

# TNA1MCTA027-17SA, TDA1MCTA027-17SA e TDB1MCTA027-17SA - TEORIA DOS GRAFOS - Maycon Sambinelli - 2021.Q3

Painel / Meus cursos / GRAPH THEORY

Seu progresso

- Turmas: TNA1MCTA027-17SA, TDA1MCTA027-17SA e TDB1MCTA027-17SA
- Professor: Maycon Sambinelli
- E-mail: m.sambinelli@ufabc.edu.br
- Caixa de Sugestões



## Objetivos - Alternar

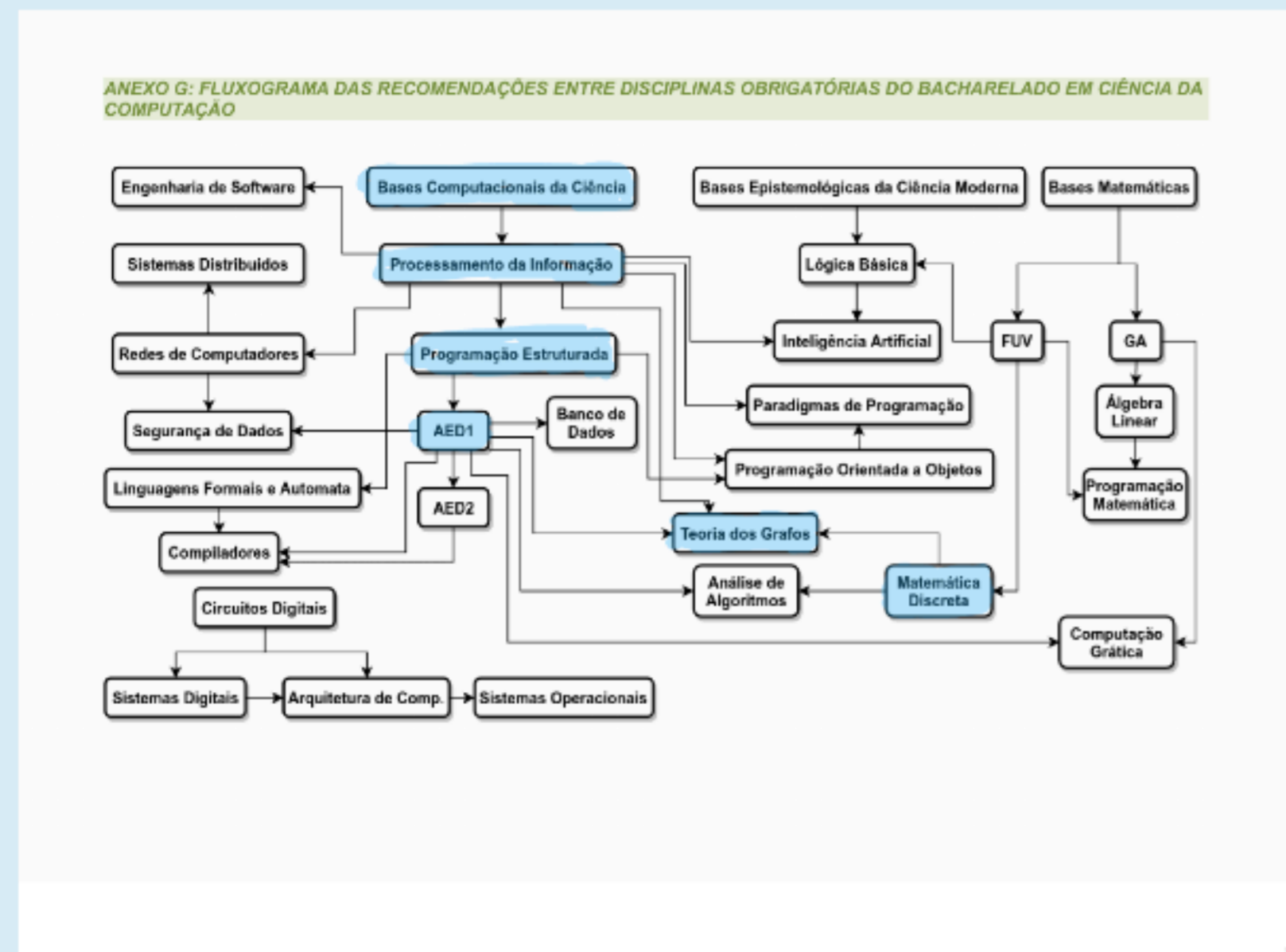
- Apresentar os conceitos e resultados básicos da Teoria dos Grafos;
- Permitir o uso de grafos e suas propriedades para modelar problemas computacionais;
- Apresentar algoritmos eficientes para problemas recorrentes em computação;
- Tornar familiares certos padrões de soluções que ocorrem frequentemente em problemas envolvendo grafos.

