Teoria dos Grafos e Computabilidade

— Overview —

Silvio Jamil F. Guimarães

Graduate Program in Informatics – PPGINF
Laboratory of Image and Multimedia Data Science – IMScience
Pontifical Catholic University of Minas Gerais – PUC Minas

Feb 2025

Teoria dos Grafos e Computabilidade

— Motivação —

Silvio Jamil F. Guimarães

Graduate Program in Informatics – PPGINF
Laboratory of Image and Multimedia Data Science – IMScience
Pontifical Catholic University of Minas Gerais – PUC Minas

Feb 2025

Motivação

Porque estudar grafos?

- Arcabouço matemático com aplicação em diversas áreas do conhecimento
- Utilizados na definição e/ou resolução de problemas
- Estudar grafos é mais uma forma de solucionar problemas computáveis
- Os estudos teóricos em grafos buscam o desenvolvimento de algoritmos mais eficientes.
- Abstração matemática que representa situações reais através de um diagrama.

Motivação

Porque estudar grafos?

- Arcabouço matemático com aplicação em diversas áreas do conhecimento
- Utilizados na definição e/ou resolução de problemas
- Estudar grafos é mais uma forma de solucionar problemas computáveis
- Os estudos teóricos em grafos buscam o desenvolvimento de algoritmos mais eficientes.
- Abstração matemática que representa situações reais através de um diagrama.

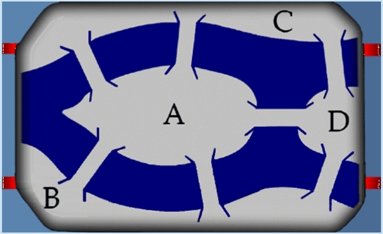
Áreas de conhecimento

Genética, química, pesquisa operacional, telecomunicações, engenharia elétrica, redes de computadores, conexão de vôos aéreos, restrições de precedência, fluxo de programas, dentre outros

Motivação

Pontes de Königsberg

O rio Pregel divide o centro da cidade de Königsberg (Prússia no século XVII, atual Kaliningrado, Rússia) em quatro regiões. Essas regiões são ligadas por um complexo de sete (7) pontes, conforme mostra a figura. Discutia-se nas ruas da cidade a possibilidade de atravessar todas as pontes, voltando ao lugar de onde se saiu, sem repetir alguma. Havia-se tornado uma lenda popular a possibilidade da façanha quando **Euler**, em 1736, provou que **não existia caminho** que possibilitasse tais restrições.



Motivação

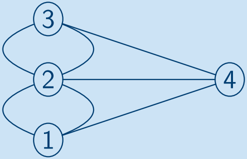
Pontes de Königsberg

- Resolvido em 1736 por Leonhard Euler
- Necessário um modelo para representar o problema
- Abstração de detalhes irrelevantes:
  - Área de cada ilha
  - Formato de cada ilha
  - Tipo da ponte, etc.

Motivação

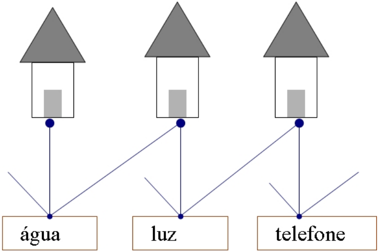
Pontes de Königsberg

- Resolvido em 1736 por Leonhard Euler
- Necessário um modelo para representar o problema
- Abstração de detalhes irrelevantes:
  - Área de cada ilha
  - Formato de cada ilha
  - Tipo da ponte, etc.
- Euler generalizou o problema através de um modelo de grafos



Problemas das 3 casas

É possível conectar os 3 serviços às 3 casas sem haver cruzamento de tubulação?



Quantas cores são necessárias para colorir o mapa do Brasil, sendo que estados adjacentes não podem ter a mesma cor?

De forma a reduzir seus custos operacionais, uma empresa de transporte de cargas deseja oferecer aos motoristas de sua frota um mecanismo que os auxilie a selecionar o melhor caminho (o de menor distância) entre quaisquer duas cidades por ela servidas, de forma a que sejam minimizados os custos de transporte.

Teoria dos Grafos e Computabilidade

— Definição de Grafos —

Silvio Jamil F. Guimarães

Graduate Program in Informatics – PPGINF
Laboratory of Image and Multimedia Data Science – IMScience
Pontifical Catholic University of Minas Gerais – PUC Minas

Feb 2025

Conceitos

Grafo

Grafo é uma coleção de vértices e arestas

Vértices

Vértice é um objeto simples que pode ter nomes e outros atributos

Arestas

Arestas é uma conexão entre dois vértices

Conceitos

Grafo

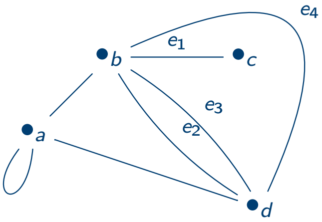
Grafo é uma coleção de vértices e arestas

Vértices

Vértice é um objeto simples que pode ter nomes e outros atributos

Arestas

Arestas é uma conexão entre dois vértices



Conceitos

Grafo

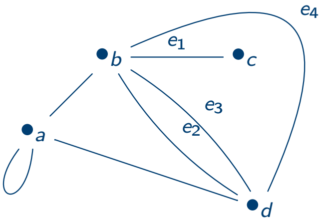
Grafo é uma coleção de vértices e arestas

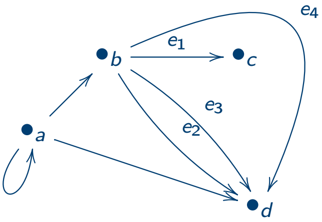
Vértices

Vértice é um objeto simples que pode ter nomes e outros atributos

Arestas

Arestas é uma conexão entre dois vértices





Modelagem de grafo

- No problema das casas
  - Vértices são casas e serviços
  - Arestas são as tubulações entre casas e serviços
- No problema da coloração de mapas
  - Vértices são estados
  - Arestas relacionam estados vizinhos
- No problema do caminho mais curto
  - Vértices são as cidades
  - Arestas são as ligações entre as cidades

Problemas interessantes

Problema das 4 cores

Qual a quantidade mínima de cores para colorir um mapa de tal forma que países fronteiriços possuam cores diferentes? Apresenta-se um exemplo em que 3 cores não são suficientes. Uma prova de que 5 cores é suficiente foi formulada. Conjecturou-se então que 4 cores seriam suficientes. Esta questão ficou em aberto até 1976 quando Appel e Haken provaram para 4 cores

Problema do ciclo Hamiltoniano (Hamilton 1859)

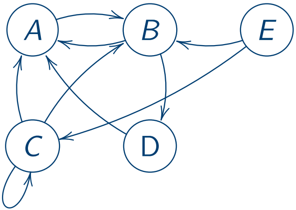
Existem n cidades. Cada par de cidades pode ser adjacente ou não arbitrariamente. Partindo de uma cidade qualquer, o problema consiste em determinar um trajeto que passe exatamente uma vez em cada cidade e retorne ao ponto de partida.

Teoria das árvores

problemas de circuitos elétricos e Química Orgânica

Grafo direcionado

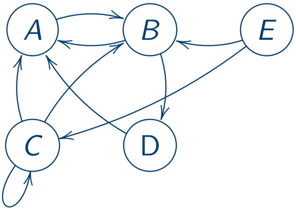
Par G=(V,E), onde V é um conjunto finito e E é uma relação binária em V.



Conceitos

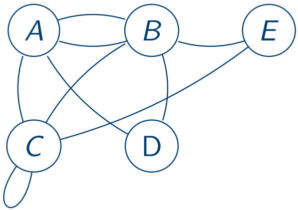
Grafo direcionado

Par G=(V,E), onde V é um conjunto finito e E é uma relação binária em V.



Grafo não direcionado

Par G=(V,E) onde o conjunto de arestas E consiste em pares de vértices não orientados. A aresta (v_i, v_j) e (v_j, v_i) são consideradas a mesma aresta.



Teoria dos Grafos e Computabilidade

— Terminologia —

Silvio Jamil F. Guimarães

Graduate Program in Informatics – PPGINF
Laboratory of Image and Multimedia Data Science – IMScience
Pontifical Catholic University of Minas Gerais – PUC Minas

Terminologia

Loop

uma aresta associada ao par de vértices (v_i, v_i)

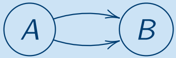
Terminologia

Loop

uma aresta associada ao par de vértices (v_i, v_i)

Arestas paralelas

quando mais de uma aresta está associada ao mesmo par de vértices



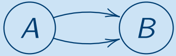
Terminologia

Loop

uma aresta associada ao par de vértices (v_i, v_i)

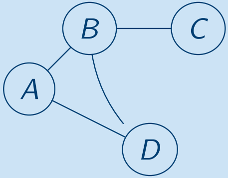
Arestas paralelas

quando mais de uma aresta está associada ao mesmo par de vértices



Grafo simples

um grafo que não possui loops e nem arestas paralelas



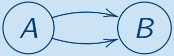
Terminologia

Loop

uma aresta associada ao par de vértices (v_i, v_i)

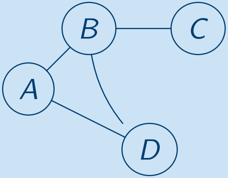
Arestas paralelas

quando mais de uma aresta está associada ao mesmo par de vértices



Grafo simples

um grafo que não possui loops e nem arestas paralelas



Vértices adjacentes

Dois vértices são ditos adjacentes se eles são pontos finais de uma mesma aresta

Terminologia

Grau de um vértice

- Grafo não direcionado:
  - grau $d(v)$ - número de arestas que incidem em v.
- Grafo direcionado:
  - grau de entrada $d^{-}(v)$- número de arestas que chegam em v
  - grau de saída $d^{+}(v)$- número de arestas que saem em v

Terminologia

Grau de um vértice

- Grafo não direcionado:
  - grau $d(v)$ - número de arestas que incidem em v.
- Grafo direcionado:
  - grau de entrada $d^{-}(v)$- número de arestas que chegam em v
  - grau de saída $d^{+}(v)$- número de arestas que saem em v

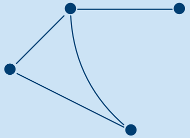
Um laço conta duas vezes para o grau de um vértice

Terminologia

Grau de um vértice

- Grafo não direcionado:
  - grau $d(v)$ - número de arestas que incidem em v.
- Grafo direcionado:
  - grau de entrada $d^{-}(v)$- número de arestas que chegam em v
  - grau de saída $d^{+}(v)$- número de arestas que saem em v

Um laço conta duas vezes para o grau de um vértice

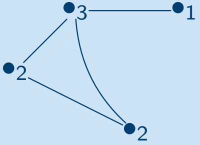


Terminologia

Grau de um vértice

- Grafo não direcionado:
  - grau $d(v)$ - número de arestas que incidem em v.
- Grafo direcionado:
  - grau de entrada $d^{-}(v)$- número de arestas que chegam em v
  - grau de saída $d^{+}(v)$- número de arestas que saem em v

Um laço conta duas vezes para o grau de um vértice

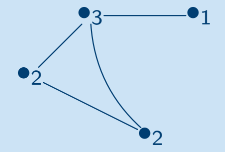


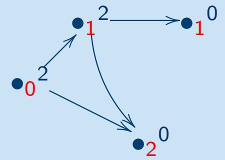
Terminologia

Grau de um vértice

- Grafo não direcionado:
  - grau $d(v)$ - número de arestas que incidem em v.
- Grafo direcionado:
  - grau de entrada $d^{-}(v)$- número de arestas que chegam em v
  - grau de saída $d^{+}(v)$- número de arestas que saem em v

Um laço conta duas vezes para o grau de um vértice



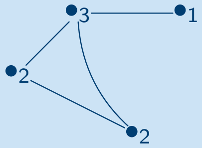


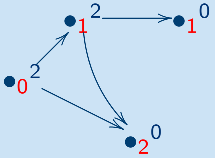
Terminologia

Grau de um vértice

- Grafo não direcionado:
  - grau $d(v)$ - número de arestas que incidem em v.
- Grafo direcionado:
  - grau de entrada $d^{-}(v)$- número de arestas que chegam em v
  - grau de saída $d^{+}(v)$- número de arestas que saem em v

Um laço conta duas vezes para o grau de um vértice



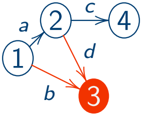


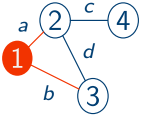
Seqüência de graus

Consiste em escrever em ordem crescente o grau de todos os seus vértices

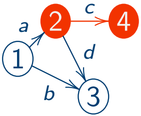
Terminologia

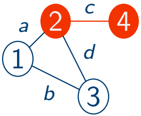
- Duas arestas não paralelas são **adjacentes** se elas são incidentes a um vértice comum



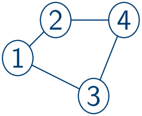


- Quando um vértice v_i é o vértice final de alguma aresta e_j, v_i e e_j são **incidentes**



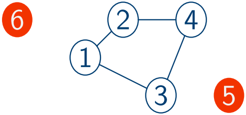


- Um grafo no qual todos os vértices possuem o mesmo grau é chamado de **grafo regular**.

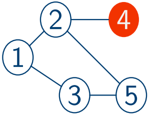


Terminologia

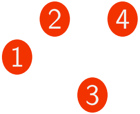
- Um vértice com nenhuma aresta incidente é chamado de **vértice isolado**.



