

ESTRUTURA DE DADOS III

Grafos - Busca

Profa. Elaine Parros Machado de Sousa

alterações: Cristina Dutra de Aguiar

Material baseado em aulas dos professores:
Gustavo Batista, Robson Cordeiro, Moacir Ponti Jr.,
Maria Cristina Oliveira e Thiago A. S. Pardo

BUSCA EM GRAFOS: MOTIVAÇÃO

- Percorrer um grafo é um **problema fundamental**
 - deve-se ter uma forma **sistemática** de visitar as arestas e os vértices
 - o algoritmo deve ser suficientemente **flexível** para se adequar à diversidade de grafos
- Requisitos
 - não deve haver repetições (desnecessárias) de visitas a um vértice e/ou aresta
 - todos os vértices e/ou arestas devem ser visitados



BUSCA EM GRAFOS: TIPOS DE BUSCA

○ Exemplos:

- dado um grafo $\mathbf{G} = (\mathbf{V}, \mathbf{A})$ e um vértice $\mathbf{v} \in \mathbf{V} \Rightarrow$ encontrar todos os vértices em \mathbf{G} que estão conectados a \mathbf{v} .
- dado um grafo $\mathbf{G} = (\mathbf{V}, \mathbf{A}) \Rightarrow$ visitar todos os vértices de \mathbf{G} .

○ Duas maneiras principais de realizar essas tarefas:

- **Busca em profundidade**
- **Busca em largura**



Busca em Largura: Definição

○ *Breadth-First Search – BFS*

- expande a fronteira entre vértices descobertos e não descobertos uniformemente através da largura da fronteira.

○ Características

- o algoritmo descobre todos os vértices a uma distância k do vértice origem antes de se descobrir qualquer vértice a uma distância $k+1$
- a busca em largura permite descobrir todos os vértices alcançáveis a partir de um vértice de origem u , com o menor número de arestas entre u e todos os outros vértices



Busca em Largura: Estratégia

- Cada vértice é colorido de **branco**, **cinza** ou **preto**.
 - todos os vértices são inicialmente **brancos**.
 - quando um vértice v é “descoberto” pela primeira vez ele torna-se **cinza**.
 - quando todos os vértices adjacentes a v forem “descobertos”, v torna-se **preto**.



Busca em Largura: Estratégia

○ Observações

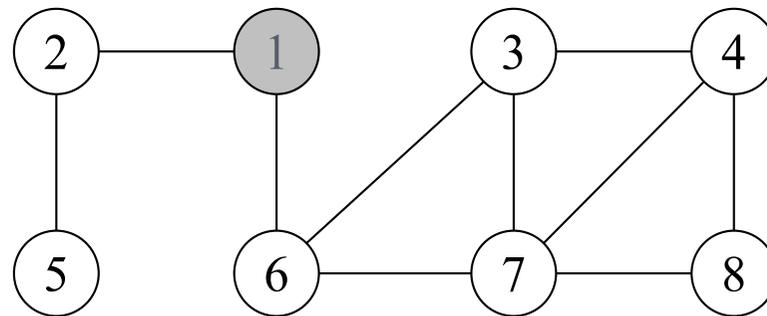
- vértices cinza e preto já foram “descobertos”, mas são diferenciados para assegurar que a busca ocorra em largura
- se $(u,v) \in A$ e o vértice u é preto, então o vértice v tem que ser cinza ou preto.
 - todos os vértices adjacentes a um vértice preto já foram “descobertos”
- vértices cinza podem ter alguns vértices adjacentes brancos, representando a fronteira entre vértices “descobertos” e não “descobertos”. 

Busca em Largura: Fila

- Uso de uma **fila** para organizar os vértices que devem ser descobertos
 - 1 a fila começa com o vértice origem
 - 2 o primeiro vértice da fila é recuperado e processado, sendo que seus vértices adjacentes são inseridos no final da fila
 - 3 se a fila está vazia, o processo termina. Caso contrário, volta-se ao passo 2



Busca em Largura: Exemplo



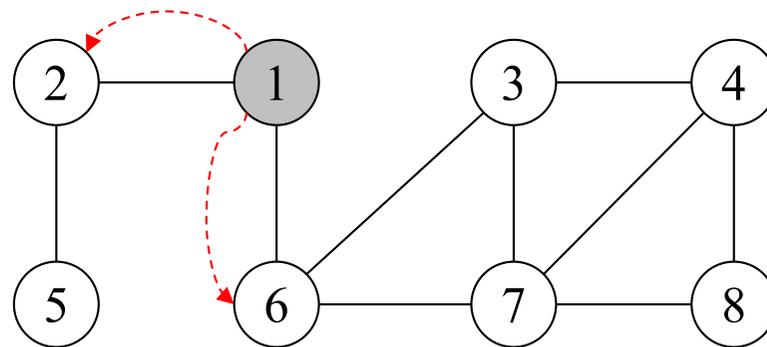
Vértice origem: 1

Distância k do vértice origem: 0

Ação: vértice 1 torna-se cinza



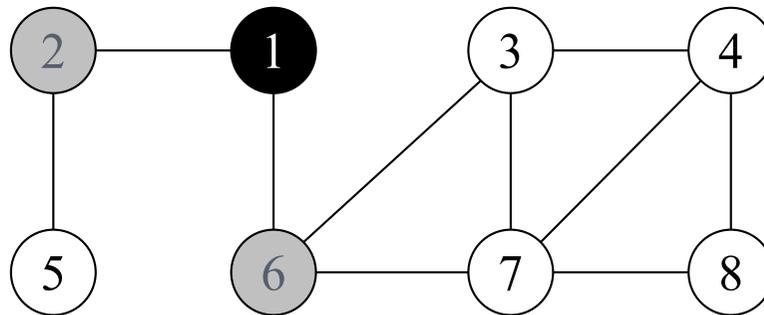
Busca em Largura: Exemplo



Vértices não descobertos adjacentes a 1: 2, 6
Distância k do vértice origem: 1



Busca em Largura: Exemplo



q	6	2			
k	1	1			

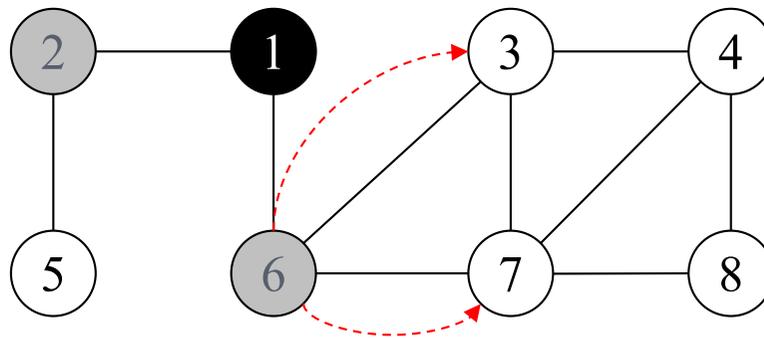
Vértices não descobertos adjacentes a 1: 2, 6

Distância k do vértice origem: 1

Ação: vértice 1 torna-se preto e vértices 2 e 6 tornam-se cinza



Busca em Largura: Exemplo

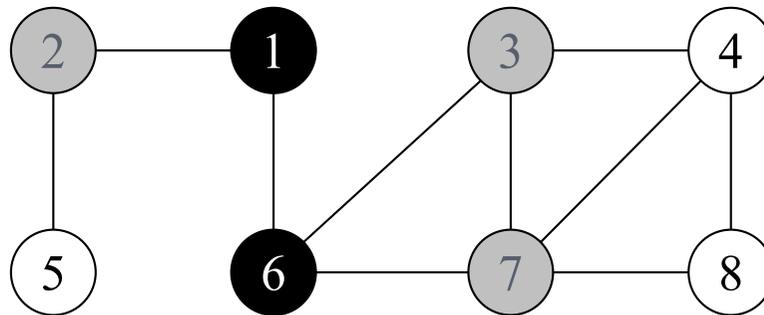


q	6	2			
k	1	1			

Vértices não descobertos adjacentes a 6: 3, 7
Distância k do vértice origem: 2



Busca em Largura: Exemplo



q	2	3	7		
k	1	2	2		

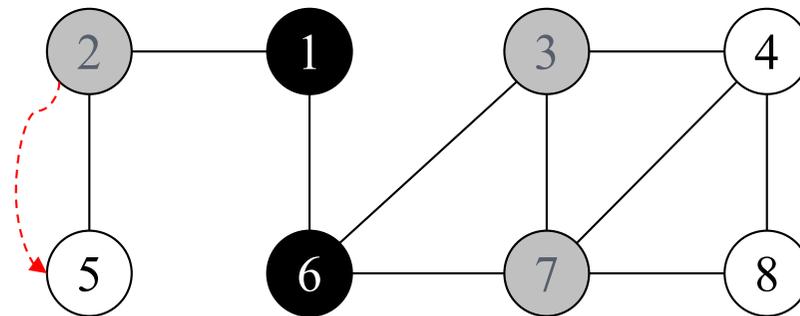
Vértices não descobertos adjacentes a 6: 3, 7

Distância k do vértice origem: 2

Ação: vértice 6 torna-se preto e vértices 3 e 7 tornam-se cinza



Busca em Largura: Exemplo

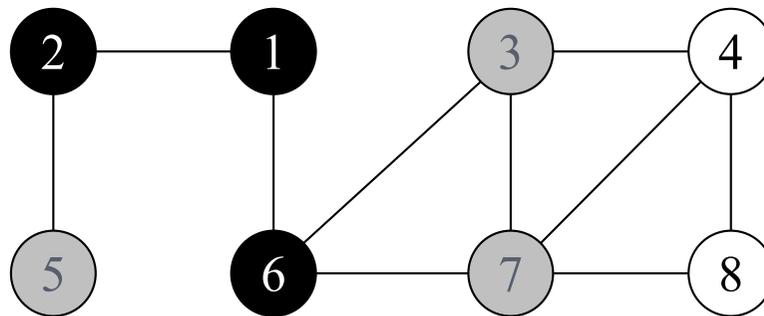


q	2	3	7		
k	1	2	2		

Vértices não descobertos adjacentes a 2: 5
Distância k do vértice origem: 2



Busca em Largura: Exemplo



q	3	7	5		
k	2	2	2		

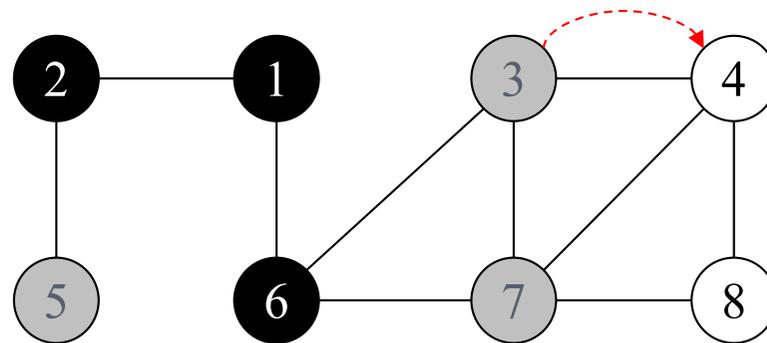
Vértices não descobertos adjacentes a 2: 5

Distância k do vértice origem: 2

Ação: vértice 2 torna-se preto e vértice 5 torna-se cinza



Busca em Largura: Exemplo

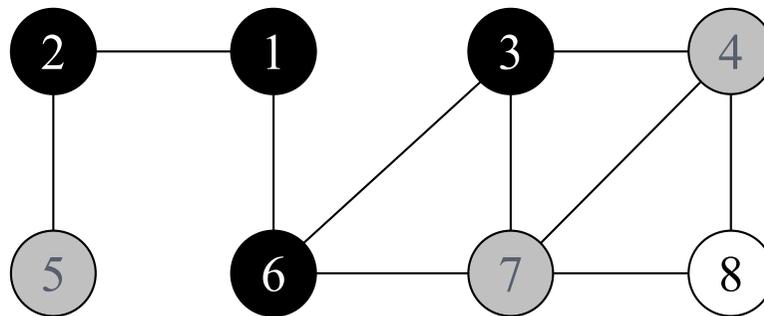


q	3	7	5		
k	2	2	2		

Vértices não descobertos adjacentes a 3: 4
Distância k do vértice origem: 3



Busca em Largura: Exemplo



q	7	5	4		
k	2	2	3		

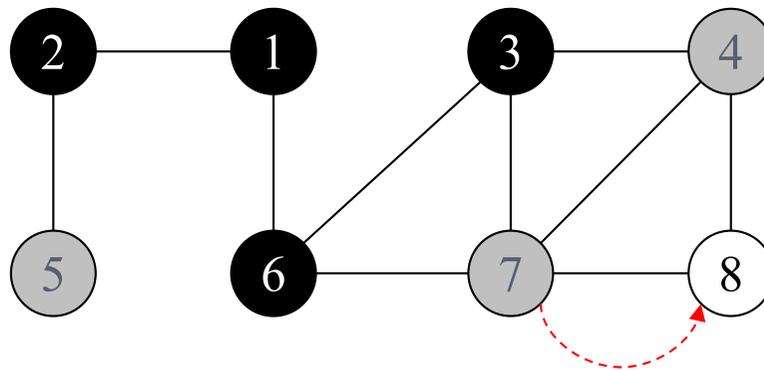
Vértices não descobertos adjacentes a 3: 4