## Ementa da Disciplina - Alternar

Conceitos básicos de grafos dirigidos e não dirigidos. Passeios, caminhos, circuitos. Grafos bipartidos e multipartidos. Subgrafos. Isomorfismo. Conexidade. Florestas e árvores. Exemplos de problemas de interesse: coloração de vértices; clique máximo; caixeiro viajante; problemas de fluxo. Estruturas de dados para a representação de grafos. Percursos em grafos: em largura, em profundidade. Ordenação topológica. Árvores geradoras mínimas. Algoritmo de Kruskal. Caminhos mínimos em grafos: algoritmo de Dijkstra, algoritmo de Floyd-Warshall. Emparelhamentos: Teorema de Hall. (Disponível na pg. 79 do projeto pedagógico.)

## Recomendação - Alternar

Disciplinas: Matemática Discreta; Processamento da Informação; Algoritmos e Estruturas de Dados I



Para facilitar o acompanhamento do curso, é recomendado que você já possua:

- Familiaridade com programação, com boas noções de algoritmos
- Familiaridade com estruturas de dados básicas (vetores, listas, pilhas, filas e árvores)
- Capacidade para reconhecer argumentos lógicos em uma prova matemática (por indução, contradição, construção, contrapositiva)
- Familiaridade com linguagem matemática (como quantificadores lógicos, conjuntos, somatórios e manipulação de funções)

### Materiais de apoio para esses tópicos:

- Velleman, D. J., *How to Prove It: A Structured Approach*. Second Edition. Cambridge University Press. 2006.
- Albert R. Meyer, Eric Lehman, and Frank Thomson Leighton, *Mathematics for Computer Science*, 2018
- O que é uma prova matemática, do prof. Paulo Feofiloff, da USP.
- Matemática discreta para computação, dos profs. Anamaria Gomide e Jorge Stolfi, da Unicamp.
- Indução matemática, do prof. Cid Carvalho de Souza, da Unicamp.
- Portal da Matemática da OBMEP.
- Portal da Matemática da OBMEP (apostilas).
- Indução e contagem, do prof. Rogério Steffenon e Felipe Guarnieri, da Unisinos.
- Indução Matemática, Abramo Hefez
- Projeto de algoritmos (em C), do prof. Paulo Feofiloff, da USP.
- Estruturas de dados (em C), do prof. Paulo Feofiloff, da USP.
- Notas de aula, da disciplina de Programação Estruturada, da prof. Caria Lintzmayer (introdução à programação em C, recursão, vetores e listas).

## Bibliografia - Alternar

- [CLRS3] Cormen, T. H.; Leiserson, C. E.; Rivest, R. L.; Stein, C., *Introduction to Algorithms*. 3rd ed. MIT Press. 2009. (Tradução)
- [S] Sedgwick, R., *Algorithms in C, part 5: graph algorithms*. 3rd edition. Addison-Wesley. 2002.
- [BM] Bondy, J. A.; Murty, U. S. R. *Graph theory*. Graduate Texts in Mathematics. Springer. New York. 2008.
- [LM] Lintzmayer, C. N.; Mota, G. O., *Análise de Algoritmos e Estruturas de Dados*
- [Fe] Feofiloff, P., *Algoritmos para Grafos via Sedgwick*