- Um vértice com grau 1 é chamado de **vértice pendente**



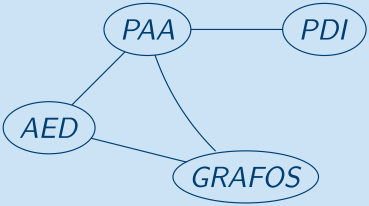
- Um grafo sem nenhuma aresta é chamado de **grafo nulo**. Todos os vértices em um grafo nulo são vértices isolados



Grafos valorado e rotulado

Grafo rotulado

Um grafo G(V,A) é dito ser rotulado em vértices (ou arestas) quando a cada vértice (ou aresta) estiver associado um rótulo



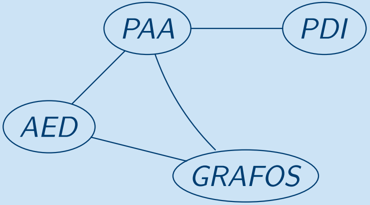
Grafos valorado e rotulado

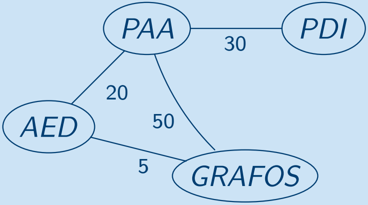
Grafo rotulado

Um grafo G(V,A) é dito ser rotulado em vértices (ou arestas) quando a cada vértice (ou aresta) estiver associado um rótulo

Grafo valorado

Um grafo G(V,A) é dito ser valorado quando existe uma ou mais funções relacionando V e/ou A com um conjunto de números.





Terminologia

Grafo completo

Um grafo G=(V,E) é completo se para cada par de vértices v_i e v_j existe uma aresta entre v_i e v_j. Em um grafo completo quaisquer dois vértices distintos são adjacentes (K_n)

Arestas no grafo completo

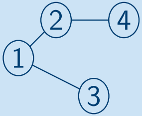
Seja K_n um grafo completo com n vértices. O número de arestas é :

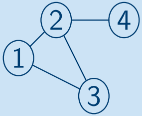
$$|E| = \frac{(n-1) \times \ n}{2}$$

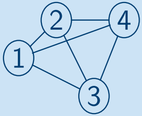
Terminologia

Grafo conexo

Existe pelo menos um caminho entre todos os pares de vértices

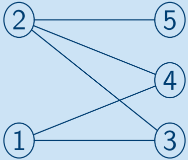


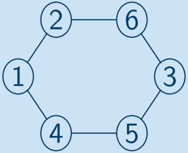




Grafo bipartido

Um grafo é dito ser bipartido quando seu conjunto de vértices V puder ser particionado em dois subconjuntos V_1 e V_2, tais que toda aresta de G une um vértice de V_1 a outro de V_2.

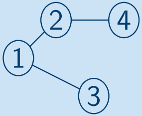


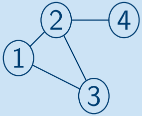


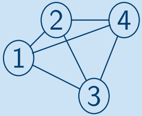
Terminologia

Grafo conexo

Existe pelo menos um caminho entre todos os pares de vértices

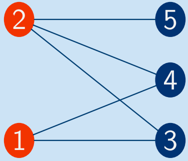


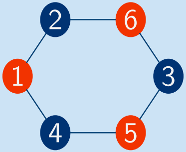




Grafo bipartido

Um grafo é dito ser bipartido quando seu conjunto de vértices V puder ser particionado em dois subconjuntos V_1 e V_2, tais que toda aresta de G une um vértice de V_1 a outro de V_2.





Grafo bipartido completo

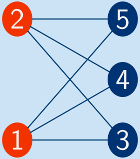
Grafo bipartido completo

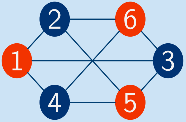
Um grafo é dito ser bipartido quando seu conjunto de vértices V puder ser particionado em dois subconjuntos V_1 e V_2, tais que toda aresta de G une um vértice de V_1 a outro de V_2, e que todo vértice de V_1 é adjacente a todo vértice de V_2.

Grafo bipartido completo

Grafo bipartido completo

Um grafo é dito ser bipartido quando seu conjunto de vértices V puder ser particionado em dois subconjuntos V_1 e V_2, tais que toda aresta de G une um vértice de V_1 a outro de V_2, e que todo vértice de V_1 é adjacente a todo vértice de V_2.





Arestas no grafo bipartido completo

Seja K_{mn} um grafo bipartido completo com n vértices em V_1 e m vértices em V_2. O número de arestas é :

$$|E| = n \times \ m$$

Propriedade de grau

Grau par

O número de arestas incidentes a um vértice v_i é chamado de grau, $d(v_i)$, do vértice i. A **soma** dos graus de todos os vértices de um grafo G é duas vezes o número de arestas de G.

$$\sum_{i=1}^{n} d(v_i) = 2e$$

Propriedade de grau

Grau par

O número de arestas incidentes a um vértice v_i é chamado de grau, $d(v_i)$, do vértice i. A **soma** dos graus de todos os vértices de um grafo G é duas vezes o número de arestas de G.

$$\sum_{i=1}^{n} d(v_i) = 2e$$

TEOREMA: Vértice de grau ímpar

O número de vértices de grau ímpar em um grafo é par

$$\sum_{i=1}^{n} d(v_i) = \sum_{d(v_j) par} d(v_j) + \sum_{d(v_k) impar} d(v_k)$$

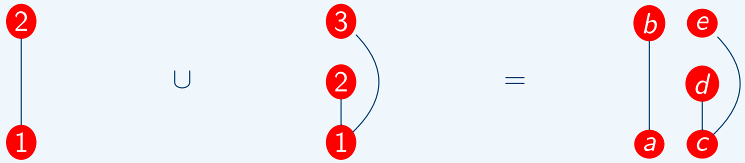
Operações sobre grafos

União

Seja $G_1 = (V_1, A_1)$ e $G_2 = (V_2, A_2)$ dois grafos. O grafo $G = G_1 \cup G_2$ é formado pelo grafo com conjunto de vértices $V_1 \cup V_2$ e conjunto de arestas $E_1 \cup E_2$.

União

Seja $G_1 = (V_1, A_1)$ e $G_2 = (V_2, A_2)$ dois grafos. O grafo $G = G_1 \cup G_2$ é formado pelo grafo com conjunto de vértices $V_1 \cup V_2$ e conjunto de arestas $E_1 \cup E_2$.



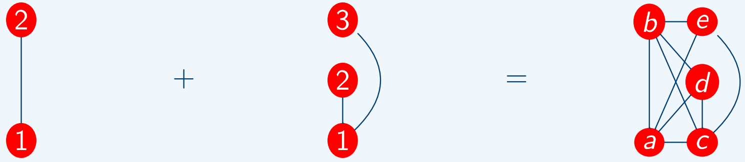
Operações sobre grafos

Soma

Seja $G_1 = (V_1, A_1)$ e $G_2 = (V_2, A_2)$ dois grafos. O grafo $G = G_1 + G_2$ é formado por $G_1 \cup G_2$ e de arestas ligando cada vértice de V_1 a V_2

Soma

Seja $G_1 = (V_1, A_1)$ e $G_2 = (V_2, A_2)$ dois grafos. O grafo $G = G_1 + G_2$ é formado por $G_1 \cup G_2$ e de arestas ligando cada vértice de V_1 a V_2



Propriedades de soma e união

Propriedades

- Podem ser aplicadas a qualquer número finito de grafos
- São operações associativas
- São operações comutativas

Grafos direcionados

Defina soma e união para grafos direcionados. As propridades de associação e comutação são mantidas?

Remoção de aresta e de vértice

Remoção de aresta

Se e é uma aresta de um grafo G, denota-se $G - e$ o grafo obtido de G pela remoção da aresta e. Se E é um conjunto de arestas em G, denota-se G-E ao grafo obtido pela remoção das arestas em E.

Remoção de vértice

Se v é um vértice de um grafo G denota-se por $G - v$ o grafo obtido de G pela remoção do vértice v conjuntamente com as arestas incidentes a v. Denota-se $G - S$ ao grafo obtido pela remoção dos vértices em S, sendo S um conjunto qualquer de vértices de G.

Contração de aresta/vértice

Denota-se por G/e o grafo obtido pela contração da aresta e. Remova $e = (v, w)$ de G e una suas extremidades v e w de tal forma que o vértice resultante seja incidente às arestas originalmente incidentes a v e w.

Matriz de incidência nó-arco

Seja um grafo $G = (V, A)$ em que $|V| = n$ e $|A| = m$. Uma matriz de incidência $A_{n \times m}$ nó-arco é representada por:

- Uma linha para cada nó
- Uma coluna para cada aresta

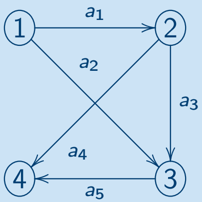
$$a = (i, j) \in A \Rightarrow \begin{bmatrix} 0 & +1 & 0 & -1 & 0 \end{bmatrix}^T$$

Matriz de incidência nó-arco

Seja um grafo $G = (V, A)$ em que $|V| = n$ e $|A| = m$. Uma matriz de incidência $A_{n \times m}$ nó-arco é representada por:

- Uma linha para cada nó
- Uma coluna para cada aresta

$$a = (i, j) \in A \Rightarrow \begin{bmatrix} 0 & +1 & 0 & -1 & 0 \end{bmatrix}^T$$



$$A_{n \times m} = \begin{bmatrix} +1 & +1 & 0 & 0 & 0 \\ -1 & 0 & +1 & +1 & 0 \\ 0 & -1 & -1 & 0 & +1 \\ 0 & 0 & 0 & -1 & -1 \end{bmatrix}$$

Matriz de adjacência

Seja um grafo $G = (V, A)$ em que $|V| = n$ e $|A| = m$. Uma matriz de adjacência $A_{n \times n}$ é representada por:

- Uma linha para cada nó
- Uma coluna para cada nó

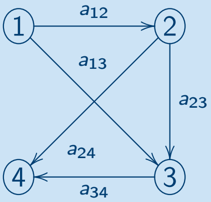
$$a_{ij} = \begin{cases} 1, & (i,j) \in A \\ 0, & (i,j) \notin A \end{cases}$$

Matriz de adjacência

Seja um grafo $G = (V, A)$ em que $|V| = n$ e $|A| = m$. Uma matriz de adjacência $A_{n \times n}$ é representada por:

- Uma linha para cada nó
- Uma coluna para cada nó

$$a_{ij} = \begin{cases} 1, & (i,j) \in A \\ 0, & (i,j) \notin A \end{cases}$$



$$A_{n \times n} = \begin{bmatrix} 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

Teoria dos Grafos e Computabilidade

— Lista de adjacência —

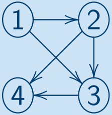
Silvio Jamil F. Guimarães

Graduate Program in Informatics – PPGINF
Laboratory of Image and Multimedia Data Science – IMScience
Pontifical Catholic University of Minas Gerais – PUC Minas

Feb 2025

Lista de adjacência

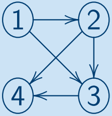
Seja um grafo $G = (V, A)$ em que $|V| = n$ e $|A| = m$. Uma lista de adjacência $A_{n \times n}$ é representada por uma lista de nós (ou vértices) em que cada nó aponta para a lista de seus sucessores (ou nós adjacentes).



$\Rightarrow$ n vértices + m arestas = $m + n$ elementos na lista

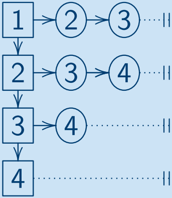
Lista de adjacência

Seja um grafo $G = (V, A)$ em que $|V| = n$ e $|A| = m$. Uma lista de adjacência $A_{n \times n}$ é representada por uma lista de nós (ou vértices) em que cada nó aponta para a lista de seus sucessores (ou nós adjacentes).

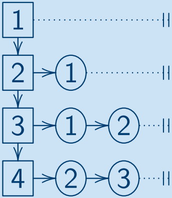


$\Rightarrow$ n vértices + m arestas = $m + n$ elementos na lista

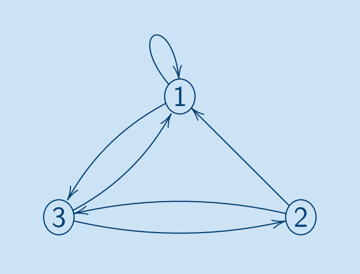
SUCESSORES



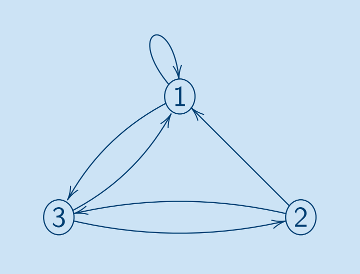
PREDECESSORES

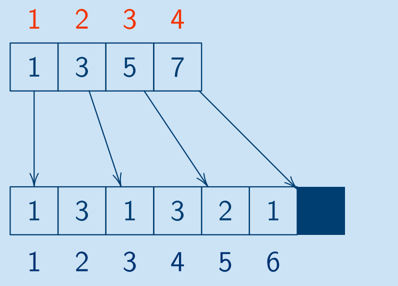


Lista de adjacência

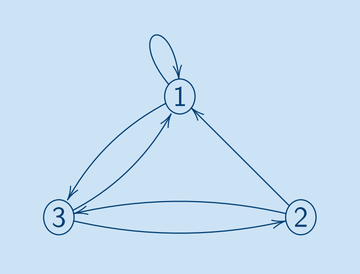


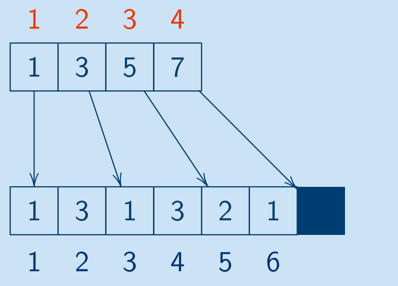
Lista de adjacência

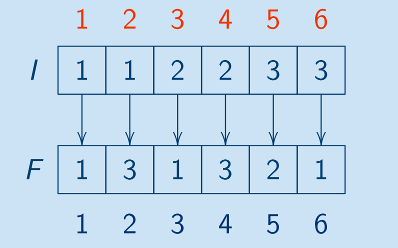




Lista de adjacência





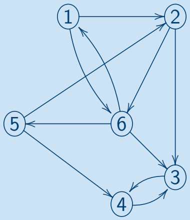


Monte o grafo a partir da representação

$$A_{n \times n} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 \end{bmatrix}$$