Distância k do vértice origem: 3

Ação: vértice 3 torna-se preto e vértice 4 torna-se cinza



Busca em Largura: Exemplo

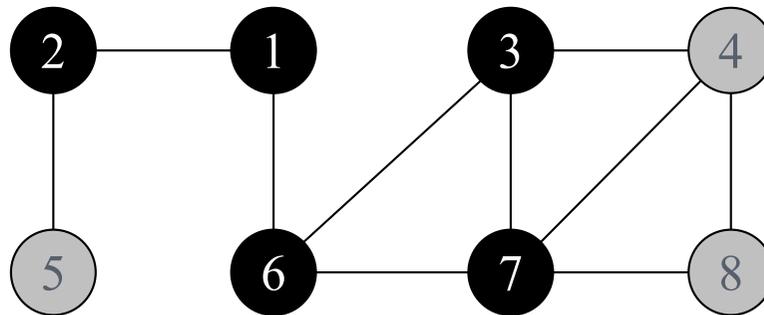


q	7	5	4		
k	2	2	3		

Vértices não descobertos adjacentes a 7: 8
Distância k do vértice origem: 3



Busca em Largura: Exemplo



q	5	4	8		
k	2	3	3		

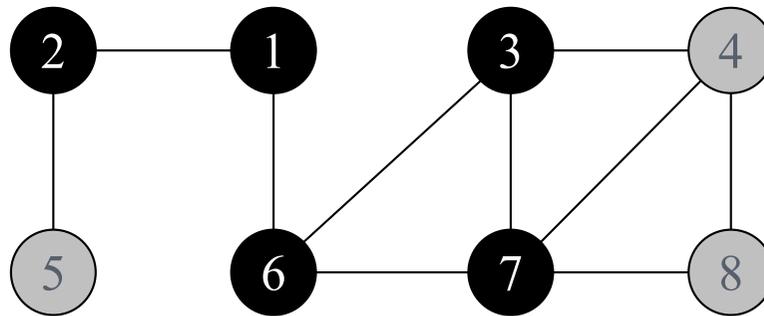
Vértices não descobertos adjacentes a 7: 8

Distância k do vértice origem: 3

Ação: vértice 7 torna-se preto e vértice 8 torna-se cinza



Busca em Largura: Exemplo

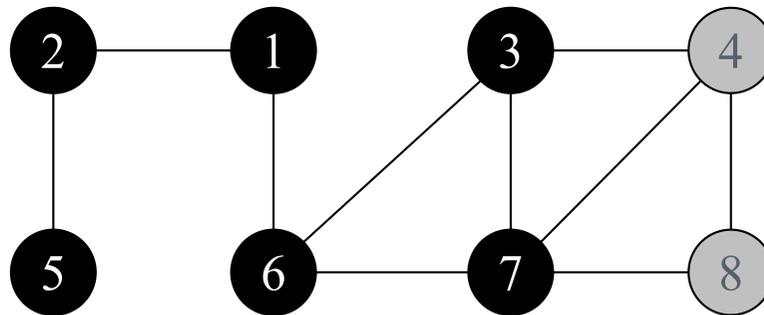


q	5	4	8		
k	2	3	3		

Vértices não descobertos adjacentes a 5: nenhum
Distância k do vértice origem: -



Busca em Largura: Exemplo

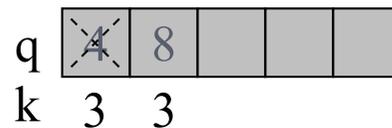
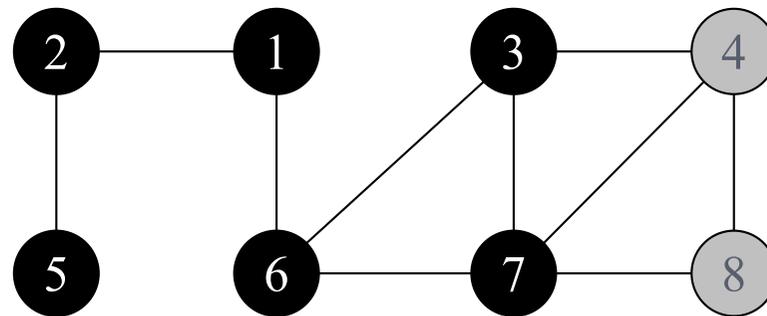


q	4	8			
k	3	3			

Vértices não descobertos adjacentes a 5: nenhum
Distância k do vértice origem: -
Ação: vértice 5 torna-se preto



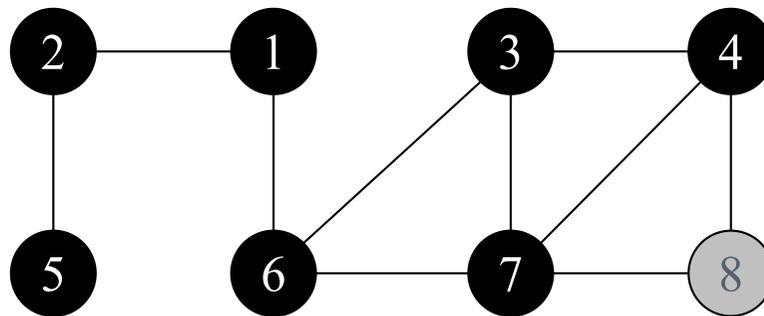
Busca em Largura: Exemplo



Vértices não descobertos adjacentes a 4: nenhum
Distância k do vértice origem: -



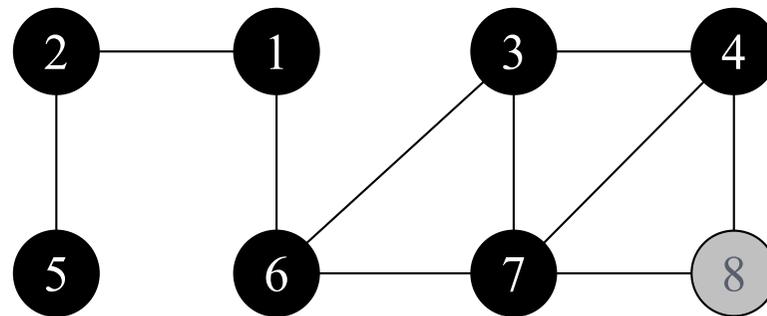
Busca em Largura: Exemplo



Vértices não descobertos adjacentes a 4: nenhum
Distância k do vértice origem: -
Ação: vértice 4 torna-se preto



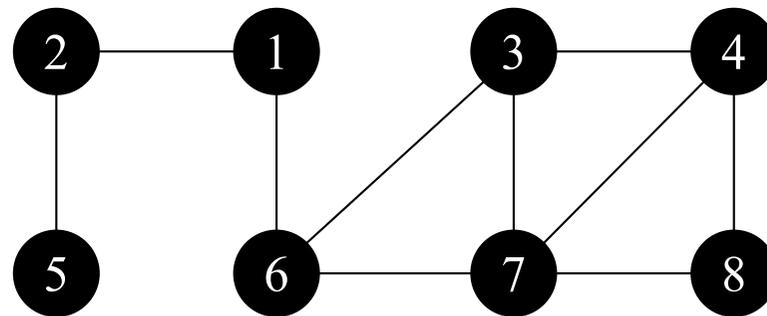
Busca em Largura: Exemplo



Vértices não descobertos adjacentes a 8: nenhum
Distância k do vértice origem: -



Busca em Largura: Exemplo



Vértices não descobertos adjacentes a 8: nenhum
Distância k do vértice origem: -
Ação: vértice 8 torna-se preto



Busca em Largura: Complexidade

$$O(|V| + |A|)$$

○ Característica

- linear em relação ao tamanho da representação do grafo usando listas de adjacência

○ $O(|V|)$

- todos os vértices são colocados na fila no máximo uma vez, ou seja, são executadas $|V|$ iterações com custo $O(1)$ cada uma delas

○ $O(|A|)$

- cada lista de adjacência é percorrida no máximo uma vez quando o vértice é retirado da fila



Busca em Largura: Uso

- O algoritmo é base para outros algoritmos importantes:
 - encontrar a árvore geradora mínima (**MST**) – **Algoritmo de Prim**
 - encontrar o caminho mais curto de um vértice v a todos os outros – **Algoritmo de Dijkstra**

a busca em largura resulta no **caminho mais curto** entre o vértice origem u e um vértice qualquer v .



BUSCA EM PROFUNDIDADE: DEFINIÇÃO

○ *Depth-First Search – DFS*

○ Características:

- o algoritmo busca o vértice “mais profundo” no grafo sempre que possível
- as arestas são exploradas a partir do vértice v mais recentemente descoberto e que ainda possui arestas não exploradas saindo dele
- quando todas as arestas adjacentes a v tiverem sido exploradas, a busca “anda para trás” (*backtracking*) para explorar vértices que saem do vértice a partir do qual v foi descoberto



BUSCA EM PROFUNDIDADE: ESTRATÉGIA

- Cada vértice é colorido de **branco**, **cinza** ou **preto**
 - todos os vértices são inicialmente **brancos**
 - quando um vértice v é “descoberto” pela primeira vez ele torna-se **cinza** e recebe um marcador de **tempo de descoberta**
 - quando todos os vértices adjacentes a v forem completamente “descobertos”, v torna-se **preto** e recebe um marcador de **tempo de término**



BUSCA EM PROFUNDIDADE: PILHA

- Uso de uma **pilha** para organizar os vértices que devem ser descobertos
 - a cada escolha de caminho a ser percorrido, empilha-se o vértice original e segue-se o caminho
 - cada vez que o caminho acaba, retorna-se ao vértice anterior empilhado

pilha pode ser implementada de forma implícita (via recursão) ou explícita



BUSCA EM PROFUNDIDADE: EXECUÇÃO

- Execução do algoritmo
 - gera uma **árvore de busca em profundidade**
- Classificação das arestas do grafo
 - **arestas de árvore**: arestas que ocorrem na árvore de busca em profundidade
 - **arestas de retorno**: arestas que ligam um nó a um antecessor na árvore
 - **arestas de avanço**: arestas que ligam um nó a um descendente na árvore
 - **arestas de cruzamento**: demais arestas

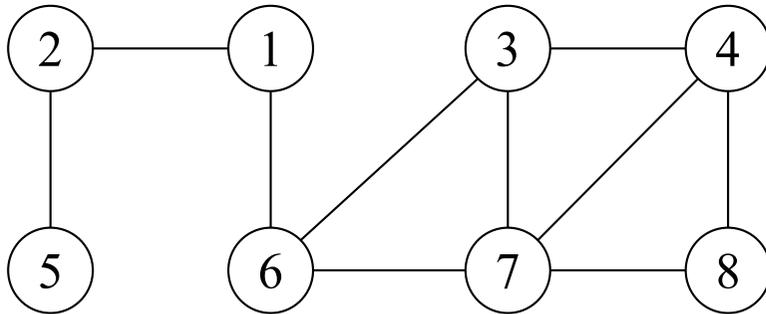


BUSCA EM PROFUNDIDADE: EXECUÇÃO

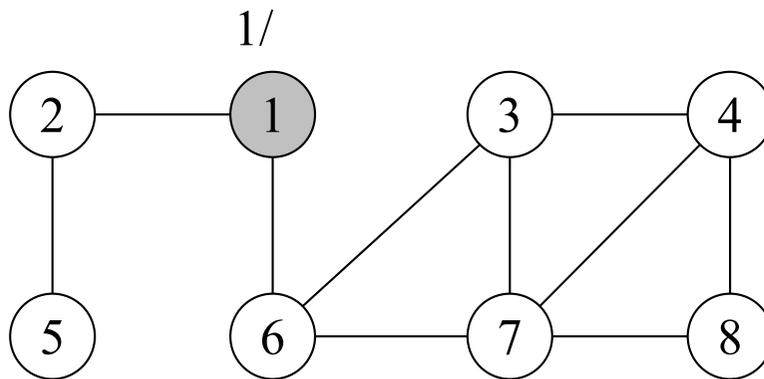
- Cada aresta (u,v)
 - classificada pela **cor do vértice v** alcançado quando a aresta é percorrida pela primeira vez
- Classificação das arestas do grafo
 - **arestas de árvore**: cor de v = branco
 - **arestas de retorno**: cor de v = cinza
 - **arestas de avanço**: cor de v = preto e $\text{tempoDescoberta}(u) < \text{tempoDescoberta}(v)$
 - **arestas de cruzamento**: cor de v = preto e $\text{tempoDescoberta}(u) > \text{tempoDescoberta}(v)$



