

UNIVERSIDADE FEDERAL DA BAHIA  
INSTITUTO DE COMPUTAÇÃO  
MATA53 - TEORIA DOS GRAFOS

ANOTAÇÕES 2024.2 GRAFOS

LUIS FELIPE

Salvador - Bahia  
25 DE JANEIRO DE 2025

# Sumário

<b>1</b>	<b>Introdução</b>	<b>1</b>
1.1	Contextualização e Motivação . . . . .	1
1.2	Exemplos de Operadores Matemáticos . . . . .	1
1.3	Exemplos de Teoremas e Definições . . . . .	2
1.4	Implementação . . . . .	2

# Capítulo 1

## Introdução

### 1.1 Contextualização e Motivação

A teoria dos grafos oferece um poderoso conjunto de ferramentas matemáticas para modelar e resolver problemas complexos de otimização em redes. No contexto da segurança pública, o problema de posicionamento de câmeras de vigilância pode ser elegantemente modelado como um problema de cobertura mínima de vértices (Minimum Vertex Cover). Nesta abordagem, os vértices do grafo representam possíveis localizações de câmeras, e as arestas representam as áreas que precisam ser monitoradas. O bairro de Ondina, em Salvador, apresenta um cenário ideal para aplicação deste conceito, por concentrar pontos estratégicos como a Universidade Federal da Bahia, estabelecimentos comerciais, hotéis e áreas residenciais, além de um intenso fluxo turístico devido às suas praias.

### 1.2 Exemplos de Operadores Matemáticos

Considere  $x = \frac{\pi}{4}$ :

- $\text{sen}(x) = \frac{\sqrt{2}}{2}$
- $\text{tg}(x) = 1$
- $\text{cotg}(x) = 1$
- $\text{arctg}(1) = \frac{\pi}{4}$
- $\text{agm}(1, \sqrt{2}) \approx 1.198$

## 1.3 Exemplos de Teoremas e Definições

**Definição 1.3.1.** Um grafo  $G = (V, E)$  é uma estrutura matemática composta por um conjunto  $V$  de vértices e um conjunto  $E$  de arestas que conectam pares de vértices.

**Teorema 1.3.1.** Todo grafo conexo com  $n$  vértices tem pelo menos  $n - 1$  arestas.

**Problema 1.3.1.** Dado um grafo  $G$ , encontre o menor conjunto de vértices que cobre todas as arestas.

**Corolário 1.3.1.** Se um grafo tem todos os vértices de grau par, então ele possui um ciclo euleriano.

**Proposição 1.3.1.** Em qualquer grafo, a soma dos graus de todos os vértices é igual ao dobro do número de arestas.

**Lema 1.3.1.** Em um grafo bipartido, não existem ciclos de comprimento ímpar.

## 1.4 Implementação

Segue o código de exemplo para representação de um grafo:

```
class Graph:
    def __init__(self):
        self.vertices = {}
        self.edges = []

    def add_vertex(self, vertex):
        if vertex not in self.vertices:
            self.vertices[vertex] = []

    def add_edge(self, v1, v2):
        if v1 in self.vertices and v2 in self.vertices:
            self.vertices[v1].append(v2)
            self.vertices[v2].append(v1)
            self.edges.append((v1, v2))

    def print_graph(self):
        for vertex in self.vertices:
            print(f"Vertex-{vertex}:-{self.vertices[vertex]}")
```

**Teorema 1.4.1.** O teorema 1 é uma proposição que afirma que o teorema 1 é uma proposição.