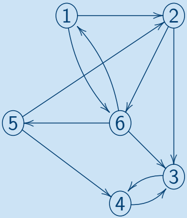
Monte o grafo a partir da representação

$$A_{n \times n} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 \end{bmatrix}$$



Monte o grafo a partir da representação

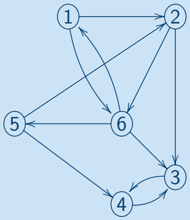
$$A_{n \times n} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 \end{bmatrix}$$



MATRIZ DE INCIDÊNCIA

Monte o grafo a partir da representação

$$A_{n \times n} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 \end{bmatrix}$$



MATRIZ DE INCIDÊNCIA

$$\begin{bmatrix} +1 & +1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 & +1 & 0 & +1 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 & -1 & 0 & 0 & +1 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & -1 & +1 \\ 0 & 0 & 0 & 0 & 0 & -1 & 0 & +1 & +1 & 0 & 0 \\ 0 & -1 & +1 & +1 & -1 & +1 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Teoria dos Grafos e Computabilidade

— Isomorphism —

Silvio Jamil F. Guimarães

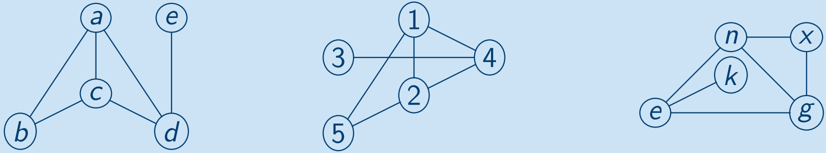
Graduate Program in Informatics – PPGINF
Laboratory of Image and Multimedia Data Science – IMScience
Pontifical Catholic University of Minas Gerais – PUC Minas

Feb 2025

Isomorfismo

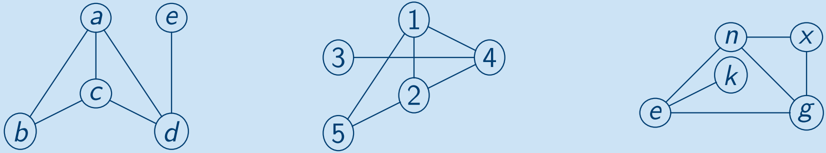
Dois grafos G e H são ditos **isomorfos** se existir uma correspondência um-para-um entre seus vértices e entre suas arestas, de maneira que as relações de incidência são preservadas

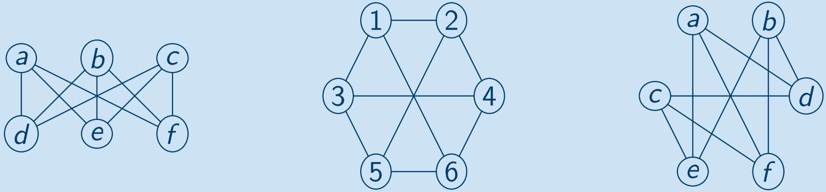
Dois grafos G e H são ditos **isomorfos** se existir uma correspondência um-para-um entre seus vértices e entre suas arestas, de maneira que as relações de incidência são preservadas



Isomorfismo

Dois grafos G e H são ditos **isomorfos** se existir uma correspondência um-para-um entre seus vértices e entre suas arestas, de maneira que as relações de incidência são preservadas





Isomorfismo

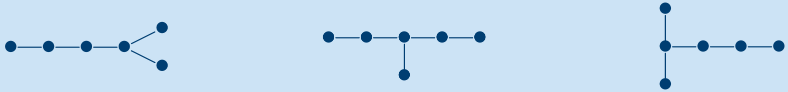
Condições necessárias mas não suficientes para que G e H sejam isomorfos:

- mesmo número de **vértices**
- mesmo número de **arestas**
- mesmo número de **componentes**
- mesmo número de **vértices com o mesmo grau**

Isomorfismo

Condições necessárias mas não suficientes para que G e H sejam isomorfos:

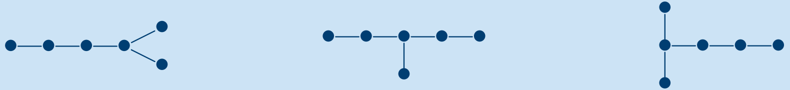
- mesmo número de **vértices**
- mesmo número de **arestas**
- mesmo número de **componentes**
- mesmo número de **vértices com o mesmo grau**



Isomorfismo

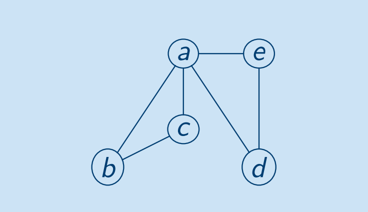
Condições necessárias mas não suficientes para que G e H sejam isomorfos:

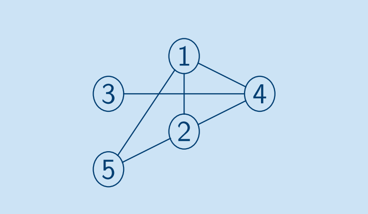
- mesmo número de **vértices**
- mesmo número de **arestas**
- mesmo número de **componentes**
- mesmo número de **vértices com o mesmo grau**



Não existe um algoritmo eficiente para determinar se dois grafos são isomorfos

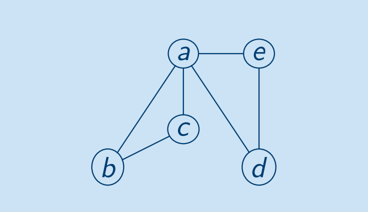
Some examples

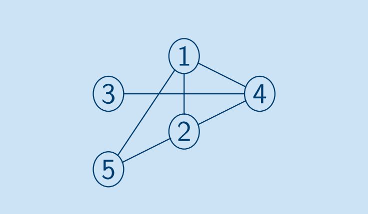




Are these two graphs Isomorphic?

Some examples

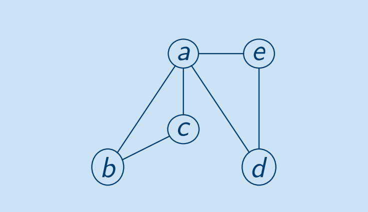




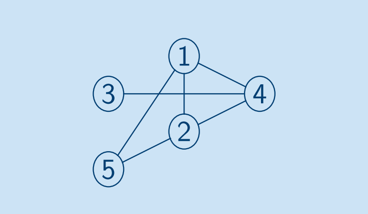
- vertices $\Longrightarrow$ 5

- vertices $\Longrightarrow$ 5

Some examples

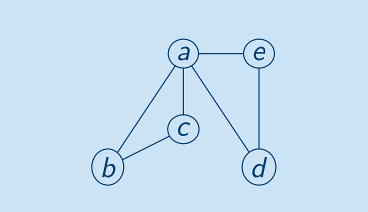


- vertices $\Longrightarrow$ 5
- edges $\Longrightarrow$ 6

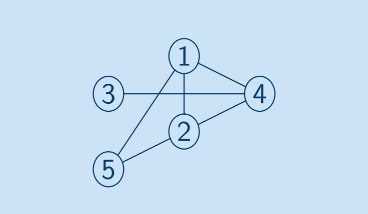


- vertices $\Longrightarrow$ 5
- edges $\Longrightarrow$ 6

Some examples

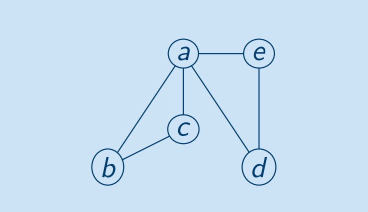


- vertices $\Longrightarrow$ 5
- edges $\Longrightarrow$ 6
- components $\Longrightarrow$ 1

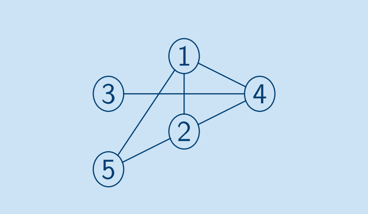


- vertices $\Longrightarrow$ 5
- edges $\Longrightarrow$ 6
- components $\Longrightarrow$ 1

Some examples

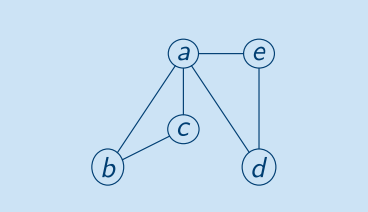


- vertices $\Longrightarrow$ 5
- edges $\Longrightarrow$ 6
- components $\Longrightarrow$ 1
- degrees $\Longrightarrow$ 2 2 2 3 4

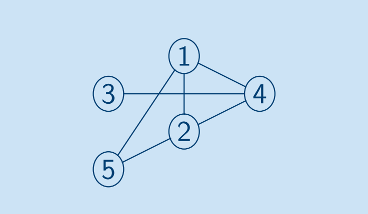


- vertices $\Longrightarrow$ 5
- edges $\Longrightarrow$ 6
- components $\Longrightarrow$ 1
- degrees $\Longrightarrow$ 1 2 3 3 3

Some examples

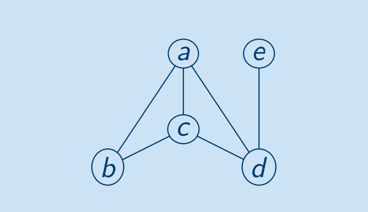


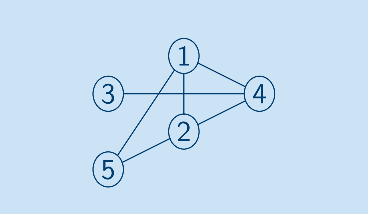
- vertices $\Longrightarrow$ 5
- edges $\Longrightarrow$ 6
- components $\Longrightarrow$ 1
- degrees $\Longrightarrow$ 2 2 2 3 4



- vertices $\Longrightarrow$ 5
- edges $\Longrightarrow$ 6
- components $\Longrightarrow$ 1
- degrees $\Longrightarrow$ 1 2 3 3 3

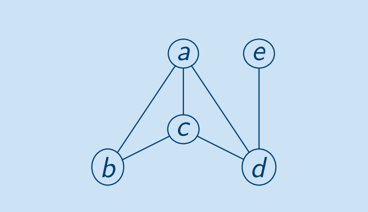
These two graphs are NOT Isomorphic



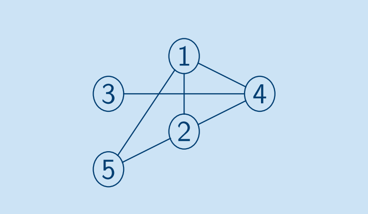


Are these two graphs Isomorphic?

Some examples

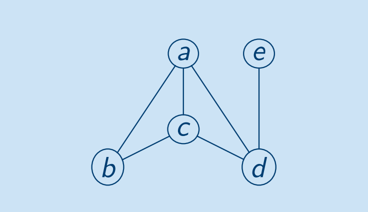


- vertices $\Longrightarrow$ 5

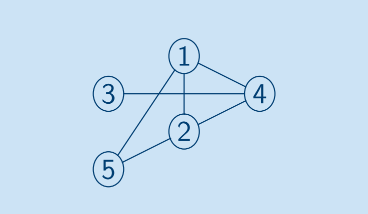


- vertices $\Longrightarrow$ 5

Some examples

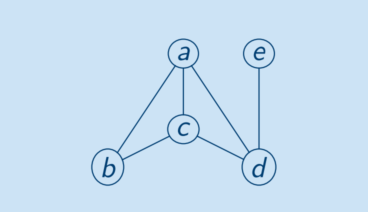


- vertices $\Longrightarrow$ 5
- edges $\Longrightarrow$ 6

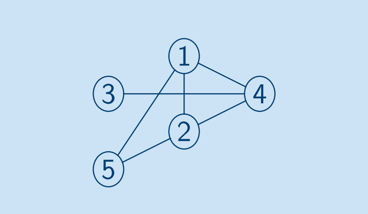


- vertices $\Longrightarrow$ 5
- edges $\Longrightarrow$ 6

Some examples

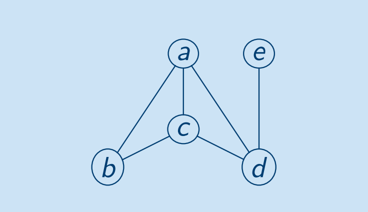


- vertices $\Longrightarrow$ 5
- edges $\Longrightarrow$ 6
- components $\Longrightarrow$ 1

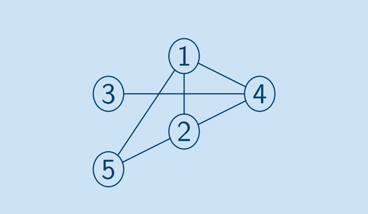


- vertices $\Longrightarrow$ 5
- edges $\Longrightarrow$ 6
- components $\Longrightarrow$ 1

Some examples

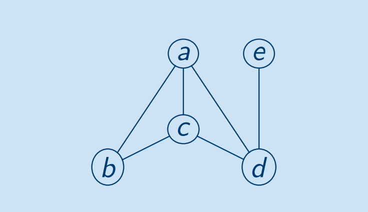


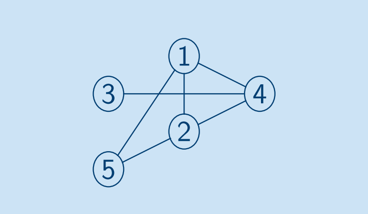
- vertices $\Longrightarrow$ 5
- edges $\Longrightarrow$ 6
- components $\Longrightarrow$ 1
- degrees $\Longrightarrow$ 1 2 3 3 3



- vertices $\Longrightarrow$ 5
- edges $\Longrightarrow$ 6
- components $\Longrightarrow$ 1
- degrees $\Longrightarrow$ 1 2 3 3 3

Some examples



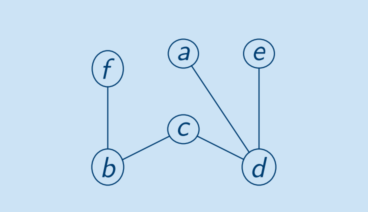


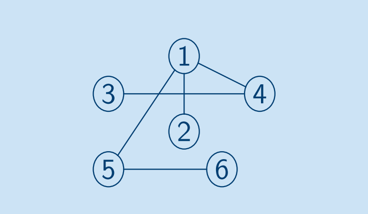
- vertices $\Longrightarrow$ 5
- edges $\Longrightarrow$ 6
- components $\Longrightarrow$ 1
- degrees $\Longrightarrow$ 1 2 3 3 3

- vertices $\Longrightarrow$ 5
- edges $\Longrightarrow$ 6
- components $\Longrightarrow$ 1
- degrees $\Longrightarrow$ 1 2 3 3 3

These two graphs are Isomorphic

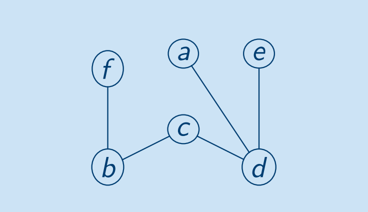
Some examples



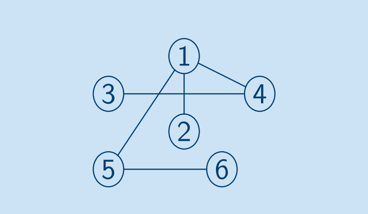


ARE THESE TWO GRAPHS ISOMORPHIC?