BUSCA EM PROFUNDIDADE: EXEMPLO 1



BUSCA EM PROFUNDIDADE: EXEMPLO 1

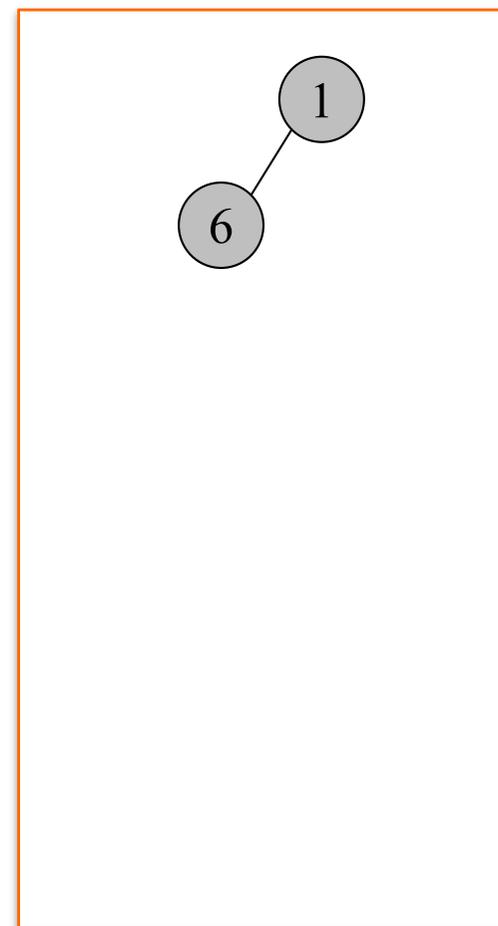
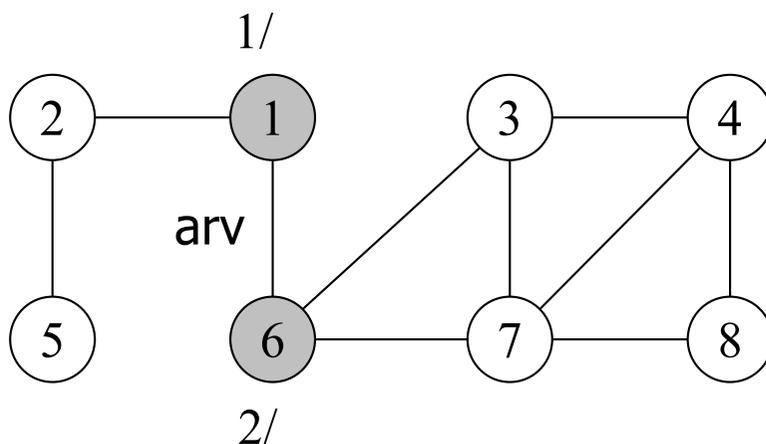


Vértice origem: 1
Tempo de descoberta: 1
Ação: vértice 1 torna-se cinza
Tempo de término: -

árvore de busca
em profundidade



BUSCA EM PROFUNDIDADE: EXEMPLO 1

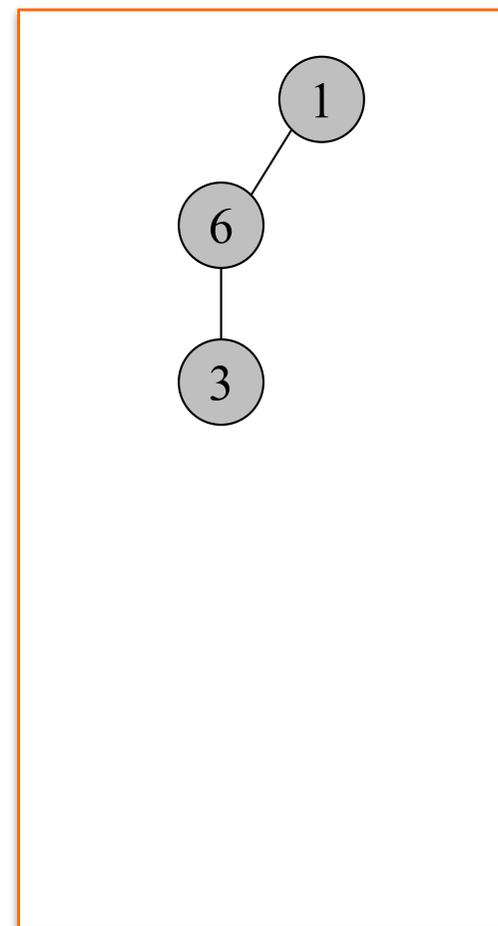
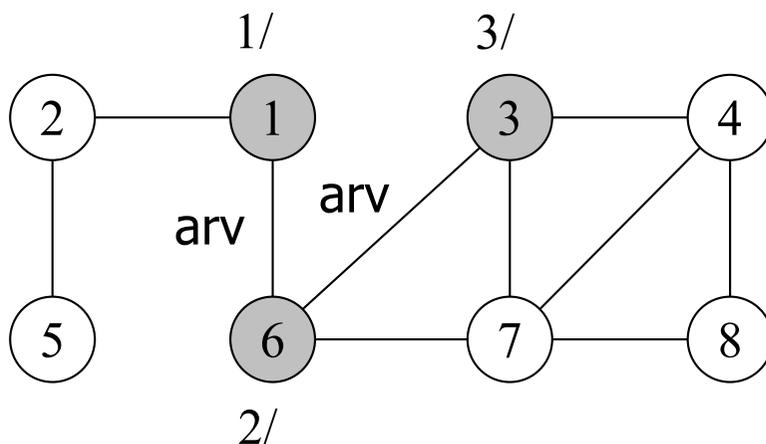


Primeiro vértice não descoberto adjacente a 1: 6
Tempo de descoberta: 2
Ação: vértice 6 torna-se cinza
Tempo de término: -

árvore de busca
em profundidade



BUSCA EM PROFUNDIDADE: EXEMPLO 1

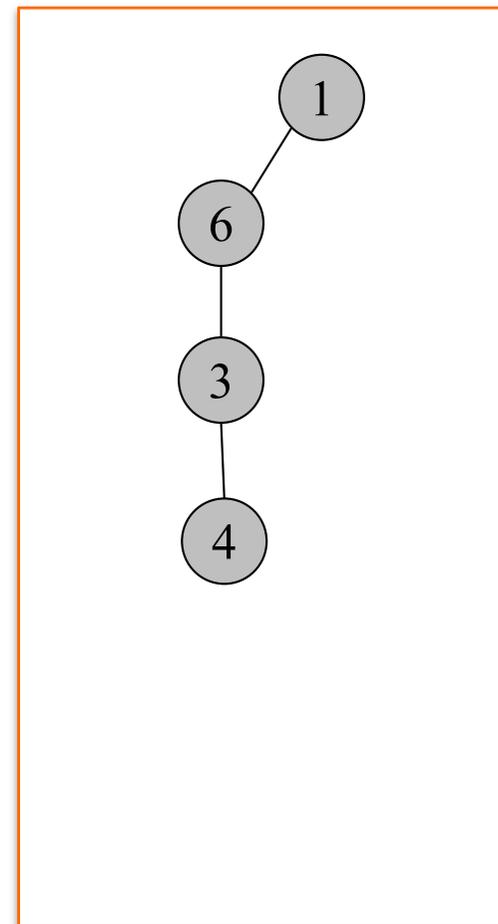
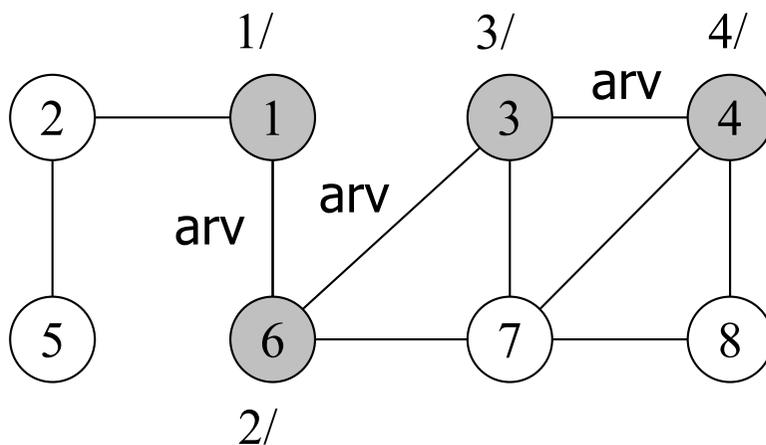


Primeiro vértice não descoberto adjacente a 6: 3
Tempo de descoberta: 3
Ação: vértice 3 torna-se cinza
Tempo de término: -

árvore de busca em profundidade



BUSCA EM PROFUNDIDADE: EXEMPLO 1

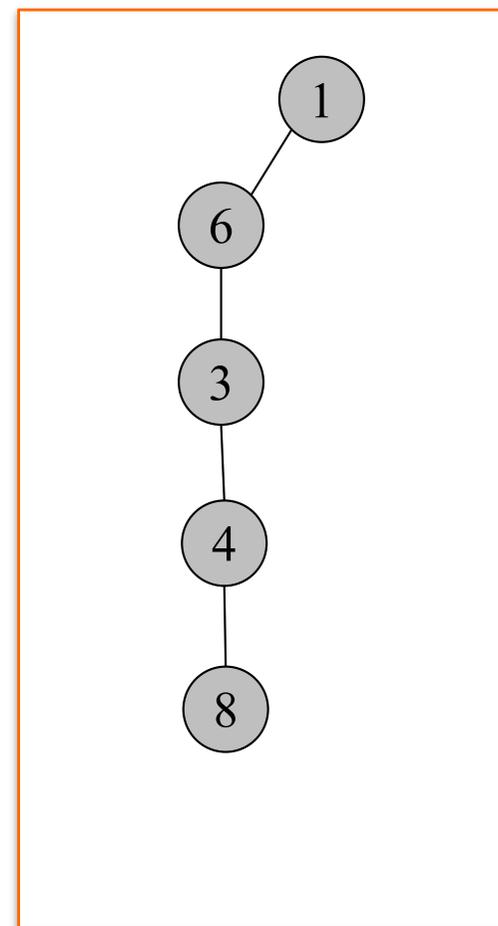
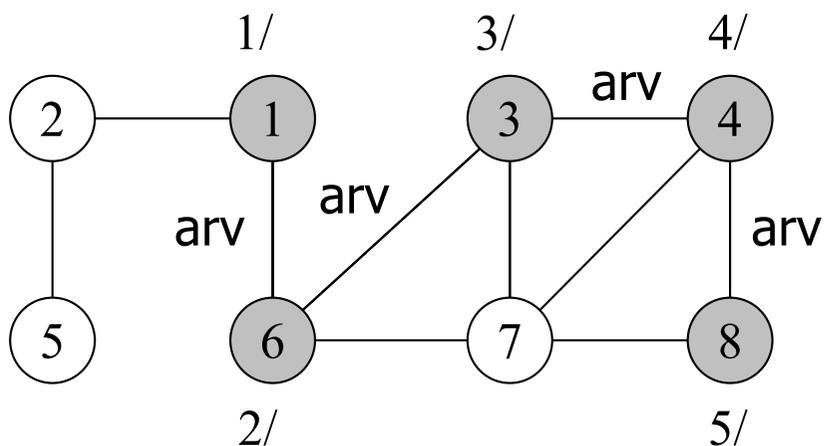


Primeiro vértice não descoberto adjacente a 3: 4
Tempo de descoberta: 4
Ação: vértice 4 torna-se cinza
Tempo de término: -

árvore de busca em profundidade



BUSCA EM PROFUNDIDADE: EXEMPLO 1

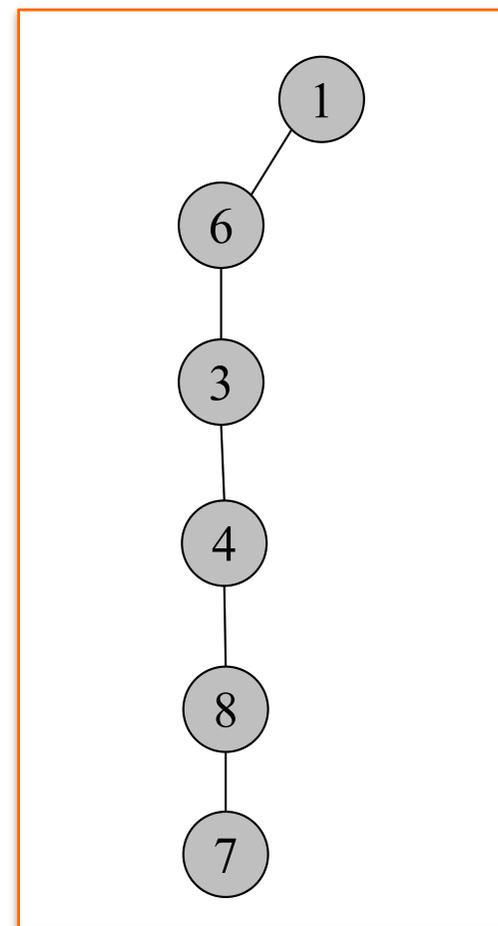
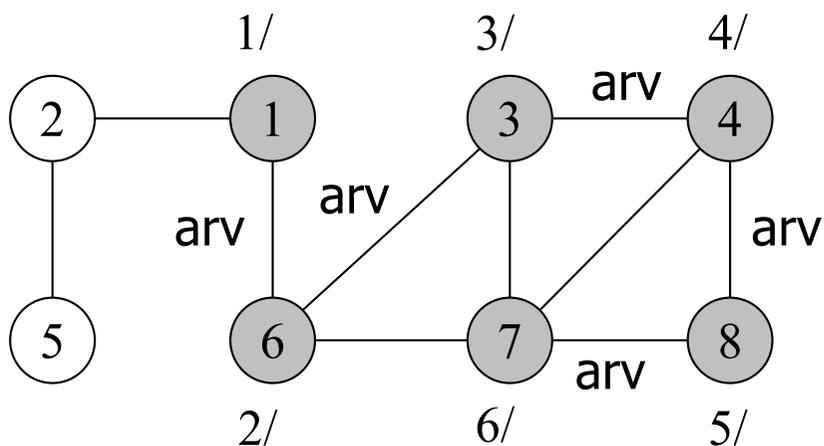


