Graph Theory. John Wiley & Sons Inc., 1985

12.) BOAVENTURA NETTO , P. O. Grafos: Teoria, Modelos, Algoritmos. Edgard Blucher, SP, quinta edição

Tutoriais, artigos, notas de aula...

Vários livros podem ser acessados no formato eletrônico (e-book) via

<https://www.udesc.br/bu/acervos/ebook>

Exemplos:



Fundamentos  
Matemáticos para a  
Ciência da Computação

Judith L. Gersting



Teoria Computacional de  
Grafos - Os Algoritmos

Jayme Luiz Szwarcfiter



Fundamentos  
Matemáticos para a  
Ciência da Computação

Judith L. Gersting



Grafos

Marco Goldberg



Algoritmos - Teoria e  
Prática

Thomas Cormen



## Aulas e interação

- O material do curso estará centrado no MOODLE
- Se necessário poderemos utilizar recurso de aula online via BBB:



BigBlueButton™

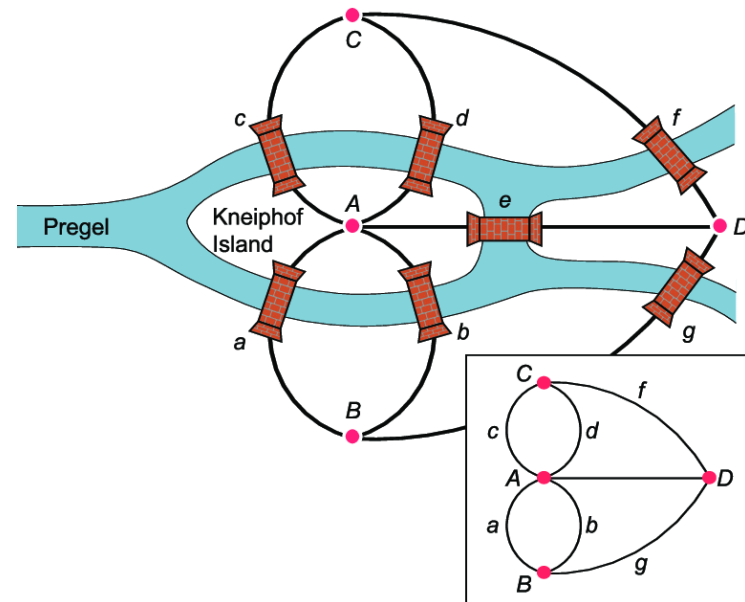
Onde você já viu um grafo?



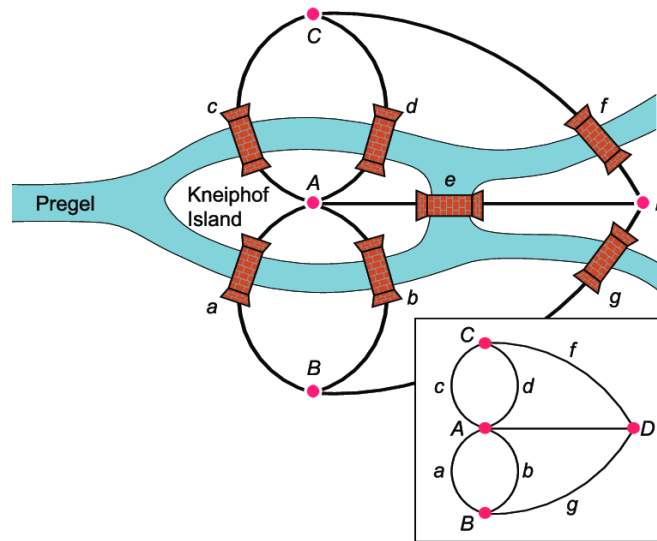
Um pouco da história oficial:

Um dos primeiros registros históricos da utilização de grafos surgiu do problema das Pontes de Königsberg. A cidade de Königsberg é a antiga capital da Prússia Oriental, conhecida atualmente pelo nome de Kaliningrado.

A cidade é dividida em 4 zonas criadas pelo percurso do rio Pregel, no séc. XVII essas zonas estavam ligadas por sete pontes conforme a figura.



Um pouco da história oficial:



Por muito tempo os habitantes da cidade questionavam se era possível cruzar as sete pontes numa caminhada contínua sem passar duas vezes por qualquer uma delas!

Em 1736, o matemático Leonhard Euler demonstrou que não existe tal trajeto, ele utilizou um modelo em grafos para uma generalização deste problema. Através desse modelo ele verificou as condições necessárias e suficientes para o trajeto desejado, ele só ocorreria quando e somente quando em cada região concorresse um número par de pontes.

Na verdade o problema consiste na determinação de um caminho Euleriano, ou seja, um trajeto que usa cada ponte exatamente uma vez...