Some examples

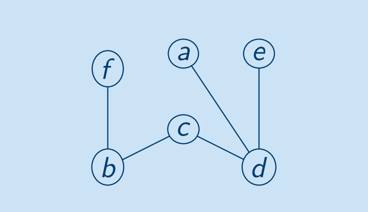


- vertices $\Longrightarrow$ 6

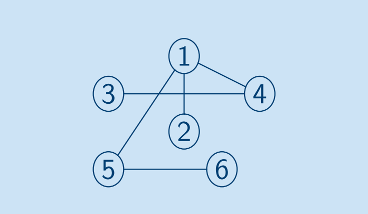


- vertices $\Longrightarrow$ 6

Some examples

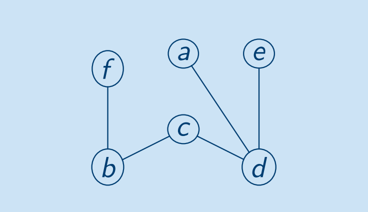


- vertices $\Longrightarrow$ 6
- edges $\Longrightarrow$ 5

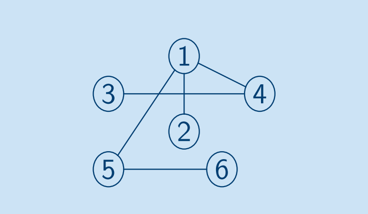


- vertices $\Longrightarrow$ 6
- edges $\Longrightarrow$ 5

Some examples

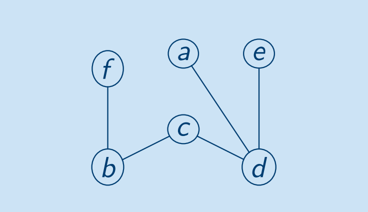


- vertices $\Longrightarrow$ 6
- edges $\Longrightarrow$ 5
- components $\Longrightarrow$ 1

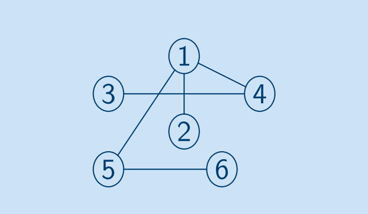


- vertices $\Longrightarrow$ 6
- edges $\Longrightarrow$ 5
- components $\Longrightarrow$ 1

Some examples

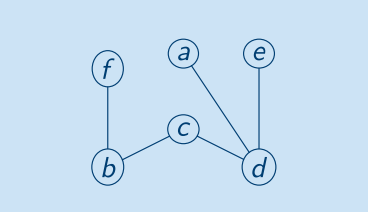


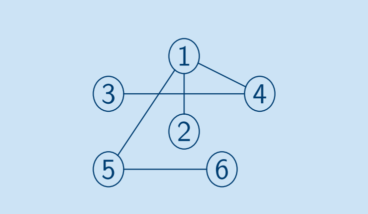
- vertices $\Longrightarrow$ 6
- edges $\Longrightarrow$ 5
- components $\Longrightarrow$ 1
- degrees $\Longrightarrow$ 1 1 1 2 3



- vertices $\Longrightarrow$ 6
- edges $\Longrightarrow$ 5
- components $\Longrightarrow$ 1
- degrees $\Longrightarrow$ 1 1 1 2 3

Some examples



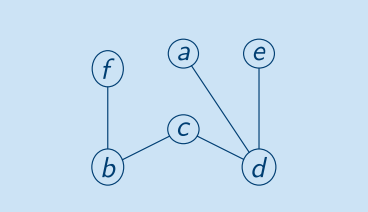


- vertices $\Longrightarrow$ 6
- edges $\Longrightarrow$ 5
- components $\Longrightarrow$ 1
- degrees $\Longrightarrow$ 1 1 1 2 3

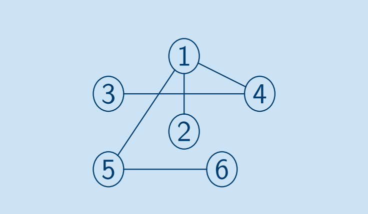
- vertices $\Longrightarrow$ 6
- edges $\Longrightarrow$ 5
- components $\Longrightarrow$ 1
- degrees $\Longrightarrow$ 1 1 1 2 3

THESE TWO GRAPHS ARE NOT ISOMORPHIC.

Some examples



- vertices $\Longrightarrow$ 6
- edges $\Longrightarrow$ 5
- components $\Longrightarrow$ 1
- degrees $\Longrightarrow$ 1 1 1 2 3

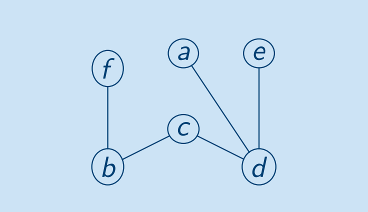


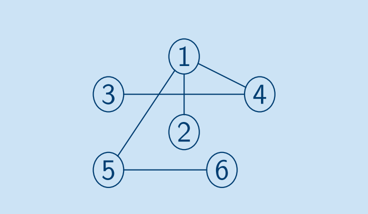
- vertices $\Longrightarrow$ 6
- edges $\Longrightarrow$ 5
- components $\Longrightarrow$ 1
- degrees $\Longrightarrow$ 1 1 1 2 3

THESE TWO GRAPHS ARE NOT ISOMORPHIC. WHY?

Some examples

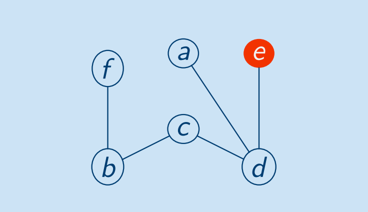
THE PROBLEM IS RELATED TO THE RELATIONSHIP BETWEEN THE VERTICES!!!

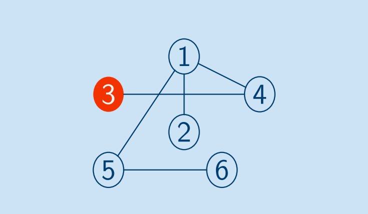




Some examples

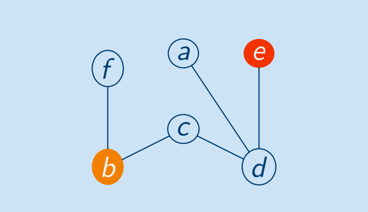
The problem is related to the relationship between the vertices!!!

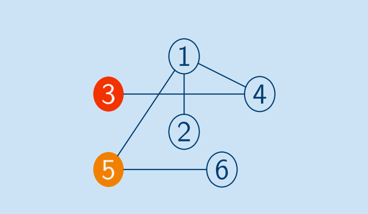




Some examples

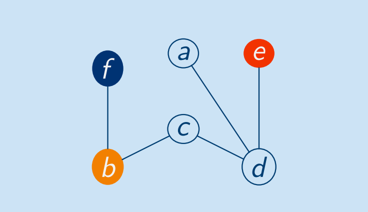
The problem is related to the relationship between the vertices!!!

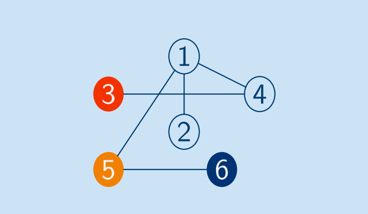




Some examples

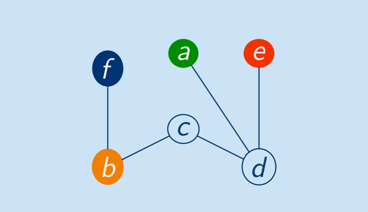
The problem is related to the relationship between the vertices!!!

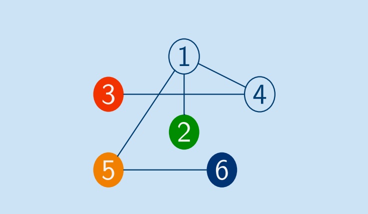




Some examples

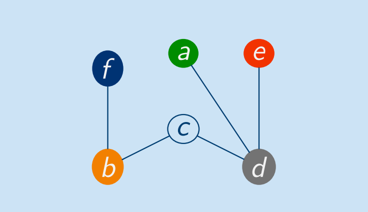
The problem is related to the relationship between the vertices!!!

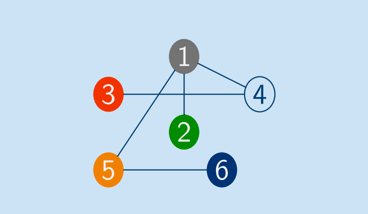




Some examples

THE PROBLEM IS RELATED TO THE RELATIONSHIP BETWEEN THE VERTICES!!!





The gray vertices (1 and d) are adjacent to vertices with different colors

Teoria dos Grafos e Computabilidade

— Important concepts —

Silvio Jamil F. Guimarães

Graduate Program in Informatics – PPGINF
Laboratory of Image and Multimedia Data Science – IMScience
Pontifical Catholic University of Minas Gerais – PUC Minas

Grafo complementar

Definição

- Seja $G = (V, E)$ um grafo simples dirigido ou não-dirigido
- O complemento de G, $C(G)$, é um grafo formado da seguinte maneira:
  - Os vértices de $C(G)$ são todos os vértices de G
  - As arestas de $C(G)$ são exatamente as arestas que faltam em G para formarmos um grafo completo

Grafo complementar

Definição

- Seja $G = (V, E)$ um grafo simples dirigido ou não-dirigido
- O complemento de G, $C(G)$, é um grafo formado da seguinte maneira:
  - Os vértices de $C(G)$ são todos os vértices de G
  - As arestas de $C(G)$ são exatamente as arestas que faltam em G para formarmos um grafo completo

Exercício

- Encontre um grafo com 5 vértices que seja isomorfo a seu complemento.

Grafo complementar

Definição

- Seja $G = (V, E)$ um grafo simples dirigido ou não-dirigido
- O complemento de G, $C(G)$, é um grafo formado da seguinte maneira:
  - Os vértices de $C(G)$ são todos os vértices de G
  - As arestas de $C(G)$ são exatamente as arestas que faltam em G para formarmos um grafo completo

Exercício

- Encontre um grafo com 5 vértices que seja isomorfo a seu complemento.
- Qual o número de arestas de um grafo que é isomorfo a seu complemento?

Sub-grafo

Sub-grafo

Um grafo $G_1 = (V_1, A_1)$ é dito ser **subgrafo** de um grafo $G(V, A)$ quando $V_1 \subset V$ e $A_1 \subset A$.

Sub-grafo induzido

Se $G_2 = (V_2, A_2)$ é um subgrafo de $G_1 = (V_1, A_1)$ e possui toda aresta (v, w) de G_1 tal que ambos, v e w, estejam em V_2, então G_2 é o **subgrafo induzido** pelo subconjunto de vértices V_2.

Sub-grafo

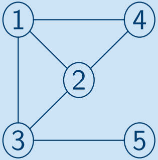
Sub-grafo

Um grafo $G_1 = (V_1, A_1)$ é dito ser **subgrafo** de um grafo $G(V, A)$ quando $V_1 \subset V$ e $A_1 \subset A$.

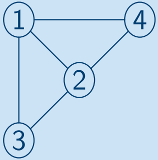
Sub-grafo induzido

Se $G_2 = (V_2, A_2)$ é um subgrafo de $G_1 = (V_1, A_1)$ e possui toda aresta (v, w) de G_1 tal que ambos, v e w, estejam em V_2, então G_2 é o **subgrafo induzido** pelo subconjunto de vértices V_2.