Primeiro vértice não descoberto adjacente a 4: 8
Tempo de descoberta: 5
Ação: vértice 8 torna-se cinza
Tempo de término: -

árvore de busca em profundidade



BUSCA EM PROFUNDIDADE: EXEMPLO 1

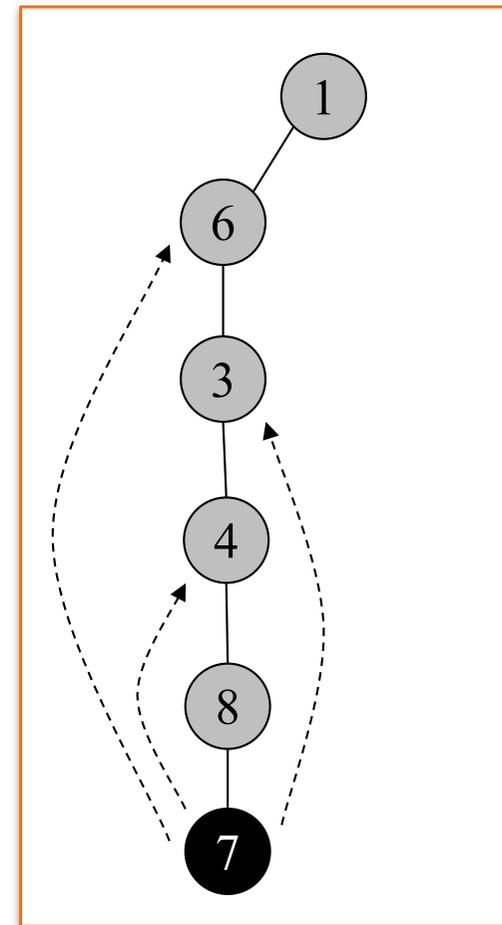
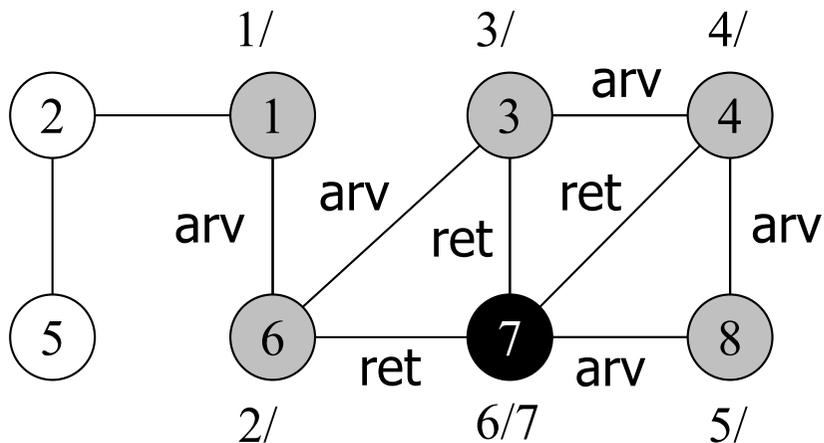


árvore de busca em profundidade

Primeiro vértice não descoberto adjacente a 8: 7
Tempo de descoberta: 6
Ação: vértice 7 torna-se cinza
Tempo de término: -



BUSCA EM PROFUNDIDADE: EXEMPLO 1

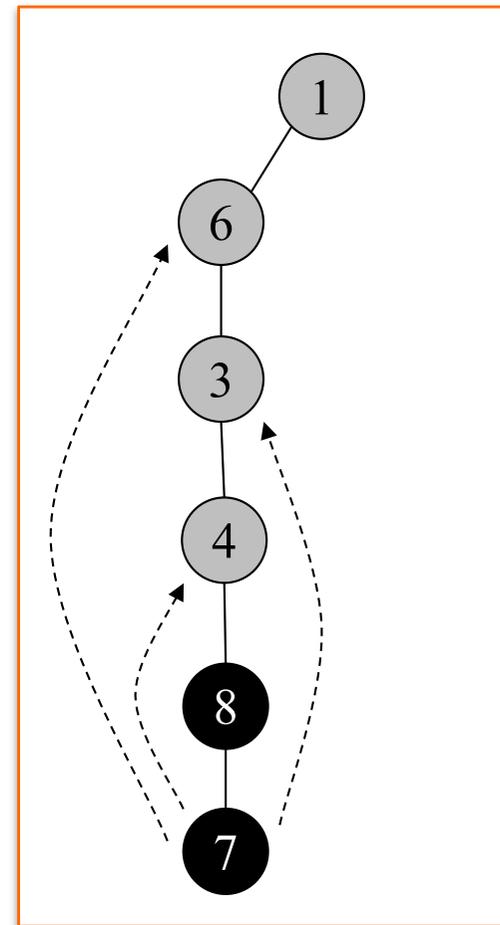
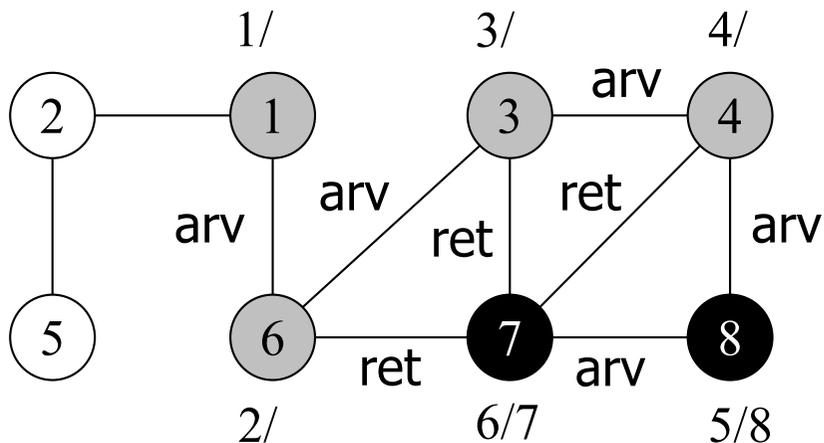


árvore de busca em profundidade

Primeiro vértice não descoberto adjacente a 7: nenhum
Tempo de descoberta: -
Ação: vértice 7 torna-se preto
Tempo de término: 7



BUSCA EM PROFUNDIDADE: EXEMPLO 1

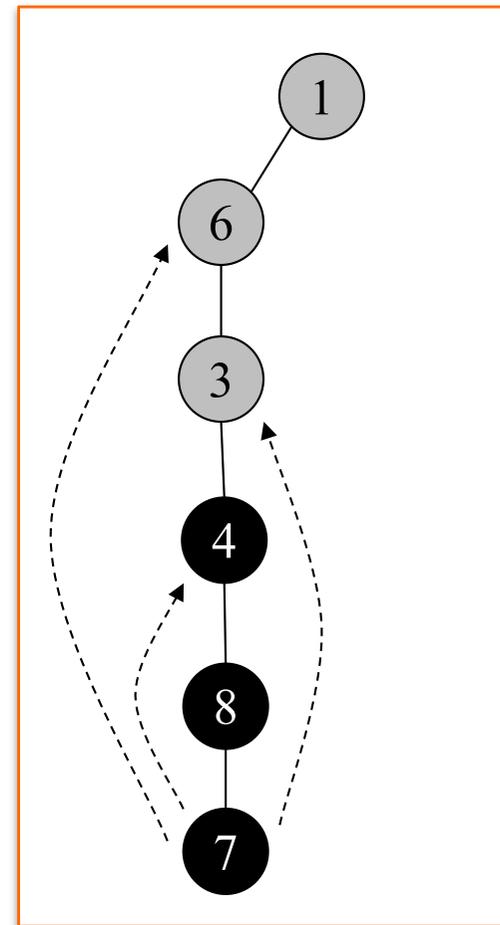
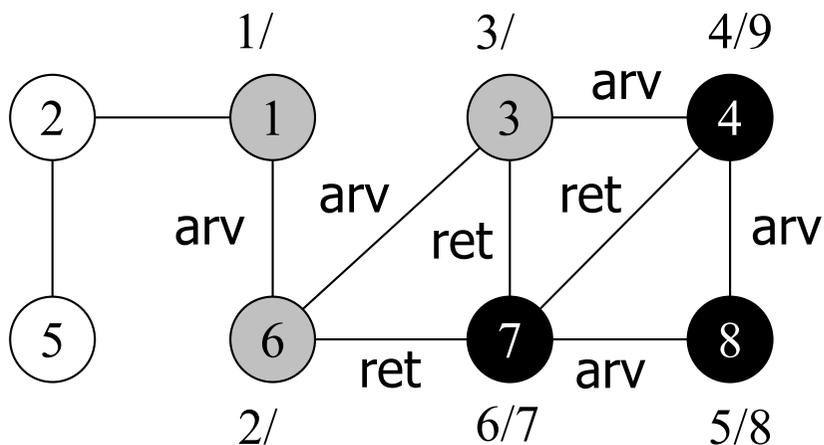


árvore de busca em profundidade

Primeiro vértice não descoberto adjacente a 8: nenhum
 Tempo de descoberta: -
 Ação: vértice 8 torna-se preto
 Tempo de término: 8



BUSCA EM PROFUNDIDADE: EXEMPLO 1

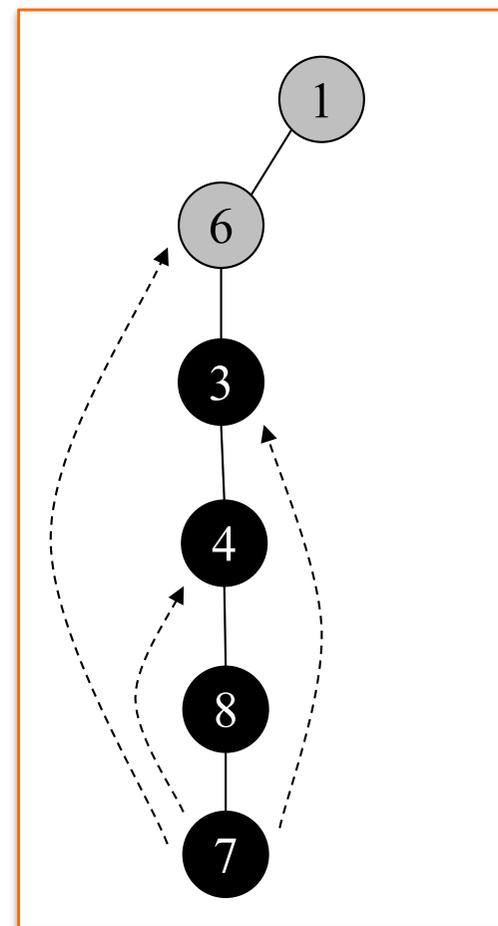
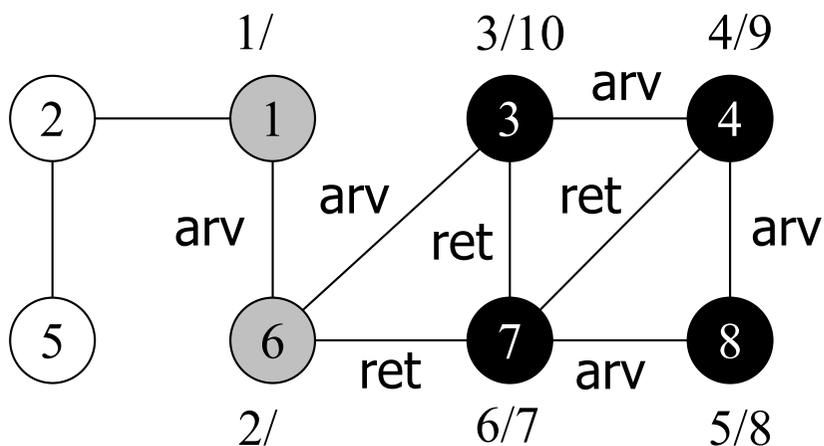


árvore de busca em profundidade

Primeiro vértice não descoberto adjacente a 4: nenhum
Tempo de descoberta: -
Ação: vértice 4 torna-se preto
Tempo de término: 9