## Cronograma - Alternar

	Data	Aula	Conteúdo
Seg	13/09	Introdução ao Curso	apresentação do curso, método de avaliação, moodle e etc
Qui	16/09	Definições básicas	Definições Básicas de Grafos: desenho de um grafo, ordem, tamanho, arestas paralelas, laço, grafo simples; Número máximo de arestas em grafos simples; Adjacência e Vizinhança; vizinhos; adjacentes; extremos; aresta incidente; vizinhança; grau; grau mínimo/máximo; Isomorfismo; Subgrafo (induzido); Passeio: vértices externos, vértices internos, comprimento, fechado e aberto; tripla; ciclo; subpasseio/subtrilha/subcaminho; conexidade; maximal e máximo; complemento; conjunto independente e clique; grafo completo; grafo ciclo;
Seg	20/09	Resultados Básicos	teo. do aperto de mãos: prova indutiva, prova por matriz de adjacências; teo. todo grafo ter um número par de vértices de grau ímpar; princípio da casa dos pombos; teo. todo grafo simples com ao menos dois vértices possui pelo menos um par de vértices com o mesmo grau; teo. se existe um passeio de u a v então existe um caminho de u a v; teo. se existe um caminho de u a v de v a w, então existe um caminho de u a w;
Qui	23/09	Grafos Bipartidos / Aresta e Vértice de Corte	def. grafo bipartido; teo. grafo é bipartido sse não contém ciclos de comprimento ímpar; def. distância entre dois vértices; def. c(G); def. aresta de corte e vértice de corte; teo. aresta é de corte sse não pertence a um ciclo;
Seg	27/09	Árvores	teo. e(G) >= v(G) - 1; def. de árvore; teo. caracterização de árvores;
Qui	30/09	Introdução à Análise de Algoritmos	Notação: O, Omega, Theta; Tempo de pior caso; Análise de tempo de algoritmos
Seg	04/10	Grafos como Estrutura de Dados	Grafo denso e esparsos; matriz de adjacências; lista de adjacências; Biblioteca em C para Grafos: graph_insert_edge; graph_remove_edge; graph_print; graph_has_edge; Grafo ponderado; Grafo com atributos; Otimizações; Grafo rotulado;
Qui	07/10	Busca em Grafos	def. uv-caminho; def. vértice alcançável; Algoritmo de Busca Genérica; Vector pred[];
Seg	11/10	Breadth-First Search	Exemplo de execução; pseudo-código; def. Single Source Shortest Path; Aplicação BFS: caminho mínimo;
Qui	14/10	DFS & Componentes conexas	Exemplo de execução; Pseudo-código; tipos de aresta: retorno, da árvore; aplicações: encontrar ciclos, aresta de corte (com pseudo-código);
Seg	18/10	Árvore Geradora (Prim)	Grafos ponderados; def. problema da árvore geradora mínima; alg. agn- genérico (prova corretude); Algoritmo de Prim: exemplo execução; pseudo-código
Qui	21/10	Árvore Geradora (Prim)	Algoritmo de Prim: corretude, análise de tempo (com lista, heap binária e heap de fibonacci); Heap binária: exemplo, pseudo-código, análise, código em C;
Seg	25/10	Árvore Geradora (Kruskal)	Algoritmo de Kruskal: ideia; exemplo de execução; pseudo-código; Estrutura para conjuntos disjuntos: exemplos, pseudo-código, análise
Qui	28/10	Euleriano	As pontes de Königsberg; def. grafo euleriano; teo. da caracterização de grafo euleriano; Algoritmo para tripla euleriana: análise, pseudo-código; def. ciclo hamiltoniano; def. TSP; alg. 2-aproximação para ao TSP (ideia);
Seg	01/11	🚫 Não haverá aula	
Qui	04/11	Digrafos + Ordenação Topológica	definições básicas: ordem, tamanho, arco, digrafo simples, adjacência e vizinhança, cauda, cabeça, vizinho de entrada/saída, vizinhança de entrada/saída, teo. soma dos graus de entrada é igual a de saída, subdigrafo, passeio/trilha/caminho/ciclo, grafo subjacente, conexidade, digrafos acíclicos, componentes fortemente conexas; representando um digrafo: matriz de adjacências, lista de adjacências; DFS; aplicações DFS: ordenação topológica; teo. D tem uma ordenação topológica sse é acíclico; teo. digrafo acíclico tem fonte; Algoritmo para encontrar uma ordenação topológica (DFS): pseudo-código;
Seg	08/11	Caminhos mínimos	motivação; def. problema do caminho mais curto de raiz única; def. relaxação de arestas; teo. geral dos caminhos mínimos
Qui	11/11	Algoritmos para o problema de caminhos mínimos (Dijkstra e Bellman)	Algoritmo de Dijkstra: ideia; código (java); análise; corretude; Algoritmo para Dag: ideia, código (java), análise, corretude; Algoritmo Bellman-Ford: ideia, código (java), análise; corretude; detecção de ciclos negativos;
Seg	15/11	Fluxo	motivação; definição; eliminação 2-ciclos; rede residual; caminho aumentador;
Qui	18/11	Fluxo	Teorema do Fluxo-Máximo Corte Mínimo; algoritmo Ford-Fulkerson: exemplo de execução, dicas de implementação
Seg	22/11	Fluxo	def. emparelhamento (máximo); redução do problema de encontrar emparelhamento máximo em um grafo bipartido para o problema de fluxo máximo; análise do algoritmo;