Exemplo

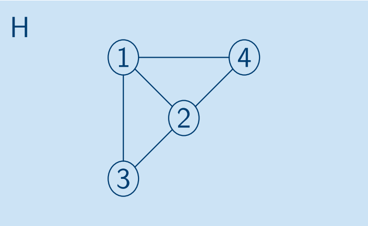


subgrafo
induzido por $\{1, 2, 3, 4\}$

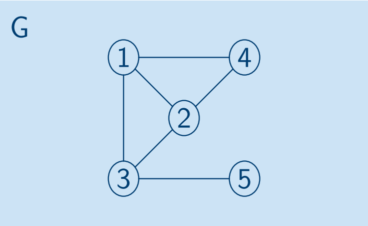


Sub-grafo

- Um grafo H é dito ser um subgrafo de um grafo G ($H \subseteq G$) se **todos** os **vértices** e todas as **arestas** de g estão em G

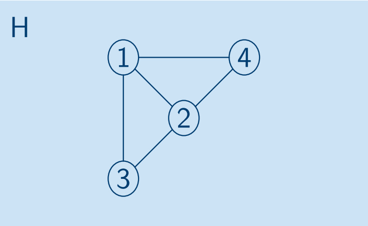


$\subseteq$

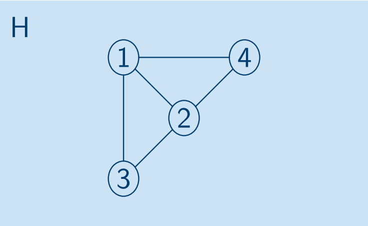


Sub-grafo

- Um grafo H é dito ser um subgrafo de um grafo G ($H \subseteq G$) se **todos** os **vértices** e todas as **arestas** de g estão em G
  - todo grafo é subgrafo de si próprio

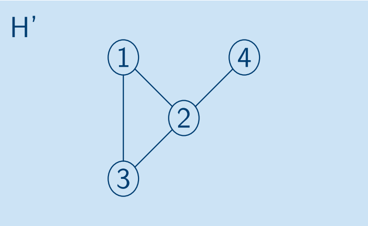


$\subseteq$

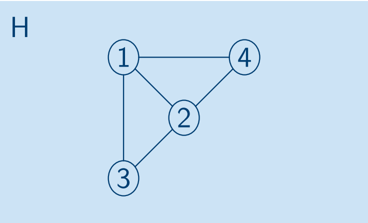


Sub-grafo

- Um grafo H é dito ser um subgrafo de um grafo G ($H \subseteq G$) se **todos** os **vértices** e todas as **arestas** de g estão em G
  - todo grafo é subgrafo de si próprio
  - o subgrafo de um subgrafo de G é subgrafo de G

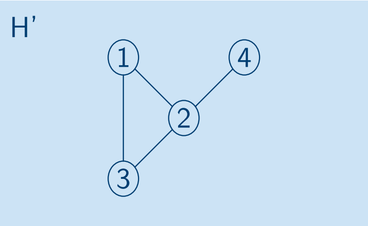


$\subseteq$

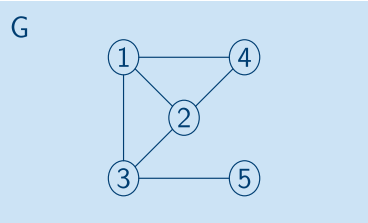


Sub-grafo

- Um grafo H é dito ser um subgrafo de um grafo G ($H \subseteq G$) se **todos** os **vértices** e todas as **arestas** de g estão em G
  - todo grafo é subgrafo de si próprio
  - o subgrafo de um subgrafo de G é subgrafo de G

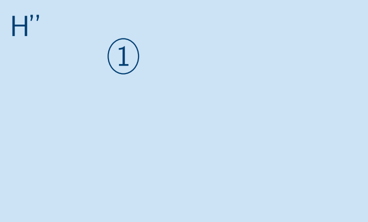


$\subseteq$

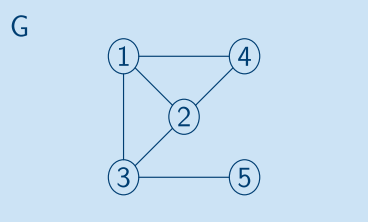


Sub-grafo

- Um grafo H é dito ser um subgrafo de um grafo G ($H \subseteq G$) se **todos** os **vértices** e todas as **arestas** de g estão em G
  - todo grafo é subgrafo de si próprio
  - o subgrafo de um subgrafo de G é subgrafo de G
  - um vértice simples de G é um subgrafo de G

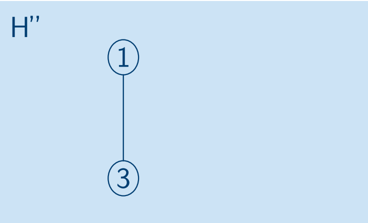


$\subseteq$

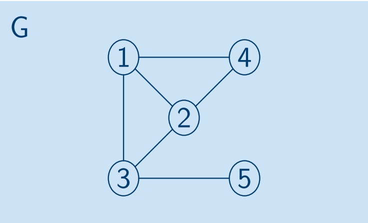


Sub-grafo

- Um grafo H é dito ser um subgrafo de um grafo G ($H \subseteq G$) se **todos** os **vértices** e todas as **arestas** de g estão em G
  - todo grafo é subgrafo de si próprio
  - o subgrafo de um subgrafo de G é subgrafo de G
  - um vértice simples de G é um subgrafo de G
  - uma aresta simples de G (juntamente com suas extremidades) é subgrafo de G

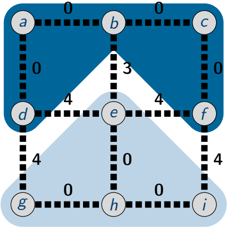


$\subseteq$



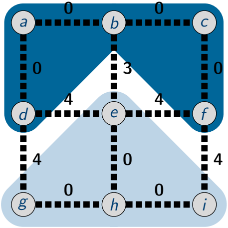
Sub-grafo

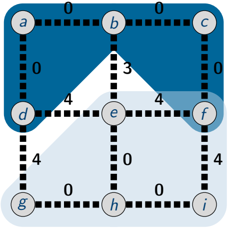
- Subgrafos disjuntos de arestas: dois (ou mais) subgrafos G_1 e G_2 de um grafo G são disjuntos de arestas se G_1 e G_2 não tiverem nenhuma aresta em comum.



Sub-grafo

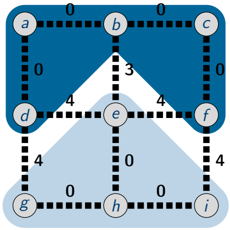
- Subgrafos disjuntos de arestas: dois (ou mais) subgrafos G_1 e G_2 de um grafo G são disjuntos de arestas se G_1 e G_2 não tiverem nenhuma aresta em comum.
  $\Rightarrow$ G_1 e G_2 podem ter vértices em comum?





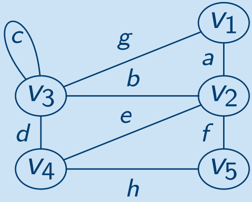
Sub-grafo

- Subgrafos disjuntos de arestas: dois (ou mais) subgrafos G_1 e G_2 de um grafo G são disjuntos de arestas se G_1 e G_2 não tiverem nenhuma aresta em comum.
- Subgrafos disjuntos de vértices: dois (ou mais) subgrafos G_1 e G_2 de um grafo G são disjuntos de vértices se G_1 e G_2 não tiverem nenhum vértice em comum.
  $\Rightarrow$ G_1 e G_2 podem ter arestas em comum?



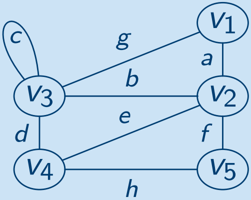
Caminhos e circuitos

- Seqüência de arestas: seqüência alternada de vértices e arestas começando e terminando com vértice. Cada aresta é incidente ao vértice que a precede e a antecede
  Ex.: $v_1\ a\ v_2\ a\ v_1\ g\ v_3$



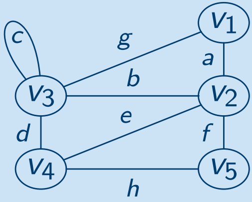
Caminhos e circuitos

- Seqüência de arestas: seqüência alternada de vértices e arestas começando e terminando com vértice. Cada aresta é incidente ao vértice que a precede e a antecede
  Ex.: $v_1\ a\ v_2\ a\ v_1\ g\ v_3$
- Caminho: seqüência de arestas no qual nenhuma aresta aparece mais de uma vez
  Ex.: $v_1\ a\ v_2\ b\ v_3\ c\ v_3\ d\ v_4\ e\ v_2\ f\ v_5$



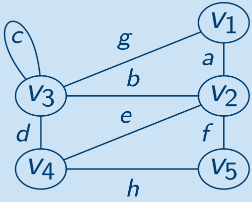
Caminhos e circuitos

- Seqüência de arestas: seqüência alternada de vértices e arestas começando e terminando com vértice. Cada aresta é incidente ao vértice que a precede e a antecede
  Ex.: $v_1\ a\ v_2\ a\ v_1\ g\ v_3$
- Caminho: seqüência de arestas no qual nenhuma aresta aparece mais de uma vez
  Ex.: $v_1\ a\ v_2\ b\ v_3\ c\ v_3\ d\ v_4\ e\ v_2\ f\ v_5$
  - Caminho aberto: vértice inicial é diferente do vértice final
    Ex.: $v_1\ a\ v_2\ b\ v_3\ c\ v_3$



Caminhos e circuitos

- Seqüência de arestas: seqüência alternada de vértices e arestas começando e terminando com vértice. Cada aresta é incidente ao vértice que a precede e a antecede
  Ex.: $v_1 \ a \ v_2 \ a \ v_1 \ g \ v_3$
- Caminho: seqüência de arestas no qual nenhuma aresta aparece mais de uma vez
  Ex.: $v_1 \ a \ v_2 \ b \ v_3 \ c \ v_3 \ d \ v_4 \ e \ v_2 \ f \ v_5$
  - Caminho aberto: vértice inicial é diferente do vértice final
    Ex.: $v_1 \ a \ v_2 \ b \ v_3 \ c \ v_3$
  - Caminho fechado: caminhos que começam e terminam no mesmo vértice
    Ex.: $v_1 \ a \ v_2 \ b \ v_3 \ c \ v_3 \ g \ v_1$

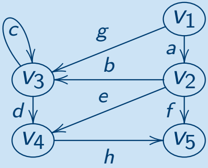


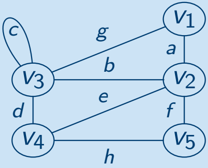
Cadeias

- Seja G um grafo dirigido e G´ o seu grafo não-dirigido associado
- Uma cadeia em G é um caminho em G´.

Cadeias

- Seja G um grafo dirigido e G´ o seu grafo não-dirigido associado
- Uma cadeia em G é um caminho em G´.





g-a-f é um caminho de G´ e uma cadeia em G

Caminhos e circuitos

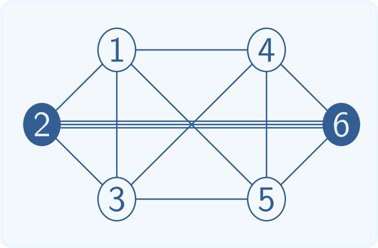
TEOREMA

Se um grafo possui exatamente 2 vértices de grau ímpar, existe um caminho entre esses dois vértices

Caminhos e circuitos

TEOREMA

Se um grafo possui exatamente 2 vértices de grau ímpar, existe um caminho entre esses dois vértices



2–6

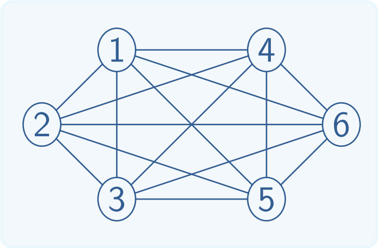
Caminhos e circuitos

TEOREMA

Um grafo simples com n vértices e k componentes possui no máximo $(n-k)(n-k+1)/2$ arestas

TEOREMA

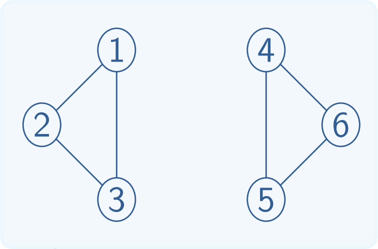
Um grafo simples com n vértices e k componentes possui no máximo $(n-k)(n-k+1)/2$ arestas



$$k = 1, n = 6 \Longrightarrow e = 15$$

TEOREMA

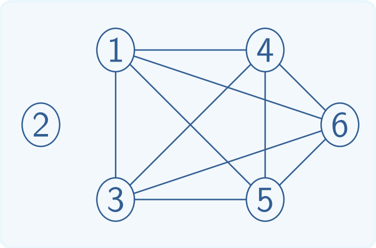
Um grafo simples com n vértices e k componentes possui no máximo $(n-k)(n-k+1)/2$ arestas



$$k = 2, n = 6 \Longrightarrow e = 6$$

TEOREMA

Um grafo simples com n vértices e k componentes possui no máximo $(n-k)(n-k+1)/2$ arestas



$$k = 2, n = 6 \Longrightarrow e = 10$$

TEOREMA

O número mínimo de arestas de um grafo simples com n vértices e k componentes é $n - k$

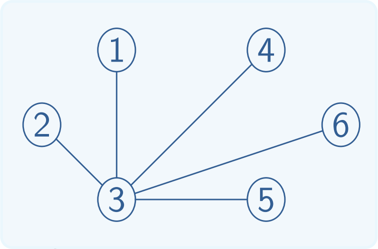
TEOREMA

O número mínimo de arestas de um grafo simples com n vértices e k componentes é $n - k$

$$k = 6, n = 6 \Longrightarrow e = 0$$

TEOREMA

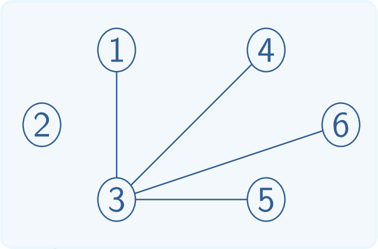
O número mínimo de arestas de um grafo simples com n vértices e k componentes é $n - k$



$$k = 1, n = 6 \Longrightarrow e = 5$$

TEOREMA

O número mínimo de arestas de um grafo simples com n vértices e k componentes é $n - k$



$k = 2, n = 6 \Longrightarrow e = 4$

Teoria dos Grafos e Computabilidade

— Depth-First search —

Silvio Jamil F. Guimarães

Graduate Program in Informatics – PPGINF
Laboratory of Image and Multimedia Data Science – IMScience
Pontifical Catholic University of Minas Gerais – PUC Minas

Caminhamento em grafos

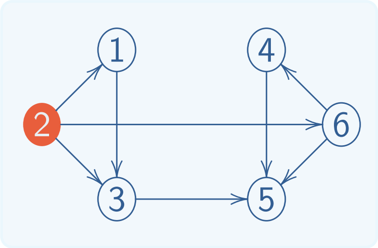
Busca em profundidade

Caminha no grafo visitando todos os seus vértices sempre procurando o **vértice mais profundo**.