BUSCA EM PROFUNDIDADE: EXEMPLO 1

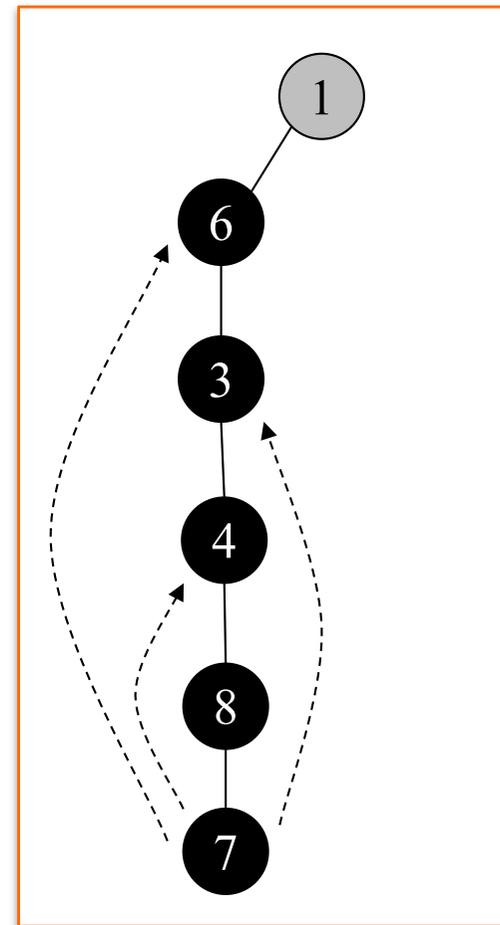
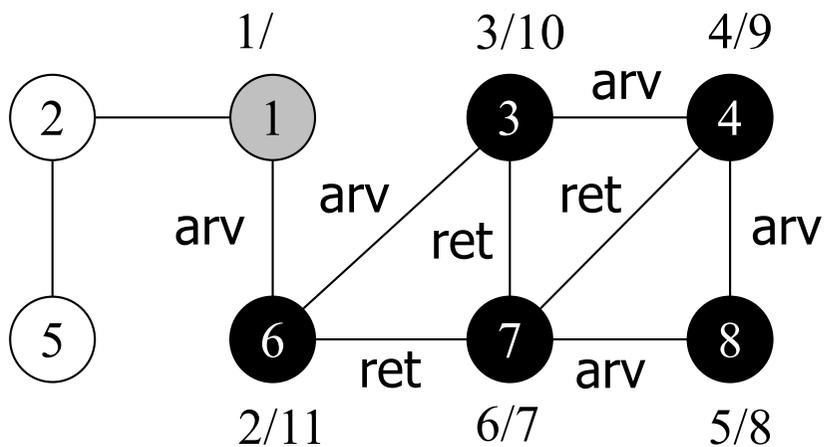


Primeiro vértice não descoberto adjacente a 3: nenhum
 Tempo de descoberta: -
 Ação: vértice 3 torna-se preto
 Tempo de término: 10

árvore de busca em profundidade



BUSCA EM PROFUNDIDADE: EXEMPLO 1

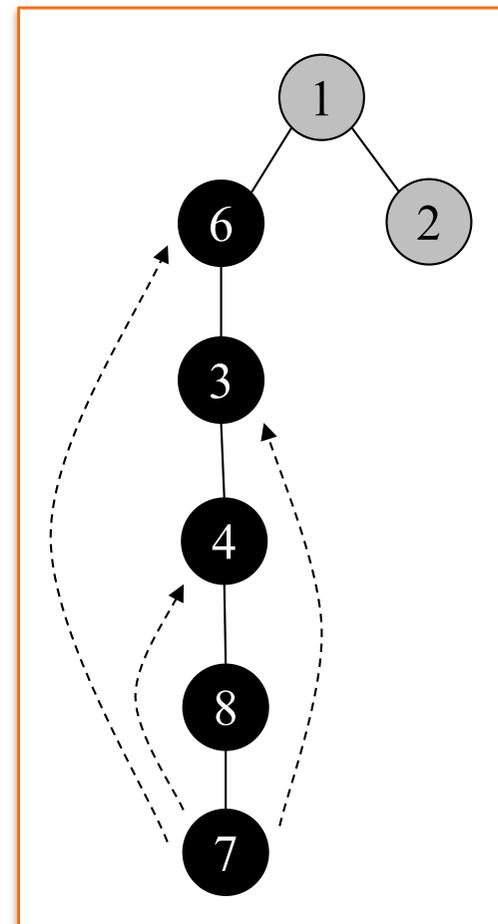
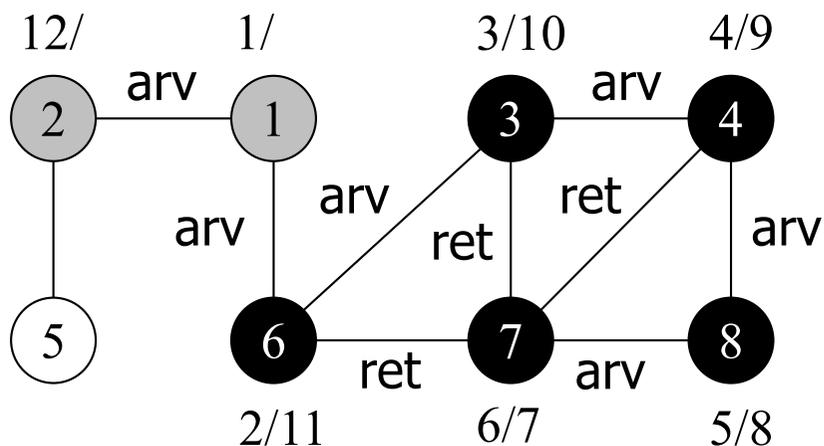


árvore de busca em profundidade

Primeiro vértice não descoberto adjacente a 6: nenhum
 Tempo de descoberta: -
 Ação: vértice 6 torna-se preto
 Tempo de término: 11



BUSCA EM PROFUNDIDADE: EXEMPLO 1

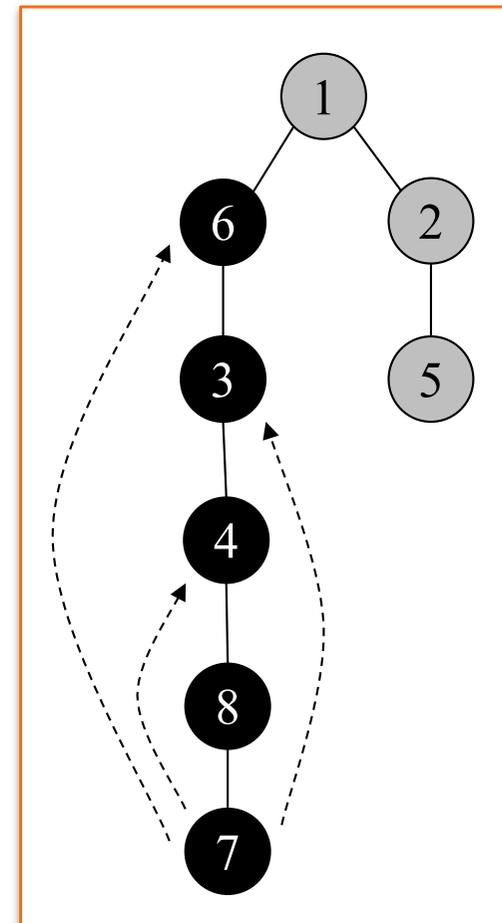
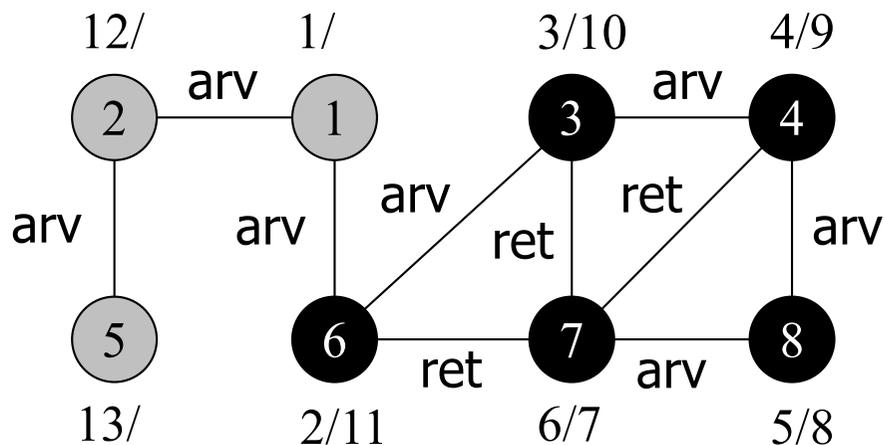


árvore de busca em profundidade

Primeiro vértice não descoberto adjacente a 1: 2
 Tempo de descoberta: 12
 Ação: vértice 2 torna-se cinza
 Tempo de término: -



BUSCA EM PROFUNDIDADE: EXEMPLO 1

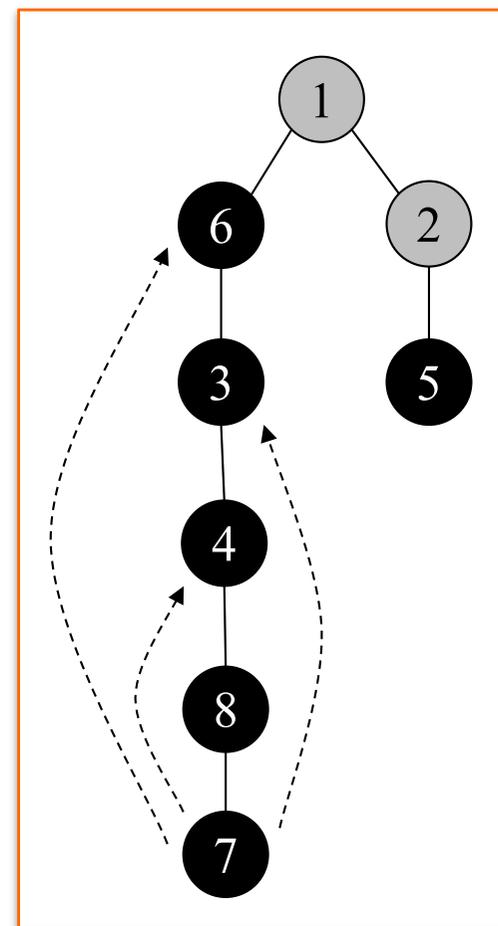
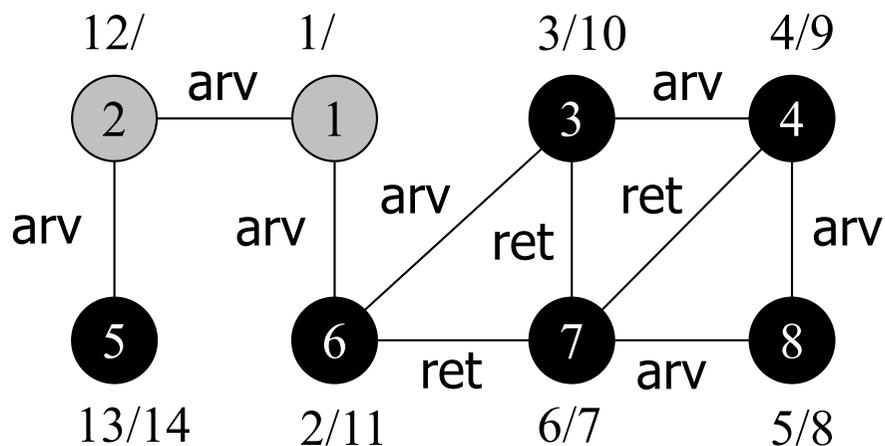


árvore de busca em profundidade

Primeiro vértice não descoberto adjacente a 2: 5
 Tempo de descoberta: 13
 Ação: vértice 5 torna-se cinza
 Tempo de término: -



BUSCA EM PROFUNDIDADE: EXEMPLO 1

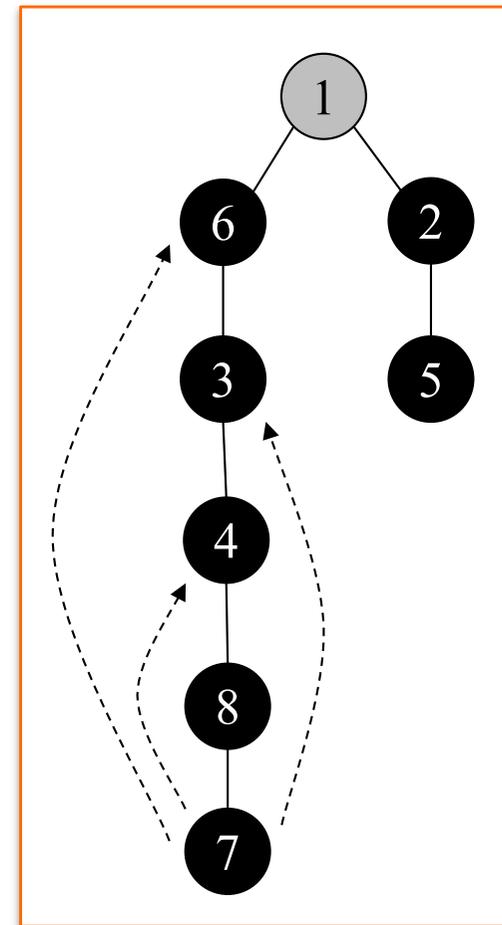
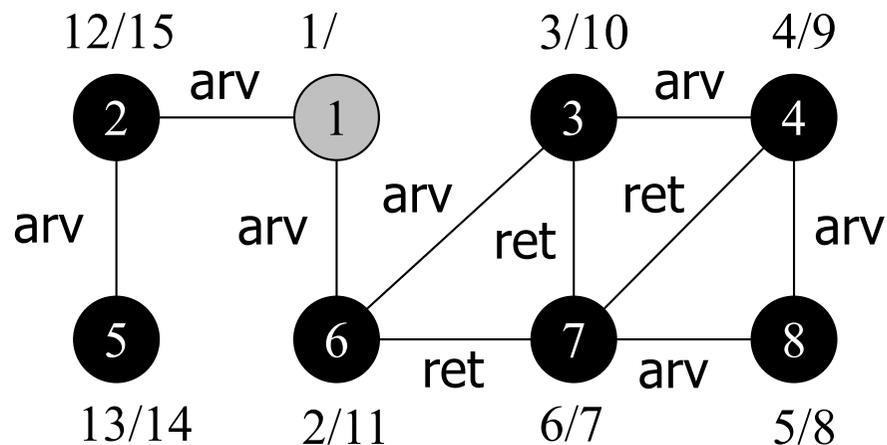