## Áreas de aplicação da teoria de grafos:

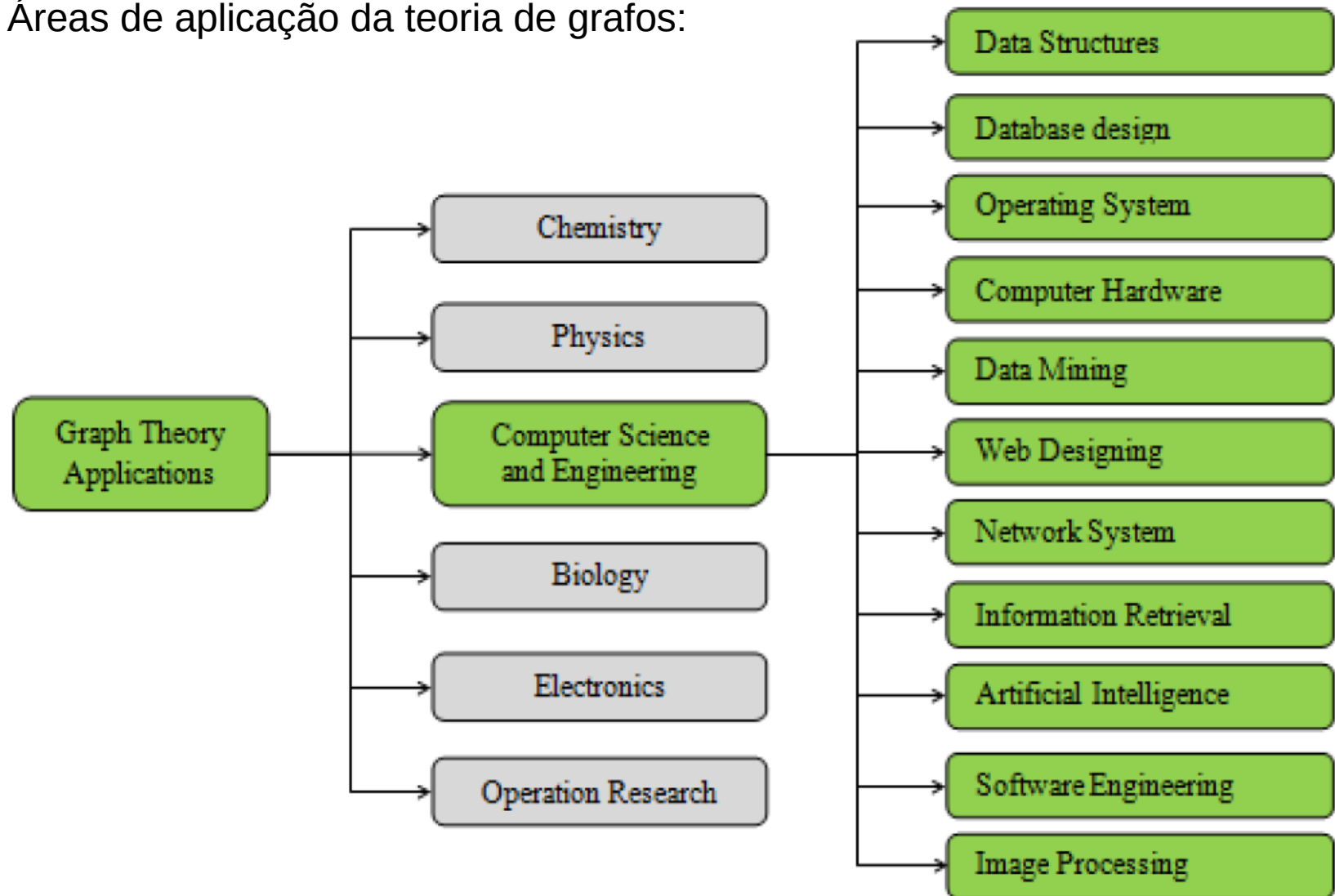


Figure 4. Graph theory applications in various fields

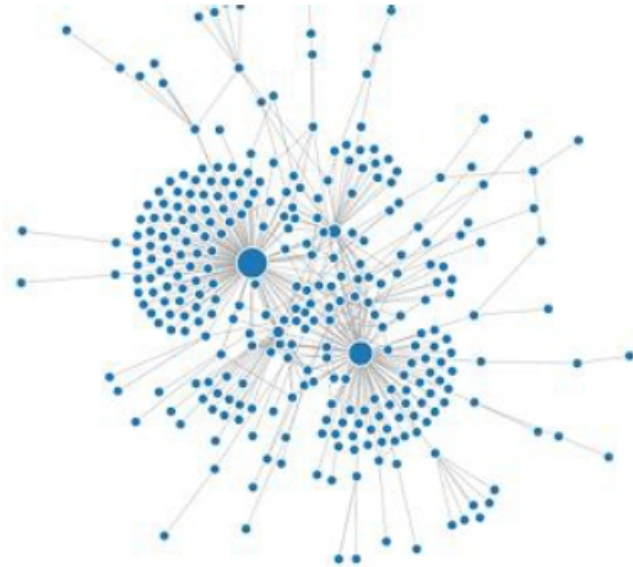
## Exemplo\_02: Facebook

Como saber se duas pessoas estão conectadas através de uma sequência de relacionamentos? Qual é o menor caminho entre duas pessoas?