Caminhamento em grafos

Busca em profundidade

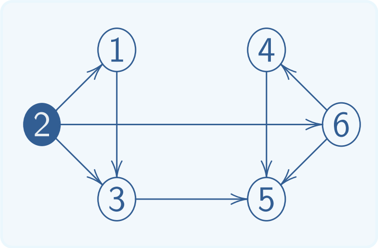
Caminha no grafo visitando todos os seus vértices sempre procurando o **vértice mais profundo**.



Caminhamento em grafos

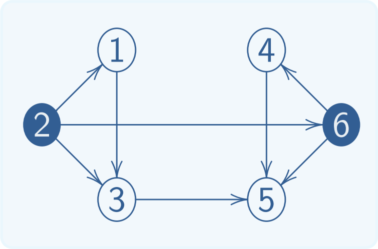
Busca em profundidade

Caminha no grafo visitando todos os seus vértices sempre procurando o **vértice mais profundo**.



Busca em profundidade

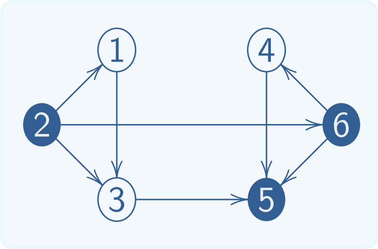
Caminha no grafo visitando todos os seus vértices sempre procurando o **vértice mais profundo**.



Caminhamento em grafos

Busca em profundidade

Caminha no grafo visitando todos os seus vértices sempre procurando o **vértice mais profundo**.



Caminhamento em grafos

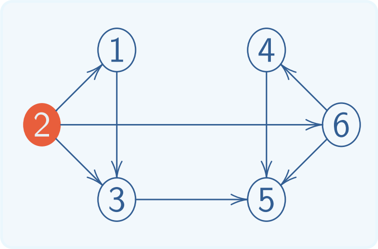
Busca em largura

Expandir o conjunto de vértices de forma uniforme em que são visitados todos os vértices de **mesma distância** ao início antes de visitar outros níveis.

Caminhamento em grafos

Busca em largura

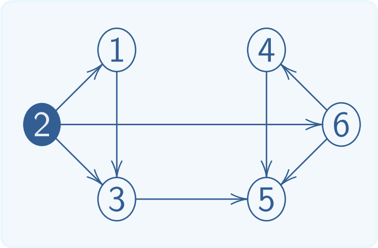
Expandir o conjunto de vértices de forma uniforme em que são visitados todos os vértices de **mesma distância** ao início antes de visitar outros níveis.



Caminhamento em grafos

Busca em largura

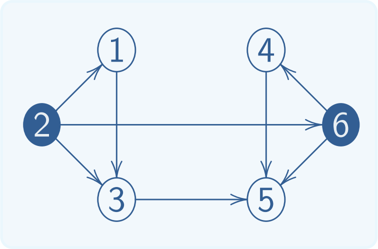
Expandir o conjunto de vértices de forma uniforme em que são visitados todos os vértices de **mesma distância** ao início antes de visitar outros níveis.



Caminhamento em grafos

Busca em largura

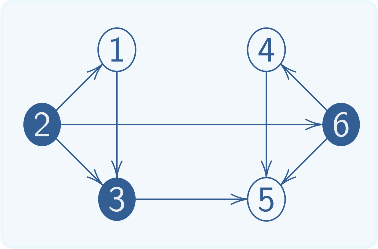
Expandir o conjunto de vértices de forma uniforme em que são visitados todos os vértices de **mesma distância** ao início antes de visitar outros níveis.

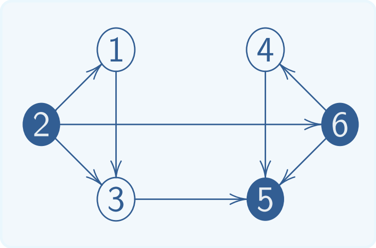


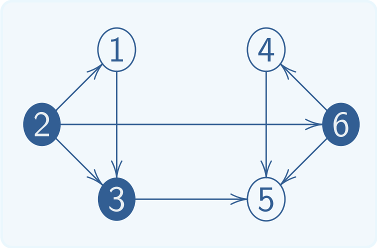
Caminhamento em grafos

Busca em largura

Expandir o conjunto de vértices de forma uniforme em que são visitados todos os vértices de **mesma distância** ao início antes de visitar outros níveis.



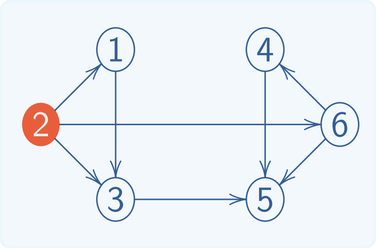

1
4
2
6
3
5


1
4
2
6
3
5

Busca em profundidade

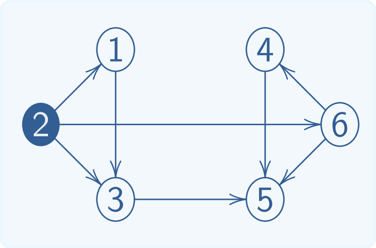
- As arestas são exploradas a partir do vértice v mais recentemente descoberto que ainda tem arestas não descobertas saindo dele;

- As arestas são exploradas a partir do vértice v mais recentemente descoberto que ainda tem arestas não descobertas saindo dele;

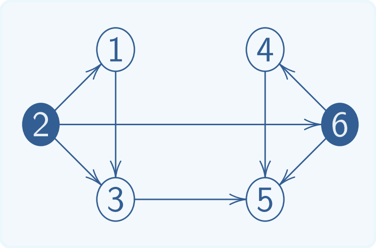


Busca em profundidade

- As arestas são exploradas a partir do vértice v mais recentemente descoberto que ainda tem arestas não descobertas saindo dele;

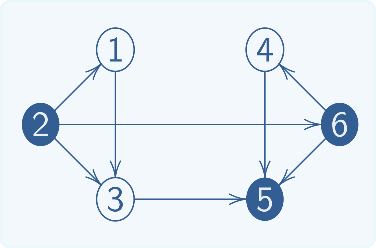


- As arestas são exploradas a partir do vértice v mais recentemente descoberto que ainda tem arestas não descobertas saindo dele;



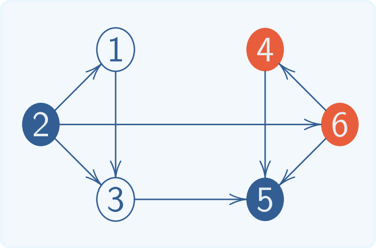
Busca em profundidade

- As arestas são exploradas a partir do vértice v mais recentemente descoberto que ainda tem arestas não descobertas saindo dele;



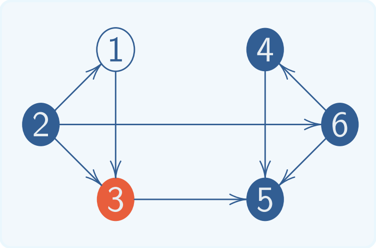
Busca em profundidade

- As arestas são exploradas a partir do vértice v mais recentemente descoberto que ainda tem arestas não descobertas saindo dele;
- Quando todas as arestas de v tiverem sido exploradas volta-se até para explorar arestas que saem do vértice a partir do qual v foi descoberto.



Busca em profundidade

- As arestas são exploradas a partir do vértice v mais recentemente descoberto que ainda tem arestas não descobertas saindo dele;
- Quando todas as arestas de v tiverem sido exploradas volta-se até para explorar arestas que saem do vértice a partir do qual v foi descoberto.



Busca em profundidade

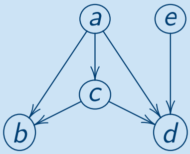
Algoritmo

- Todos os vértice são inicializados com branco
- Quando um vértice é visitado pela primeira vez ele torna-se azul
- Quando sua lista de adjacentes foi totalmente explorada ele torna-se vermelho

Busca em profundidade

Algoritmo

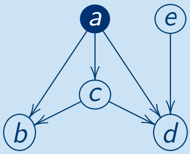
- Todos os vértice são inicializados com branco
- Quando um vértice é visitado pela primeira vez ele torna-se azul
- Quando sua lista de adjacentes foi totalmente explorada ele torna-se vermelho



Busca em profundidade

Algoritmo

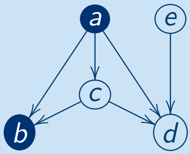
- Todos os vértice são inicializados com branco
- Quando um vértice é visitado pela primeira vez ele torna-se azul
- Quando sua lista de adjacentes foi totalmente explorada ele torna-se vermelho



Busca em profundidade

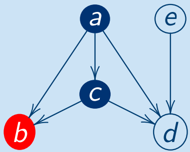
Algoritmo

- Todos os vértice são inicializados com branco
- Quando um vértice é visitado pela primeira vez ele torna-se azul
- Quando sua lista de adjacentes foi totalmente explorada ele torna-se vermelho



Algoritmo

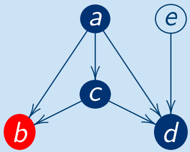
- Todos os vértice são inicializados com branco
- Quando um vértice é visitado pela primeira vez ele torna-se azul
- Quando sua lista de adjacentes foi totalmente explorada ele torna-se vermelho



Busca em profundidade

Algoritmo

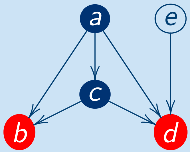
- Todos os vértice são inicializados com branco
- Quando um vértice é visitado pela primeira vez ele torna-se azul
- Quando sua lista de adjacentes foi totalmente explorada ele torna-se vermelho



Busca em profundidade

Algoritmo

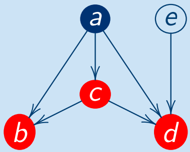
- Todos os vértice são inicializados com branco
- Quando um vértice é visitado pela primeira vez ele torna-se azul
- Quando sua lista de adjacentes foi totalmente explorada ele torna-se vermelho



Busca em profundidade

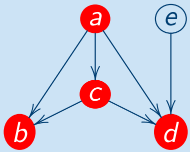
Algoritmo

- Todos os vértice são inicializados com branco
- Quando um vértice é visitado pela primeira vez ele torna-se azul
- Quando sua lista de adjacentes foi totalmente explorada ele torna-se vermelho



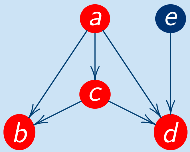
Algoritmo

- Todos os vértice são inicializados com branco
- Quando um vértice é visitado pela primeira vez ele torna-se azul
- Quando sua lista de adjacentes foi totalmente explorada ele torna-se vermelho



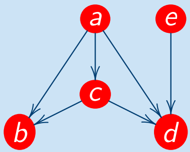
Algoritmo

- Todos os vértice são inicializados com branco
- Quando um vértice é visitado pela primeira vez ele torna-se azul
- Quando sua lista de adjacentes foi totalmente explorada ele torna-se vermelho



Algoritmo

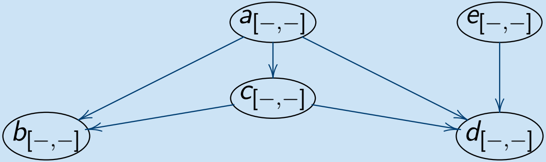
- Todos os vértice são inicializados com branco
- Quando um vértice é visitado pela primeira vez ele torna-se azul
- Quando sua lista de adjacentes foi totalmente explorada ele torna-se vermelho



Busca em profundidade

Tempos

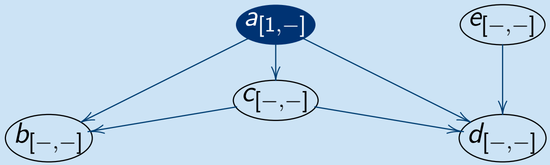
- O tempo de **descoberta** $d[v]$ é o momento em que o vértice v foi visitado pela primeira vez
- O tempo de **término** do exame da lista de adjacentes $t[v]$ é o momento em que a visita a toda lista de vértices adjacentes a v foi concluída.
- $d[v]$ e $t[v]$ são inteiros entre 1 e $2V$, onde V é o número de vértices do grafo



Busca em profundidade

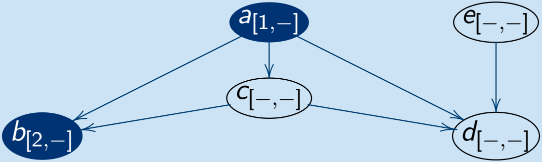
Tempos

- O tempo de **descoberta** $d[v]$ é o momento em que o vértice v foi visitado pela primeira vez
- O tempo de **término** do exame da lista de adjacentes $t[v]$ é o momento em que a visita a toda lista de vértices adjacentes a v foi concluída.
- $d[v]$ e $t[v]$ são inteiros entre 1 e $2V$, onde V é o número de vértices do grafo