árvore de busca em profundidade

Primeiro vértice não descoberto adjacente a 5: nenhum
 Tempo de descoberta: -
 Ação: vértice 5 torna-se preto
 Tempo de término: 14



BUSCA EM PROFUNDIDADE: EXEMPLO 1

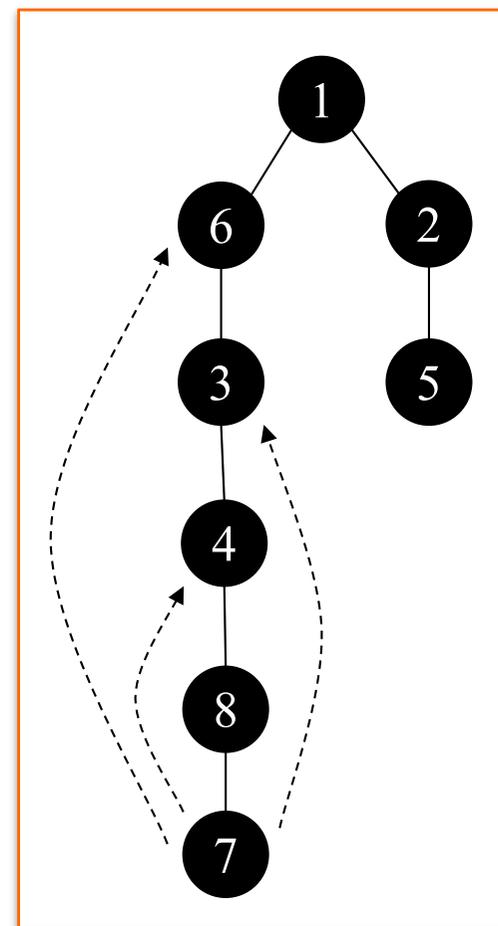
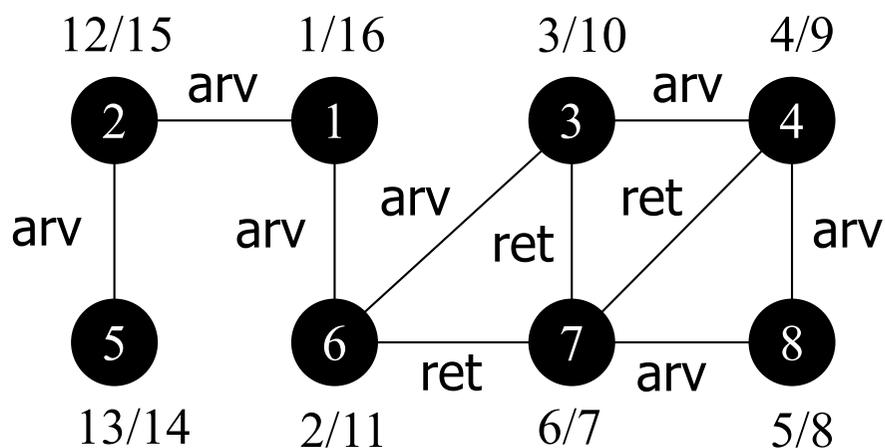


árvore de busca em profundidade

Primeiro vértice não descoberto adjacente a 2: nenhum
 Tempo de descoberta: -
 Ação: vértice 2 torna-se preto
 Tempo de término: 15



BUSCA EM PROFUNDIDADE: EXEMPLO 1

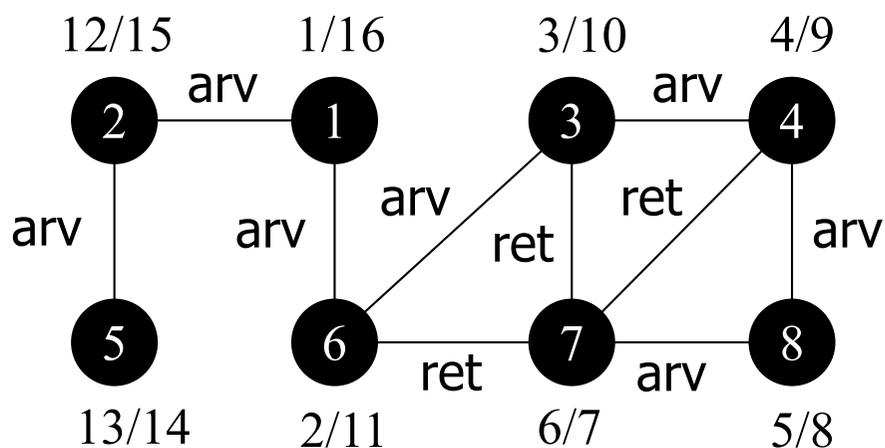


árvore de busca em profundidade

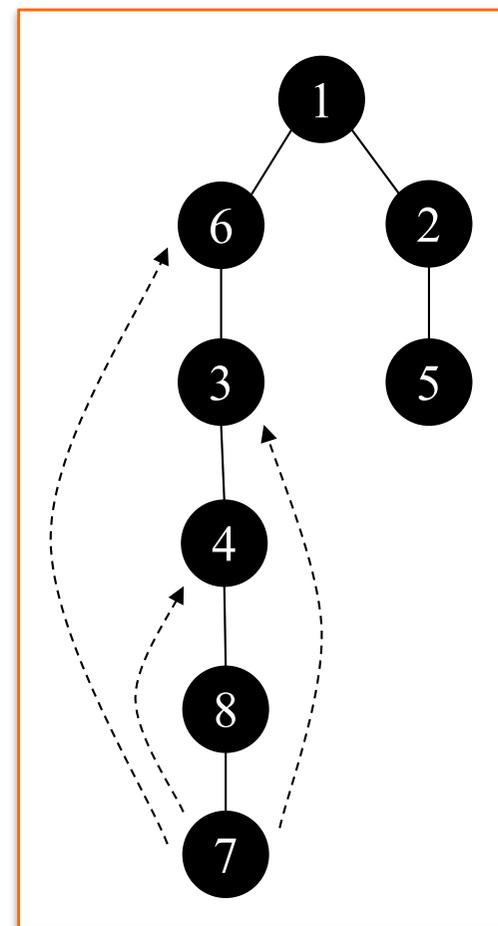
Primeiro vértice não descoberto adjacente a 1: nenhum
 Tempo de descoberta: -
 Ação: vértice 1 torna-se preto
 Tempo de término: 16



BUSCA EM PROFUNDIDADE: EXEMPLO 1



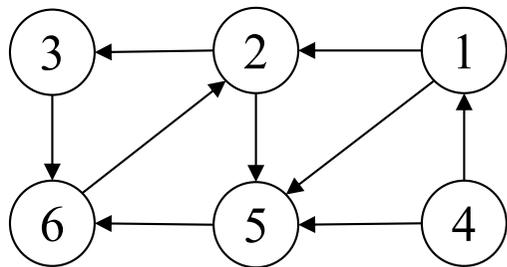
em um grafo não direcionado, todas as arestas são de **árvore** ou de **retorno**



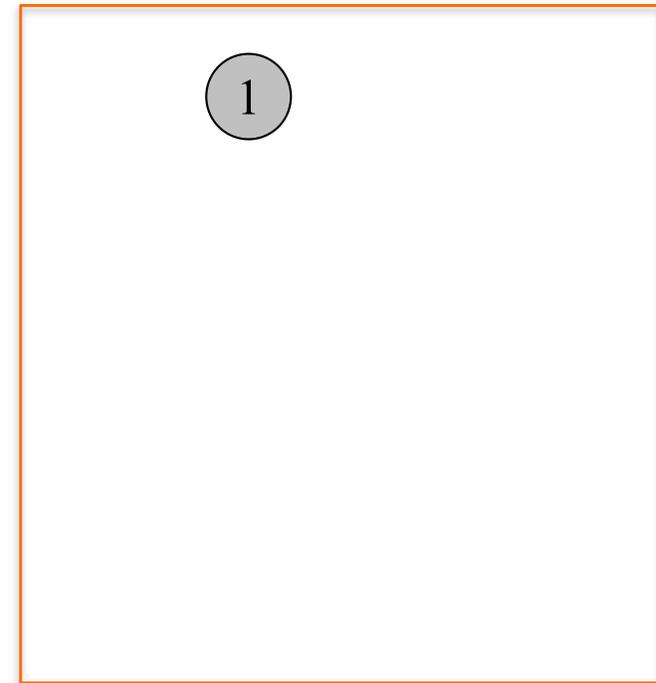
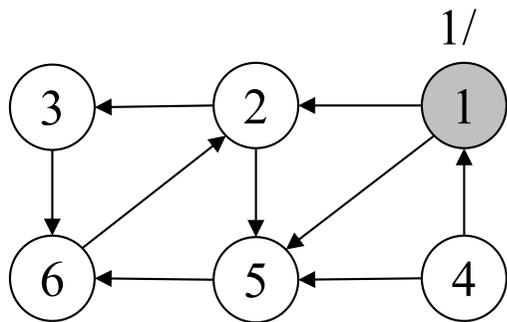
árvore de busca em profundidade



BUSCA EM PROFUNDIDADE: EXEMPLO 2



BUSCA EM PROFUNDIDADE: EXEMPLO 2



árvore de busca em profundidade

Vértice origem: 1

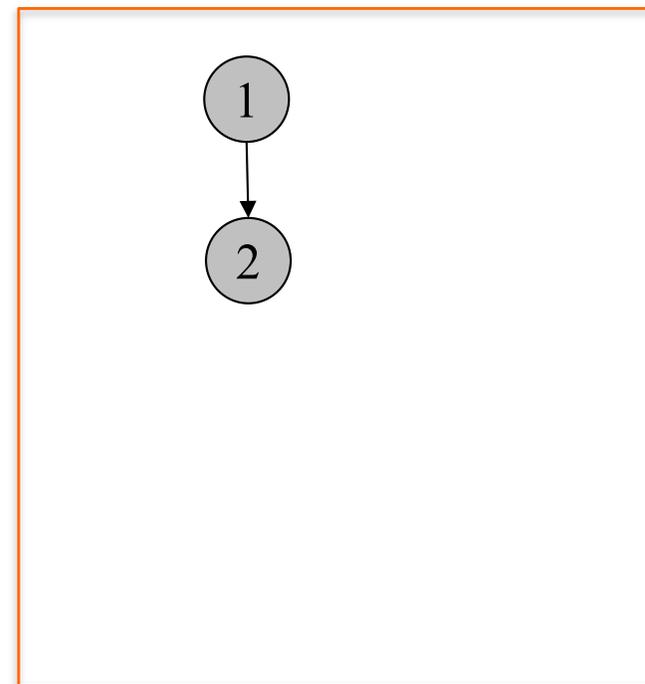
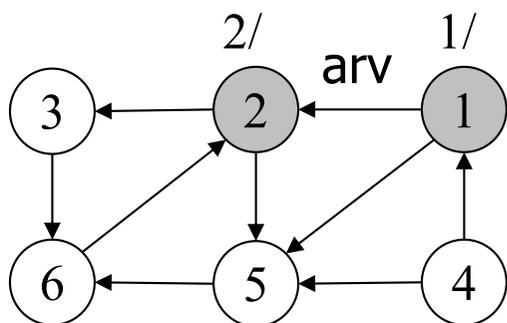
Tempo de descoberta: 1

Ação: vértice 1 torna-se cinza

Tempo de término: -



BUSCA EM PROFUNDIDADE: EXEMPLO 2

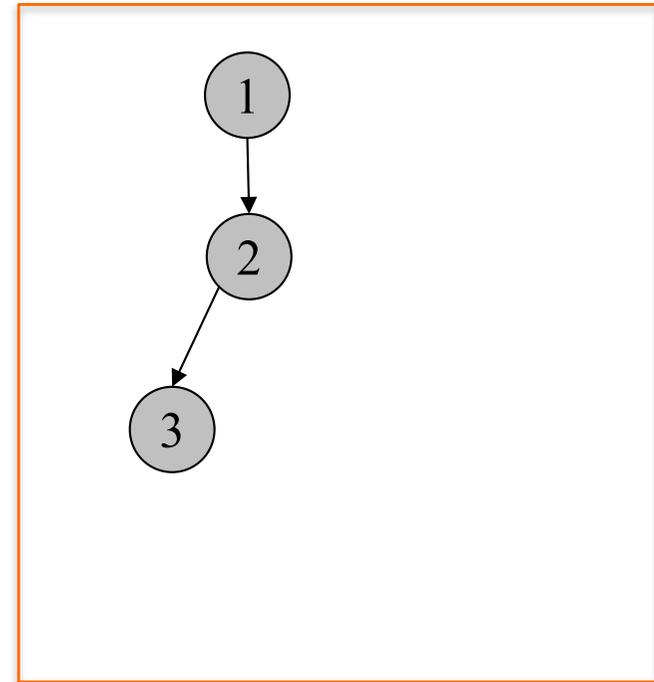
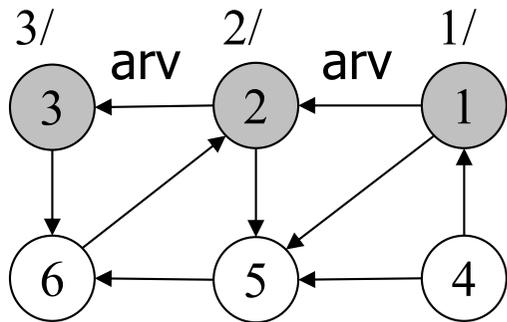