## Critérios de avaliação regular - Alternar

A média final antes da REC (MF) será calculada como sendo a média aritmética das tarefas.

O seu conceito final antes da REC (CF) será:

$$CF = \begin{cases} A, & \text{se } MF \in [8,5; 10,0] \\ B, & \text{se } MF \in [7,0; 8,5] \\ C, & \text{se } MF \in [6,0; 7,0] \\ D, & \text{se } MF \in [5,0; 6,0] \\ F, & \text{caso contrário} \end{cases}$$

🚫 Caso seja verificada ocorrência de fraude acadêmica, o aluno receberá MF = 0

## Mecanismo de recuperação - Alternar

- A recuperação será aplicada apenas aos alunos que tiverem conceito final D ou F.
- Consistirá numa avaliação, cujo conteúdo englobará todos os temas vistos durante o quadrimestre.
- A nota obtida na avaliação de recuperação (NR) será usada para obter a nota final com recuperação (NFR), que consiste na média:

$$NFR = \frac{MF + NR}{2}$$

- O conceito final com recuperação (CFR) será calculado da seguinte maneira:

$$CFR = \begin{cases} C, & \text{se } CF = D \text{ e } NFR \geq 6; \\ D, & \text{se } CF = D \text{ e } NFR < 6; \\ F, & \text{se } CF = F \text{ e } NFR \geq 5; \\ F, & \text{caso contrário.} \end{cases}$$

## Datas Importantes - Alternar

Recuperação: 18/02/2022

## Dias, horários e locais das aulas - Alternar

Os materiais de apoio referentes aos conteúdos cobertos na semana serão disponibilizados, aqui no Moodle, às segundas-feiras.

## Discord - Alternar

Neste quadrimestre, toda a nossa comunicação (avisos, dúvidas, atendimento) será feito pelo [Discord da disciplina](#).

## - Atendimento - Alternar

### Horários do Atendimento Sincrono:

- Quintas-feiras, das 10h às 12h, 16h às 18h e 21h às 23h.
  - Local: Google Meet
  - Agendar Horário

### Horários do Atendimento Assíncrono:

- 24 horas por dia, 7 dias por semana.
  - Local: [Discord](#)
  - Meta: respostas em até 24 horas (úteis).

## Fraude Acadêmica - Alternar

### Leitura obrigatória

Entre outras, o código de ética da UFABC estabelece em seu artigo 25 que é eticamente inaceitável que os discentes:

- fraudem avaliações,
- fabricarem ou falsifiquem dados,
- plagiem ou não creditem devida autoria,
- aceitem autoria de material acadêmico sem participação na produção,
- vendam ou cedam autoria de material acadêmico próprio a pessoas que não participaram da produção.

Durante a realização do curso você deve seguir as seguintes regras:

- Regra 1: Você não pode enviar para avaliação um trabalho que não seja de sua própria autoria ou que seja derivado/baseado em soluções elaboradas por outros.
- Regra 2: Você não pode discutir a sua solução com outros alunos e nem pedir aos seus colegas que compartilhem as soluções deles com você.

Qualquer violação às regras descritas acima implicará em MF = 0 para TODOS os envolvidos, sem prejuízo de outras sanções.