Tempos

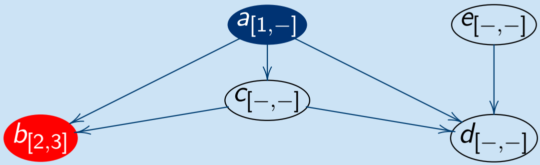
- O tempo de **descoberta** $d[v]$ é o momento em que o vértice v foi visitado pela primeira vez
- O tempo de **término** do exame da lista de adjacentes $t[v]$ é o momento em que a visita a toda lista de vértices adjacentes a v foi concluída.
- $d[v]$ e $t[v]$ são inteiros entre 1 e $2V$, onde V é o número de vértices do grafo



Busca em profundidade

Tempos

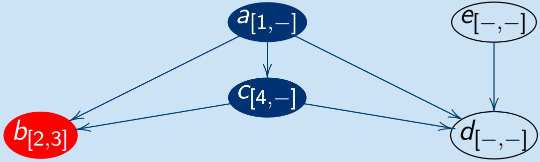
- O tempo de **descoberta** $d[v]$ é o momento em que o vértice v foi visitado pela primeira vez
- O tempo de **término** do exame da lista de adjacentes $t[v]$ é o momento em que a visita a toda lista de vértices adjacentes a v foi concluída.
- $d[v]$ e $t[v]$ são inteiros entre 1 e $2V$, onde V é o número de vértices do grafo



Busca em profundidade

Tempos

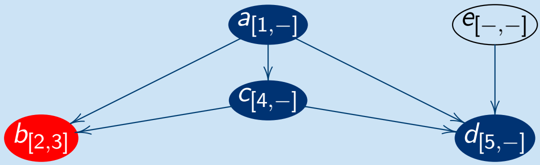
- O tempo de **descoberta** $d[v]$ é o momento em que o vértice v foi visitado pela primeira vez
- O tempo de **término** do exame da lista de adjacentes $t[v]$ é o momento em que a visita a toda lista de vértices adjacentes a v foi concluída.
- $d[v]$ e $t[v]$ são inteiros entre 1 e $2V$, onde V é o número de vértices do grafo



Busca em profundidade

Tempos

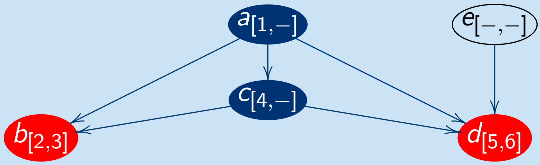
- O tempo de **descoberta** $d[v]$ é o momento em que o vértice v foi visitado pela primeira vez
- O tempo de **término** do exame da lista de adjacentes $t[v]$ é o momento em que a visita a toda lista de vértices adjacentes a v foi concluída.
- $d[v]$ e $t[v]$ são inteiros entre 1 e $2V$, onde V é o número de vértices do grafo



Busca em profundidade

Tempos

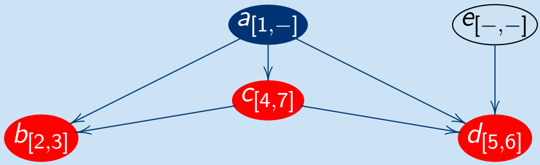
- O tempo de **descoberta** $d[v]$ é o momento em que o vértice v foi visitado pela primeira vez
- O tempo de **término** do exame da lista de adjacentes $t[v]$ é o momento em que a visita a toda lista de vértices adjacentes a v foi concluída.
- $d[v]$ e $t[v]$ são inteiros entre 1 e $2V$, onde V é o número de vértices do grafo



Busca em profundidade

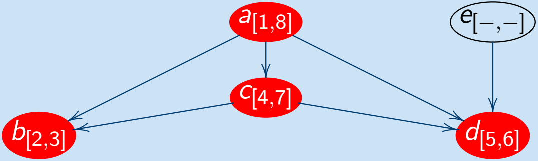
Tempos

- O tempo de **descoberta** $d[v]$ é o momento em que o vértice v foi visitado pela primeira vez
- O tempo de **término** do exame da lista de adjacentes $t[v]$ é o momento em que a visita a toda lista de vértices adjacentes a v foi concluída.
- $d[v]$ e $t[v]$ são inteiros entre 1 e $2V$, onde V é o número de vértices do grafo



Tempos

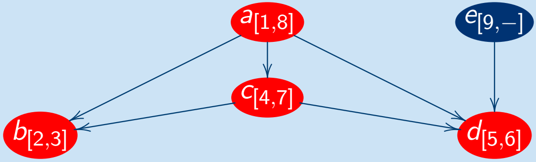
- O tempo de **descoberta** $d[v]$ é o momento em que o vértice v foi visitado pela primeira vez
- O tempo de **término** do exame da lista de adjacentes $t[v]$ é o momento em que a visita a toda lista de vértices adjacentes a v foi concluída.
- $d[v]$ e $t[v]$ são inteiros entre 1 e $2V$, onde V é o número de vértices do grafo



Busca em profundidade

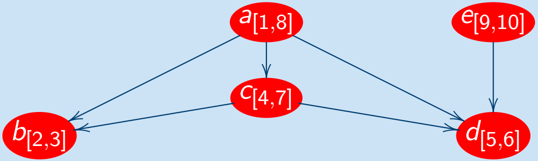
Tempos

- O tempo de **descoberta** $d[v]$ é o momento em que o vértice v foi visitado pela primeira vez
- O tempo de **término** do exame da lista de adjacentes $t[v]$ é o momento em que a visita a toda lista de vértices adjacentes a v foi concluída.
- $d[v]$ e $t[v]$ são inteiros entre 1 e $2V$, onde V é o número de vértices do grafo



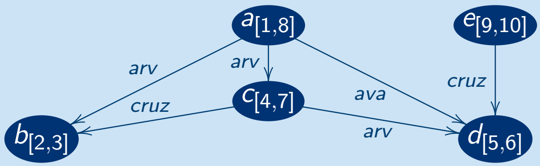
Tempos

- O tempo de **descoberta** $d[v]$ é o momento em que o vértice v foi visitado pela primeira vez
- O tempo de **término** do exame da lista de adjacentes $t[v]$ é o momento em que a visita a toda lista de vértices adjacentes a v foi concluída.
- $d[v]$ e $t[v]$ são inteiros entre 1 e $2V$, onde V é o número de vértices do grafo



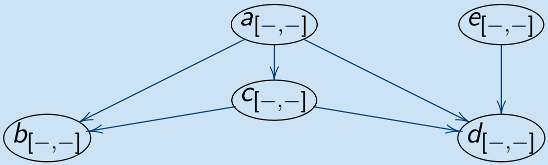
Classificação das arestas

- De árvore: uma aresta (u,v) é de **árvore** se o vértice v foi visitado a primeira vez passando pela aresta (u,v)
- De retorno: uma aresta (u,v) é uma aresta de **retorno** se esta conecta um vértice u com um predecessor v já presente em uma árvore de busca
- De avanço: Não pertencem a árvore de busca em profundidade mas conectam um vértice a um **descendente** que pertence a árvore de busca
- De cruzamento: conectam vértice de uma **mesma árvore** de busca ou de árvores diferentes

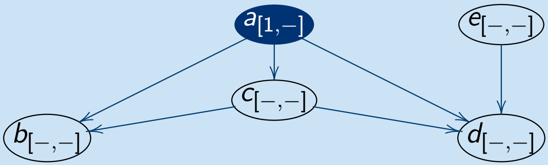


Busca em profundidade

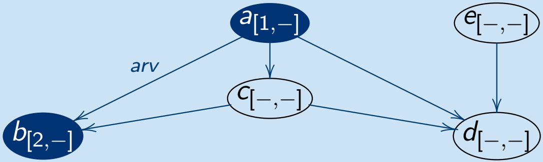
- As arestas $e = (u, v)$ podem ser classificadas pela cor do vértice v que é alcançado quando se passa pela aresta e primeira vez
  - Branco: aresta de árvore
  - Azul: aresta de retorno
  - Vermelho: (i) Se u é visitado antes de v então e é uma aresta de avanço; (ii) Se v é visitado antes de u então e é de cruzamento



- As arestas $e = (u, v)$ podem ser classificadas pela cor do vértice v que é alcançado quando se passa pela aresta e primeira vez
  - Branco : aresta de árvore
  - Azul : aresta de retorno
  - Vermelho : (i) Se u é visitado antes de v então e é uma aresta de avanço; (ii) Se v é visitado antes de u então e é de cruzamento

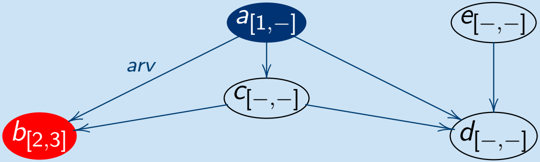


- As arestas $e = (u, v)$ podem ser classificadas pela cor do vértice v que é alcançado quando se passa pela aresta e primeira vez
  - Branco : aresta de árvore
  - Azul : aresta de retorno
  - Vermelho : (i) Se u é visitado antes de v então e é uma aresta de avanço; (ii) Se v é visitado antes de u então e é de cruzamento



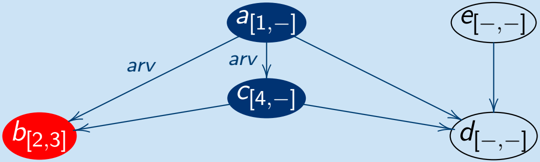
Busca em profundidade

- As arestas $e = (u, v)$ podem ser classificadas pela cor do vértice v que é alcançado quando se passa pela aresta e primeira vez
  - Branco : aresta de árvore
  - Azul : aresta de retorno
  - Vermelho : (i) Se u é visitado antes de v então e é uma aresta de avanço; (ii) Se v é visitado antes de u então e é de cruzamento

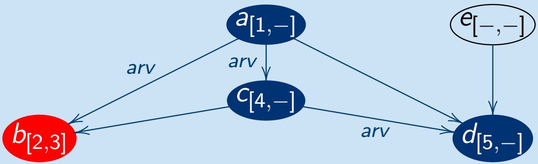


Busca em profundidade

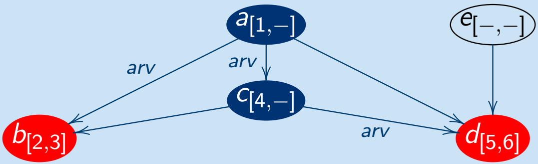
- As arestas $e = (u, v)$ podem ser classificadas pela cor do vértice v que é alcançado quando se passa pela aresta e primeira vez
  - Branco : aresta de árvore
  - Azul : aresta de retorno
  - Vermelho : (i) Se u é visitado antes de v então e é uma aresta de avanço; (ii) Se v é visitado antes de u então e é de cruzamento



- As arestas $e = (u, v)$ podem ser classificadas pela cor do vértice v que é alcançado quando se passa pela aresta e primeira vez
  - Branco : aresta de árvore
  - Azul : aresta de retorno
  - Vermelho : (i) Se u é visitado antes de v então e é uma aresta de avanço; (ii) Se v é visitado antes de u então e é de cruzamento

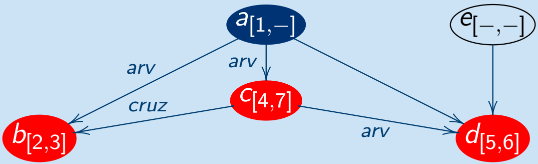


- As arestas $e = (u, v)$ podem ser classificadas pela cor do vértice v que é alcançado quando se passa pela aresta e primeira vez
  - Branco : aresta de árvore
  - Azul : aresta de retorno
  - Vermelho : (i) Se u é visitado antes de v então e é uma aresta de avanço; (ii) Se v é visitado antes de u então e é de cruzamento



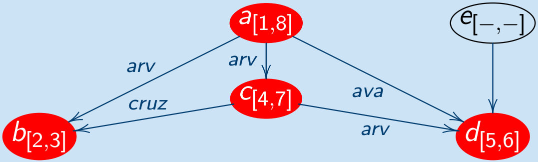
Busca em profundidade

- As arestas $e = (u, v)$ podem ser classificadas pela cor do vértice v que é alcançado quando se passa pela aresta e primeira vez
  - Branco : aresta de árvore
  - Azul : aresta de retorno
  - Vermelho : (i) Se u é visitado antes de v então e é uma aresta de avanço; (ii) Se v é visitado antes de u então e é de cruzamento



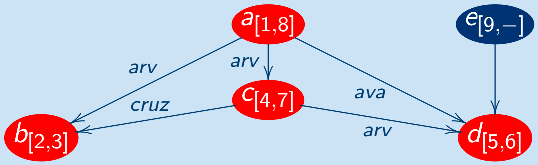
Busca em profundidade

- As arestas $e = (u, v)$ podem ser classificadas pela cor do vértice v que é alcançado quando se passa pela aresta e primeira vez
  - Branco : aresta de árvore
  - Azul : aresta de retorno
  - Vermelho : (i) Se u é visitado antes de v então e é uma aresta de avanço; (ii) Se v é visitado antes de u então e é de cruzamento

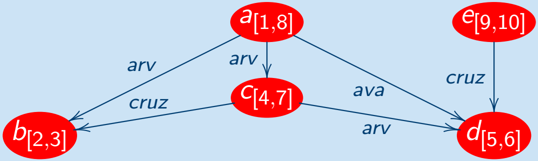


Busca em profundidade

- As arestas $e = (u, v)$ podem ser classificadas pela cor do vértice v que é alcançado quando se passa pela aresta e primeira vez
  - Branco: aresta de árvore
  - Azul: aresta de retorno
  - Vermelho: (i) Se u é visitado antes de v então e é uma aresta de avanço; (ii) Se v é visitado antes de u então e é de cruzamento



- As arestas $e = (u, v)$ podem ser classificadas pela cor do vértice v que é alcançado quando se passa pela aresta e primeira vez
  - Branco : aresta de árvore
  - Azul : aresta de retorno
  - Vermelho : (i) Se u é visitado antes de v então e é uma aresta de avanço; (ii) Se v é visitado antes de u então e é de cruzamento



Teste de circuito

- Se uma aresta de retorno é encontrada na busca em profundidade então o grafo possui um ciclo
- Um grafo é acíclico se e somente se na busca em profundidade não for encontrada nenhuma aresta de retorno

Teoria dos Grafos e Computabilidade

— Breadth-First search —

Silvio Jamil F. Guimarães

Graduate Program in Informatics – PPGINF
Laboratory of Image and Multimedia Data Science – IMScience
Pontifical Catholic University of Minas Gerais – PUC Minas

Feb 2025

Busca em largura

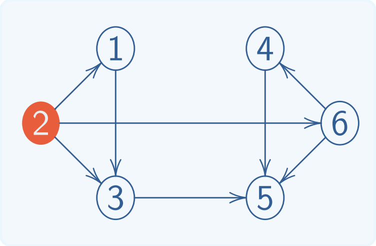
Busca em largura

- Expandir o conjunto de vértices de forma uniforme em que são visitados todos os vértices de **mesma distância** ao início antes de visitar outros níveis.