árvore de busca em profundidade

Primeiro vértice não descoberto adjacente a 1: 2
Tempo de descoberta: 2
Ação: vértice 2 torna-se cinza
Tempo de término:



BUSCA EM PROFUNDIDADE: EXEMPLO 2

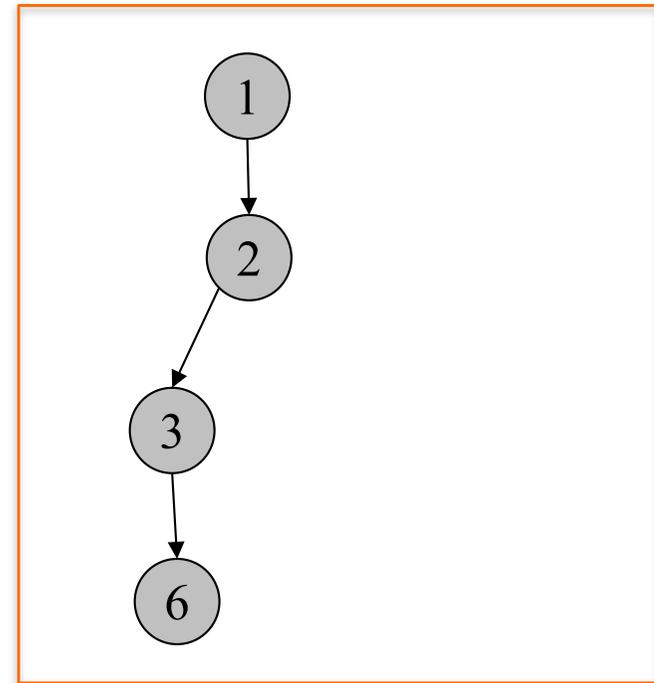
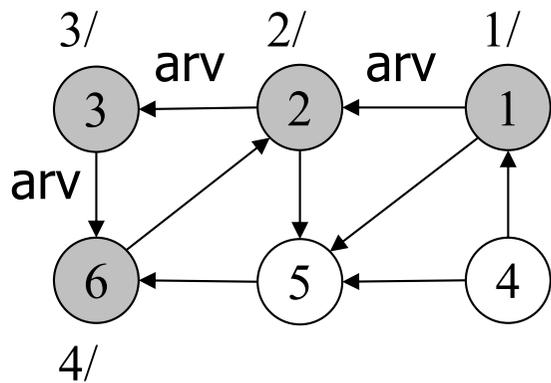


árvore de busca em profundidade

Primeiro vértice não descoberto adjacente a 2: 3
Tempo de descoberta: 3
Ação: vértice 3 torna-se cinza
Tempo de término:



BUSCA EM PROFUNDIDADE: EXEMPLO 2



árvore de busca em profundidade

Primeiro vértice não descoberto adjacente a 3: 6

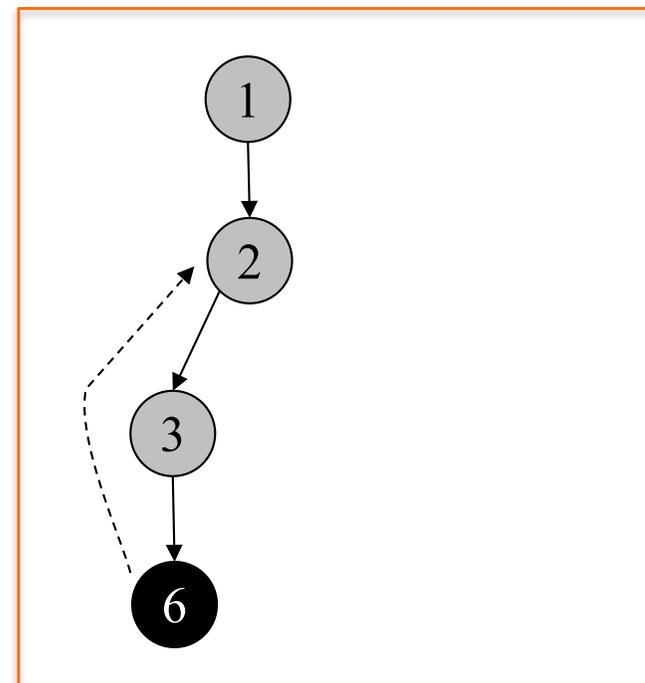
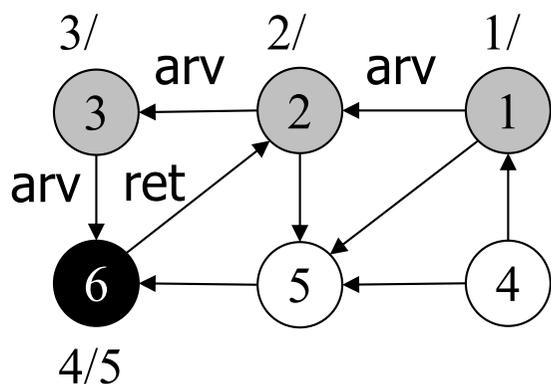
Tempo de descoberta: 4

Ação: vértice 6 torna-se cinza

Tempo de término:



BUSCA EM PROFUNDIDADE: EXEMPLO 2



árvore de busca em profundidade

Primeiro vértice não descoberto adjacente a 6: nenhum

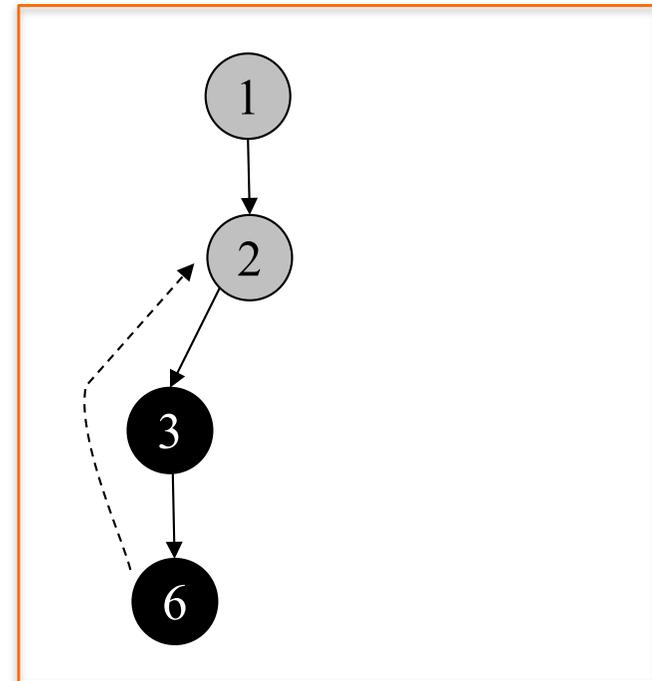
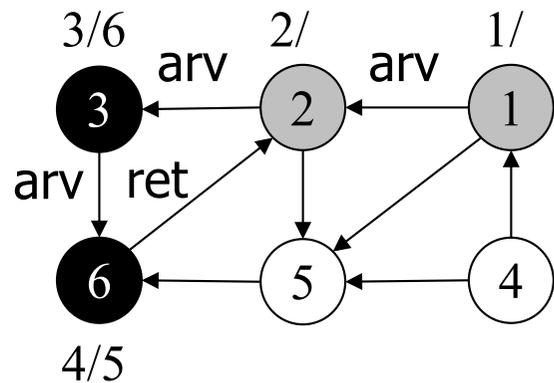
Tempo de descoberta: -

Ação: vértice 6 torna-se preto

Tempo de término: 5



BUSCA EM PROFUNDIDADE: EXEMPLO 2



árvore de busca em profundidade

Primeiro vértice não descoberto adjacente a 3: nenhum

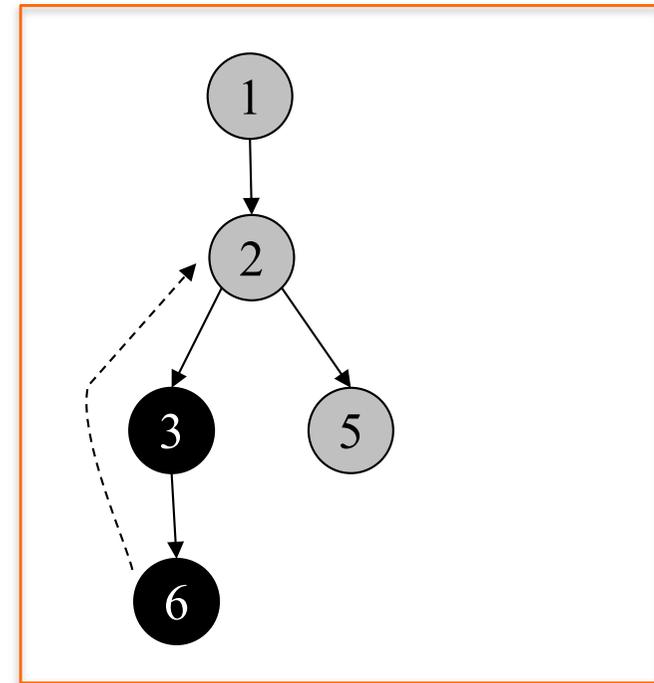
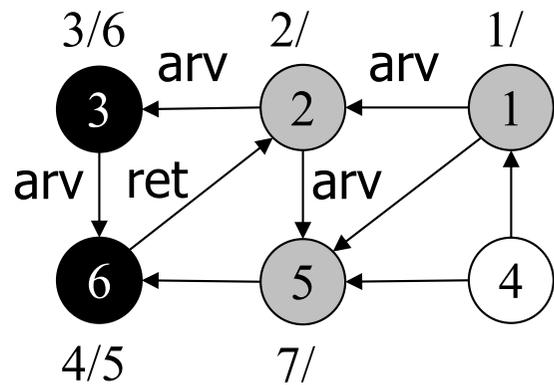
Tempo de descoberta: -

Ação: vértice 3 torna-se preto

Tempo de término: 6



BUSCA EM PROFUNDIDADE: EXEMPLO 2

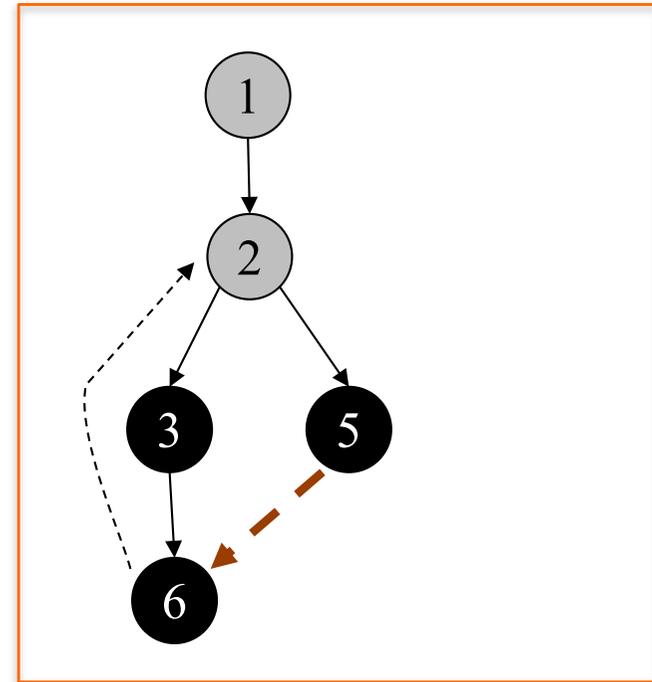
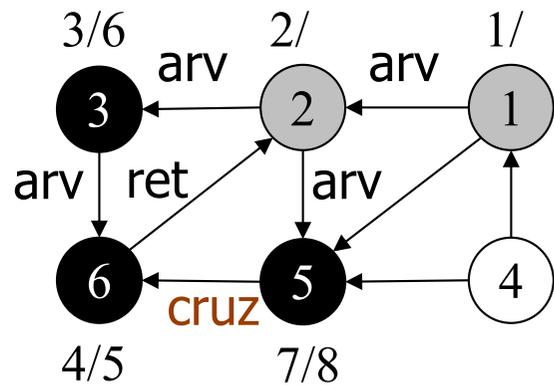


árvore de busca em profundidade

Primeiro vértice não descoberto adjacente a 2: 5
Tempo de descoberta: 7
Ação: vértice 5 torna-se preto
Tempo de término: -