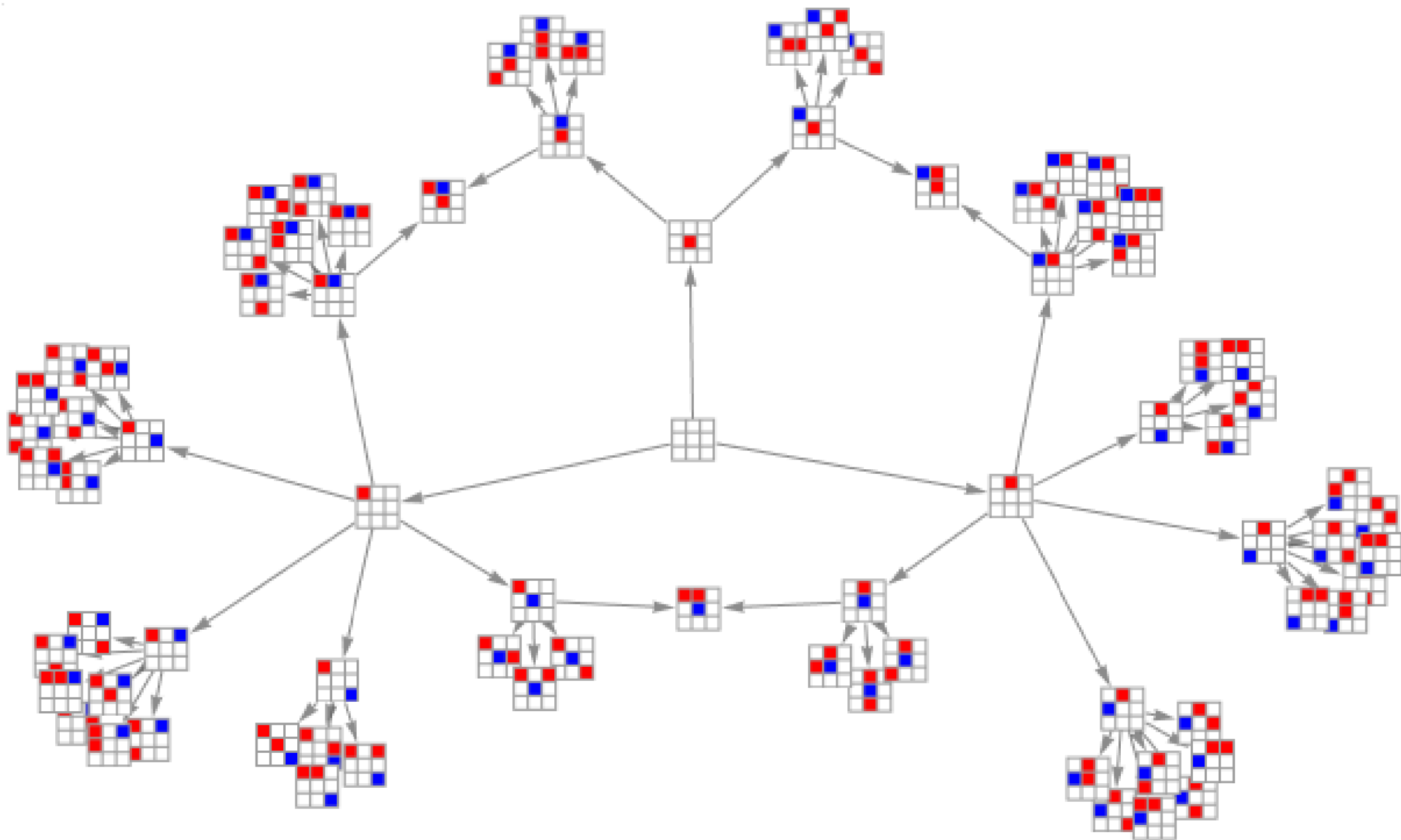
**Como abstrair o problema (via grafos)?**

