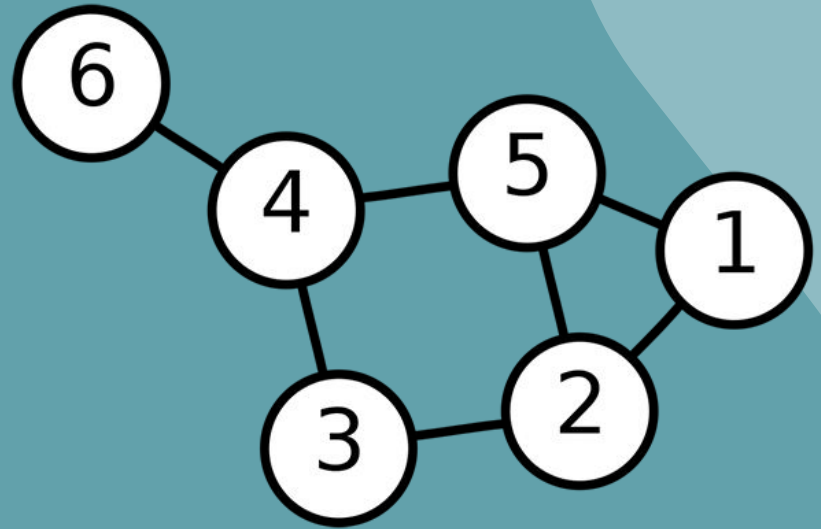


GRAFOS:

Grafos

k-partidos e

Coloração



Agenda

Grafos K-Partidos

São grafos em que os vértices podem ser particionados em K conjuntos independentes.

- Não há arestas entre os vértices dentro do mesmo conjunto, mas há arestas que conectam vértices de conjuntos diferentes.

Coloração de Vértices

Coloração de Vértices é o processo de atribuir uma cor a cada vértice de um grafo, de modo que vértices adjacentes não possuam a mesma cor.

Grafos

K-Partidos

Grafo Bipartido

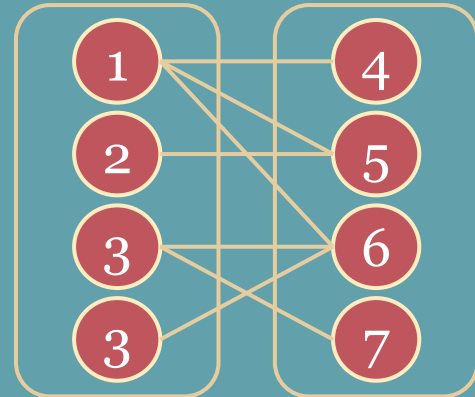
Grafo cujo conjunto de vértices V pode ser “partido” em dois conjuntos disjuntos V_1 e V_2 , de modo que não haja arestas ligando dois vértices do mesmo conjunto

- Ou seja, os vértices de V_1 só podem ter arestas ligando-os a vértices de V_2 , e vice-versa

Notação: $G = (V_1 + V_2, E)$

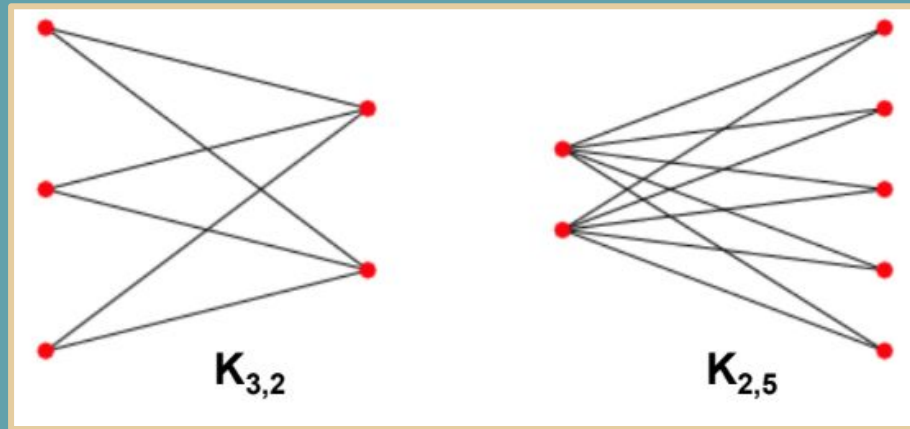
Exemplo:

- No diagrama, dois círculos delimitam as partições



Grafo Bipartido Completo

- Cada vértice tem aresta para todos os vértices da outra partição
- Nomenclatura: $K_{m,n}$, para grafos com uma partição de m elementos e outra com n