Busca em largura

Busca em largura

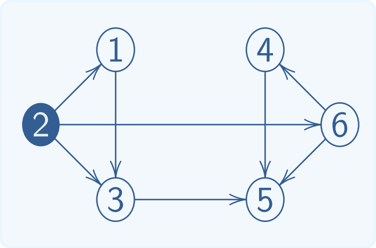
- Expandir o conjunto de vértices de forma uniforme em que são visitados todos os vértices de **mesma distância** ao início antes de visitar outros níveis.



Busca em largura

Busca em largura

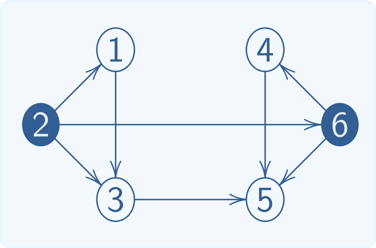
- Expandir o conjunto de vértices de forma uniforme em que são visitados todos os vértices de **mesma distância** ao início antes de visitar outros níveis.



Busca em largura

Busca em largura

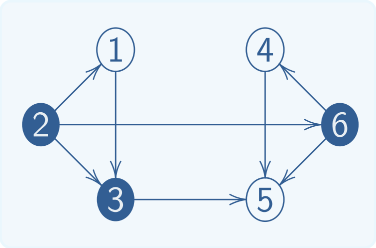
- Expandir o conjunto de vértices de forma uniforme em que são visitados todos os vértices de **mesma distância** ao início antes de visitar outros níveis.



Busca em largura

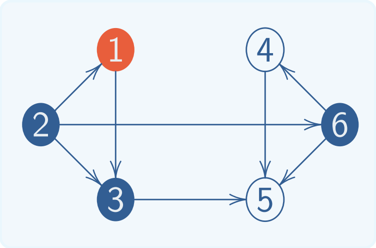
Busca em largura

- Expandir o conjunto de vértices de forma uniforme em que são visitados todos os vértices de **mesma distância** ao início antes de visitar outros níveis.



Busca em largura

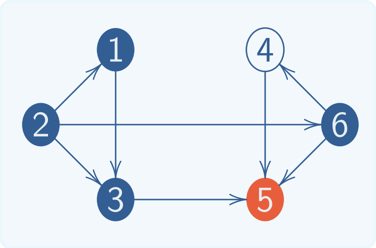
- Expandir o conjunto de vértices de forma uniforme em que são visitados todos os vértices de **mesma distância** ao início antes de visitar outros níveis.



Busca em largura

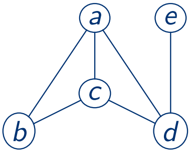
Busca em largura

- Expandir o conjunto de vértices de forma uniforme em que são visitados todos os vértices de **mesma distância** ao início antes de visitar outros níveis.
- Na busca em largura o algoritmo descobre todos os vertices a uma distância k do vértice de origem antes de descobrir os que estão a uma distância k+1



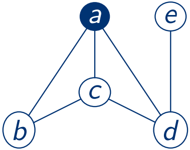
Busca em largura

- Cada vértice é colorido de branco, azul ou vermelho
- Todos os vértices são inicializados com branco
- Quando um vértice é descoberto pela primeira vez ele torna-se azul
- Vértices cujos adjacentes são todos descobertos tornam-se vermelhos
- Se $(u, v) \in A$ e o vértice u é vermelho, entao v tem quer ser azul ou vermelho
- Vértices azul podem ter adjacentes brancos.



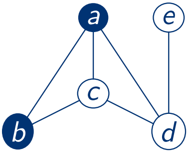
Busca em largura

- Cada vértice é colorido de branco, azul ou vermelho
- Todos os vértices são inicializados com branco
- Quando um vértice é descoberto pela primeira vez ele torna-se azul
- Vértices cujos adjacentes são todos descobertos tornam-se vermelhos
- Se $(u, v) \in A$ e o vértice u é vermelho, entao v tem quer ser azul ou vermelho
- Vértices azul podem ter adjacentes brancos.



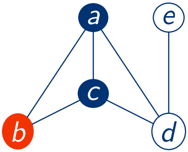
Busca em largura

- Cada vértice é colorido de branco, azul ou vermelho
- Todos os vértices são inicializados com branco
- Quando um vértice é descoberto pela primeira vez ele torna-se azul
- Vértices cujos adjacentes são todos descobertos tornam-se vermelhos
- Se $(u, v) \in A$ e o vértice u é vermelho, entao v tem quer ser azul ou vermelho
- Vértices azul podem ter adjacentes brancos.



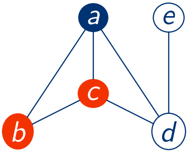
Busca em largura

- Cada vértice é colorido de branco, azul ou vermelho
- Todos os vértices são inicializados com branco
- Quando um vértice é descoberto pela primeira vez ele torna-se azul
- Vértices cujos adjacentes são todos descobertos tornam-se vermelhos
- Se $(u, v) \in A$ e o vértice u é vermelho, entao v tem quer ser azul ou vermelho
- Vértices azul podem ter adjacentes brancos.



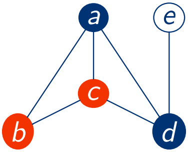
Busca em largura

- Cada vértice é colorido de branco, azul ou vermelho
- Todos os vértices são inicializados com branco
- Quando um vértice é descoberto pela primeira vez ele torna-se azul
- Vértices cujos adjacentes são todos descobertos tornam-se vermelhos
- Se $(u, v) \in A$ e o vértice u é vermelho, entao v tem quer ser azul ou vermelho
- Vértices azul podem ter adjacentes brancos.



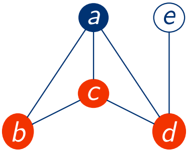
Busca em largura

- Cada vértice é colorido de branco, azul ou vermelho
- Todos os vértices são inicializados com branco
- Quando um vértice é descoberto pela primeira vez ele torna-se azul
- Vértices cujos adjacentes são todos descobertos tornam-se vermelhos
- Se $(u, v) \in A$ e o vértice u é vermelho, entao v tem quer ser azul ou vermelho
- Vértices azul podem ter adjacentes brancos.



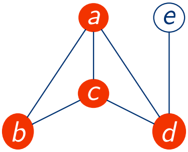
Busca em largura

- Cada vértice é colorido de branco, azul ou vermelho
- Todos os vértices são inicializados com branco
- Quando um vértice é descoberto pela primeira vez ele torna-se azul
- Vértices cujos adjacentes são todos descobertos tornam-se vermelhos
- Se $(u, v) \in A$ e o vértice u é vermelho, entao v tem quer ser azul ou vermelho
- Vértices azul podem ter adjacentes brancos.



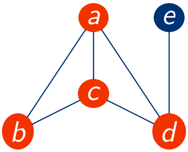
Busca em largura

- Cada vértice é colorido de branco, azul ou vermelho
- Todos os vértices são inicializados com branco
- Quando um vértice é descoberto pela primeira vez ele torna-se azul
- Vértices cujos adjacentes são todos descobertos tornam-se vermelhos
- Se $(u, v) \in A$ e o vértice u é vermelho, entao v tem quer ser azul ou vermelho
- Vértices azul podem ter adjacentes brancos.



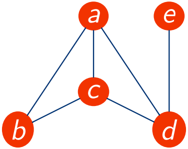
Busca em largura

- Cada vértice é colorido de branco, azul ou vermelho
- Todos os vértices são inicializados com branco
- Quando um vértice é descoberto pela primeira vez ele torna-se azul
- Vértices cujos adjacentes são todos descobertos tornam-se vermelhos
- Se $(u, v) \in A$ e o vértice u é vermelho, entao v tem quer ser azul ou vermelho
- Vértices azul podem ter adjacentes brancos.



Busca em largura

- Cada vértice é colorido de branco, azul ou vermelho
- Todos os vértices são inicializados com branco
- Quando um vértice é descoberto pela primeira vez ele torna-se azul
- Vértices cujos adjacentes são todos descobertos tornam-se vermelhos
- Se $(u, v) \in A$ e o vértice u é vermelho, entao v tem quer ser azul ou vermelho
- Vértices azul podem ter adjacentes brancos.



Alguns algoritmos

Caminho mais curto

- A busca em largura encontra o **caminho mais curto** entre dois vértice u e v.
- O caminho entre dois vertices quaisquer fica armazenado no vetor antecessor

Alguns algoritmos

Caminho mais curto

- A busca em largura encontra o **caminho mais curto** entre dois vértice u e v.
- O caminho entre dois vertices quaisquer fica armazenado no vetor antecessor

Ordenação Topológica

- Grafos direcionados **acíclicos** pode ser usados para indicar prescendência de eventos
- Uma aresta direcionada (u, v) indica que a atividade u tem que ocorrer antes da atividade v
- Os vértices ordenados topologicamente aparecem em **ordem inversa** aos seus tempos de término na busca em profundidade

Algoritmo menor caminho

Problema

- Dados: grafo $G = (V, A)$ orientado e distância c_{ij} associada à aresta $(i, j) \in A$.
- Problema: Obter o caminho **mais curto** entre dois vértices s e t.

Algoritmo menor caminho

Problema

- Dados: grafo $G = (V, A)$ orientado e distância c_{ij} associada à aresta $(i, j) \in A$.
- Problema: Obter o caminho **mais curto** entre dois vértices s e t.

Comprimento

O comprimento de um caminho é igual à **soma dos comprimentos** (distâncias) das arestas que formam o caminho. A distância ou comprimento de uma aresta pode ter diversas interpretações dependendo da aplicação: custos, distâncias, consumo de combustível, etc.

Algoritmo menor caminho

Problema

- Dados: grafo $G = (V, A)$ orientado e distância c_{ij} associada à aresta $(i, j) \in A$.
- Problema: Obter o caminho **mais curto** entre dois vértices s e t.

Comprimento

O comprimento de um caminho é igual à **soma dos comprimentos** (distâncias) das arestas que formam o caminho. A distância ou comprimento de uma aresta pode ter diversas interpretações dependendo da aplicação: custos, distâncias, consumo de combustível, etc.

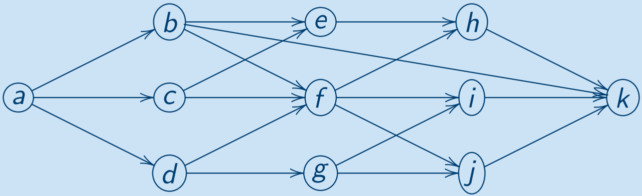
Exemplo

Dado um mapa rodoviário, determinar a **rota mais curta** de uma cidade a outra (rota mais rápida, rota com menor consumo de combustível, rota com menor valor de pedágio)

Algoritmo menor caminho

Encontre o menor caminho entre A e K

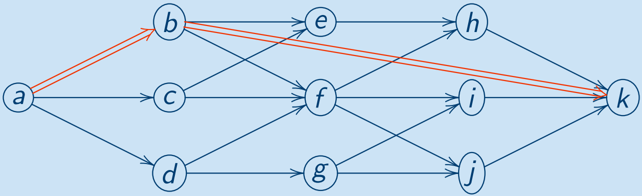
Construção de uma estrada entre duas cidades A e K. O grafo abaixo representa os diversos trechos possíveis. Determinar o trajeto ótimo (corresponde a achar o **caminho mais curto** de A a K em relação a estes custos).



Algoritmo menor caminho

Encontre o menor caminho entre A e K

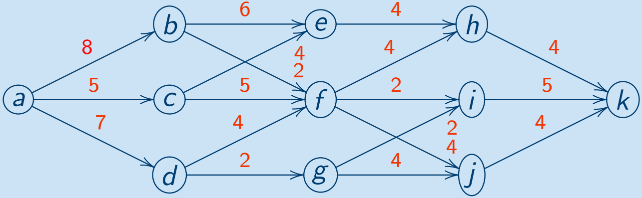
Construção de uma estrada entre duas cidades A e K. O grafo abaixo representa os diversos trechos possíveis. Determinar o trajeto ótimo (corresponde a achar o **caminho mais curto** de A a K em relação a estes custos).



Algoritmo menor caminho

Encontre o menor caminho entre A e K

Construção de uma estrada entre duas cidades A e K. O grafo abaixo representa os diversos trechos possíveis e o **custo de construção de cada um**. Determinar o trajeto ótimo cujo custo de construção seja mínimo (corresponde a achar o caminho mais curto de A a K em relação a estes custos).



Algoritmo menor caminho

Encontre o menor caminho entre A e K

Construção de uma estrada entre duas cidades A e K. O grafo abaixo representa os diversos trechos possíveis e o **custo de construção de cada um**. Determinar o trajeto ótimo cujo custo de construção seja mínimo (corresponde a achar o caminho mais curto de A a K em relação a estes custos).