**Objeto:** profiles das pessoas

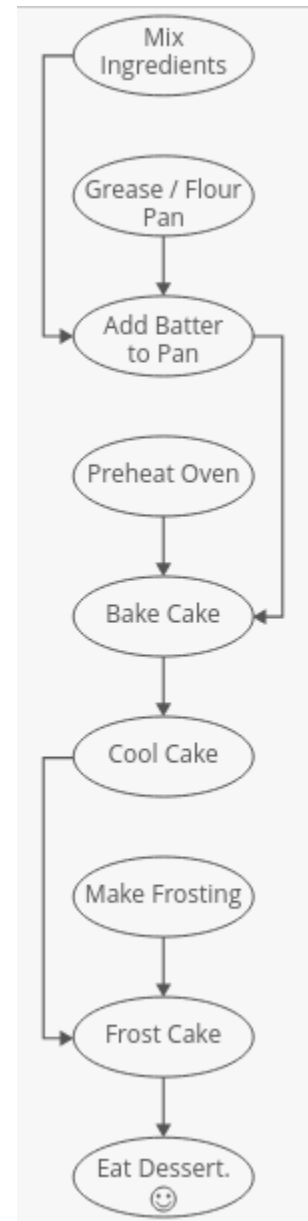
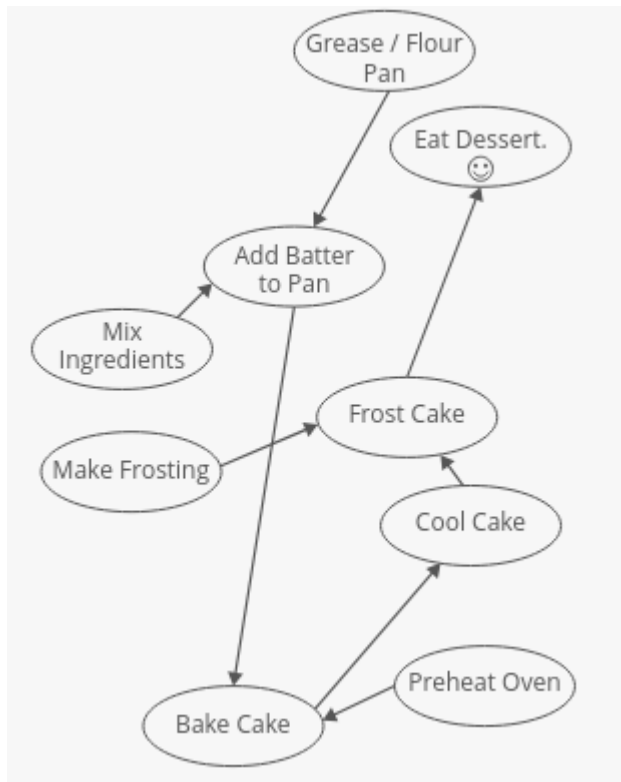
**Relacionamento:** relacionamentos declarados



O grafo dos primeiros três movimentos do jogo da velha:

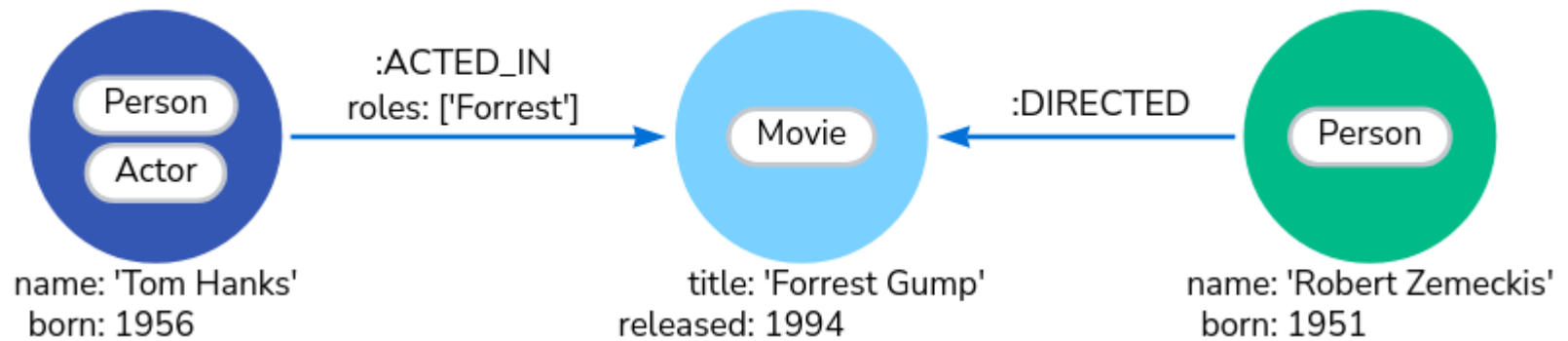


O grafo da receita de bolo a sua correspondente ordenação topológica:





## Parte de um grafo sobre o filme “Forrest Gump”



<https://neo4j.com/docs/getting-started/appendix/graphdb-concepts/>