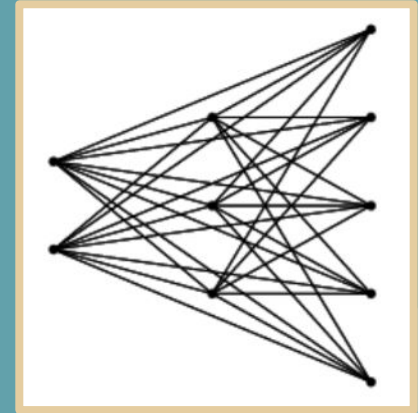
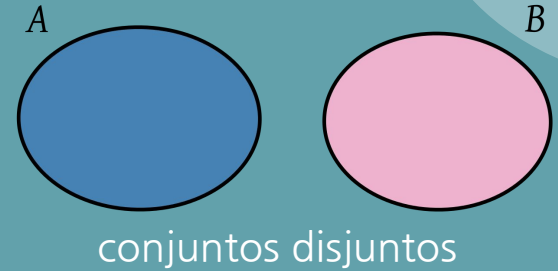
Grafo K-Partido

Generalização do conceito anterior

- É um grafo cujo conjunto de vértices pode ser particionado em k conjuntos disjuntos, de modo que não haja arestas entre vértices de um mesmo conjunto

Exemplo de grafo 3-partido

- No diagrama, vértices de uma mesma partição aparecem alinhados verticalmente

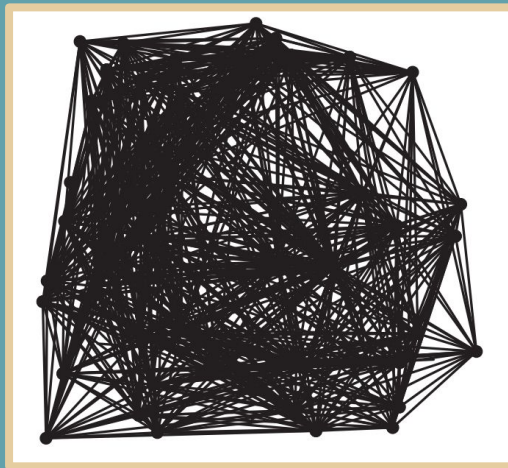


Grafo 3-partido

Grafos K-Partidos

Atenção: para ser k -partido, o grafo não precisa necessariamente ter as suas k partições explicitamente representadas - basta que se prove que elas existem

Um problema importante é o de descobrir se um grafo qualquer é k -partido, para um valor k dado



$$V = 50$$

$$E = 1000$$

Coloração de Vértices

Problema de Coloração dos Vértices

Como atribuir cores aos vértices sem que vértices adjacentes recebam as mesmas cores?

Origem: coloração de mapas políticos

- É um problema clássico de otimização combinatória em que o objetivo é colorir os vértices de um mapa (que pode ser representado por um grafo) de modo que nenhum vértice adjacente tenha a mesma cor.

Quantas cores são necessárias para colorir um mapa?

- É conhecido por ser NP-difícil

Tem aplicações práticas em diversas áreas, como no planejamento de redes de comunicação e **transporte**, design de circuitos eletrônicos, programação de horários escolares, entre outros.

Problema de Coloração dos Vértices

O número mínimo de cores que pode ser usado para colorir um grafo é chamado o **número cromático** do grafo

Ou seja, o **número cromático** é o menor valor k tal que o grafo é k -colorível

Algoritmos

- Algoritmos que encontram as soluções perfeitas são ineficientes, mas existem algoritmos eficientes que encontram boas aproximações
- Veremos apenas um algoritmo de aproximação simples...
 - **Algoritmo Sequential Coloring**

Algoritmo Sequential Coloring

- Recebe um grafo e um valor k
- Considera as cores ordenadas em uma seqüência $\{C_1, C_2, \dots, C_k\}$
- Para cada vértice, testa cada uma das cores, na ordem, até achar a primeira permitida
 - Ou seja, a primeira que não causa conflito com a cor de um vizinho

Algoritmo Sequential Coloring

Pseudo-código

```
SEQ-COLORING( grafo  $G$ , natural  $k$  )  
  para todo vértice  $v$   
    para cada cor  $C_i$  de  $\{C_1, \dots, C_k\}$   
      se ( $v$  pode assumir a cor  $C_i$ )  
        cor[ $v$ ] =  $C_i$   
      volta ao loop externo
```

OBRIGADO!

DÚVIDAS?

marcos.azevedo@unicap.br

CREDITS: This presentation template was created by **Slidesgo**,
including icons by **Flaticon** and infographics & images by
Freepik