BUSCA EM PROFUNDIDADE: EXEMPLO 2



árvore de busca em profundidade

Primeiro vértice não descoberto adjacente a 5: nenhum

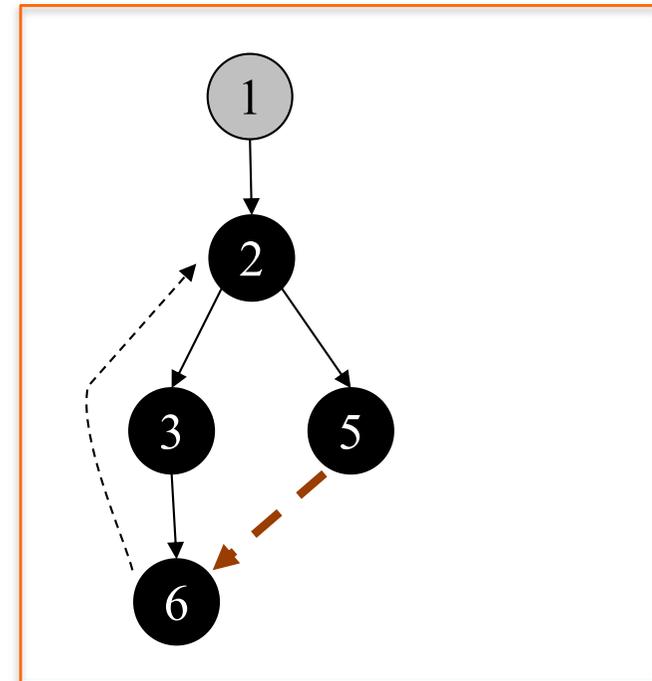
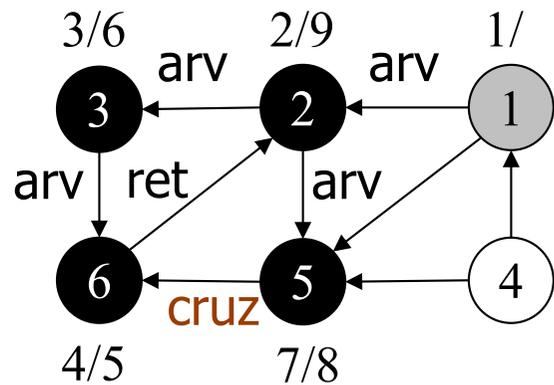
Tempo de descoberta: -

Ação: vértice 5 torna-se preto

Tempo de término: 8



BUSCA EM PROFUNDIDADE: EXEMPLO 2



árvore de busca em profundidade

Primeiro vértice não descoberto adjacente a 2: nenhum

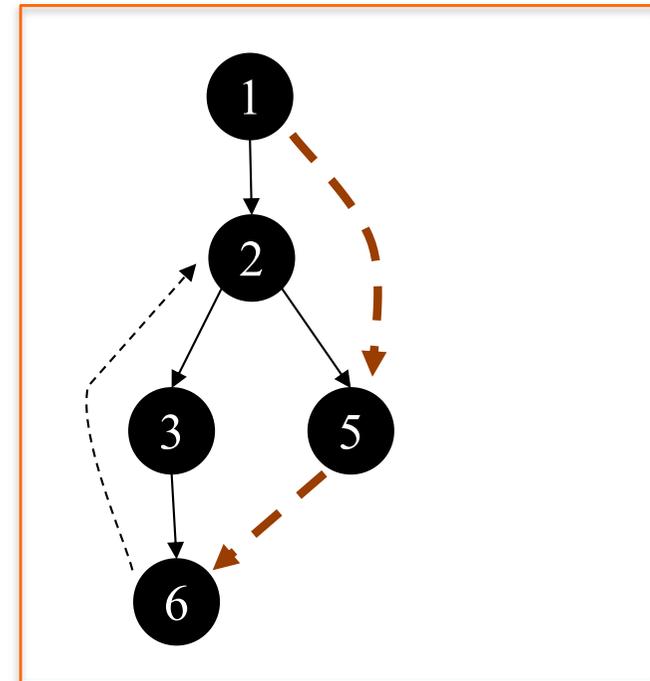
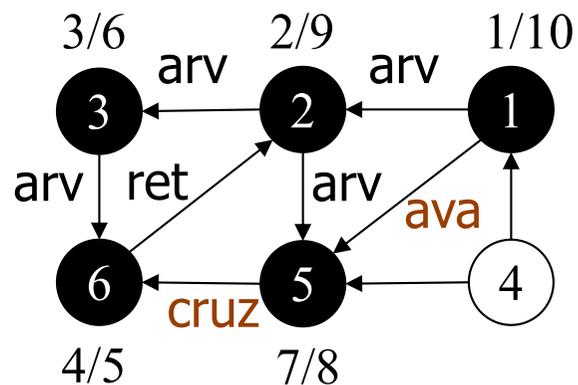
Tempo de descoberta: -

Ação: vértice 2 torna-se preto

Tempo de término: 9



BUSCA EM PROFUNDIDADE: EXEMPLO 2



árvore de busca em profundidade

Primeiro vértice não descoberto adjacente a 1: nenhum

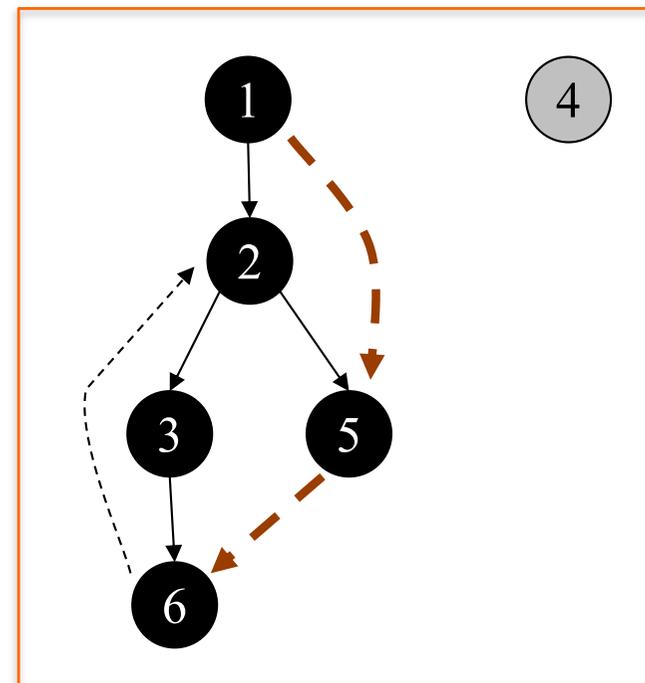
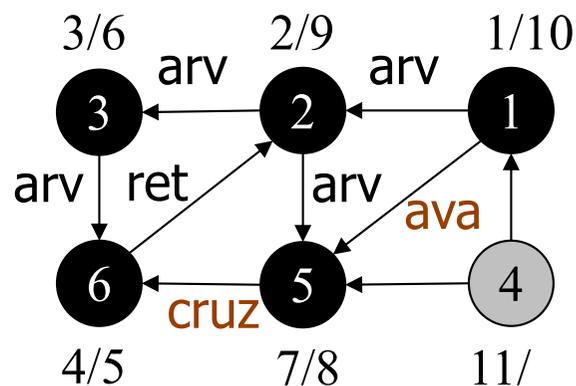
Tempo de descoberta: -

Ação: vértice 1 torna-se preto

Tempo de término: 10



BUSCA EM PROFUNDIDADE: EXEMPLO 2



árvore de busca em profundidade

Vértice origem: 4

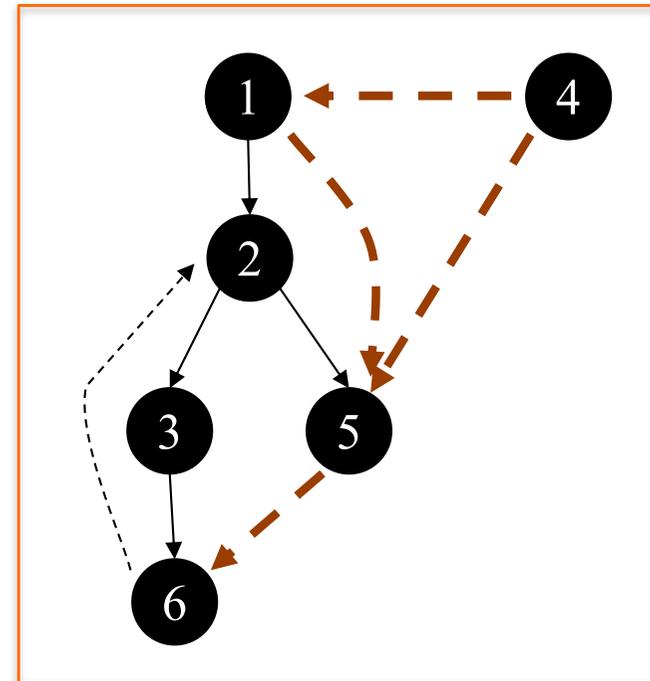
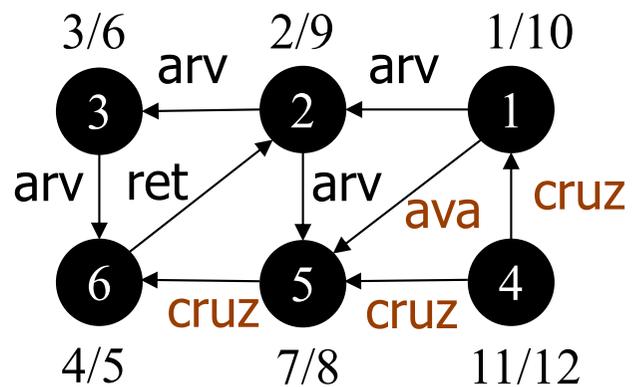
Tempo de descoberta: 11

Ação: vértice 4 torna-se cinza

Tempo de término: -



BUSCA EM PROFUNDIDADE: EXEMPLO 2



árvore de busca em profundidade

Primeiro vértice não descoberto adjacente a 4: nenhum

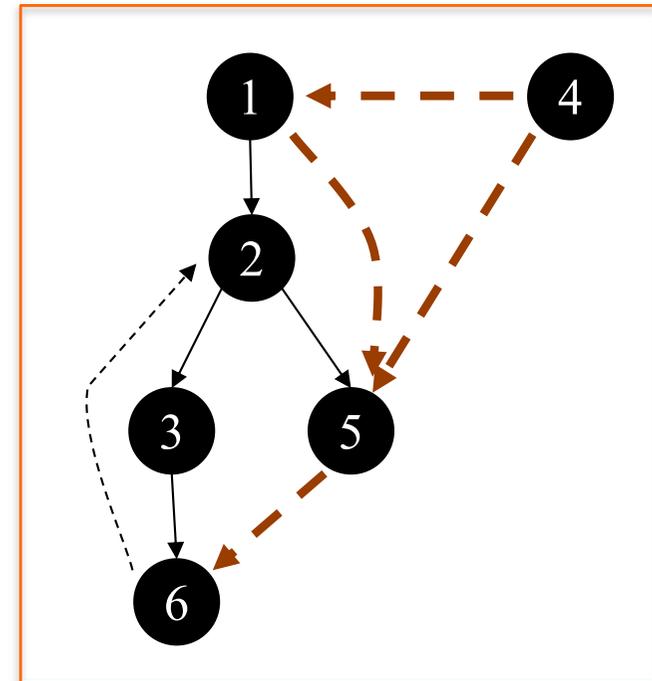
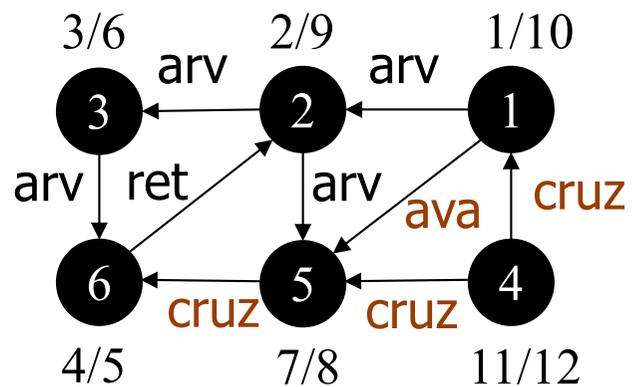
Tempo de descoberta: -

Ação: vértice 4 torna-se preto

Tempo de término: 12



BUSCA EM PROFUNDIDADE: EXEMPLO 2



em um grafo direcionado,
podem ocorrer ainda
arestas de **avanço** e de
cruzamento

árvore de busca
em profundidade



BUSCA EM PROFUNDIDADE: USO

- O algoritmo é base para outros algoritmos importantes
 - verificação de grafos acíclicos
 - descoberta de caminhos
 - ordenação topológica
 - descoberta de componentes fortemente conectados



Busca em Profundidade: Complexidade

$$O(|V| + |A|)$$

○ Característica

- linear em relação ao tamanho da representação do grafo usando listas de adjacência

○ $O(|V|)$

- cada vértice u torna-se a raiz de uma nova árvore de busca em profundidade apenas uma única vez (visitaDFS)

○ $O(|A|)$

- no visitaDFS, o laço é executado $|adj[u]|$ vezes, ou seja, $O(|A|)$ no total



BIBLIOGRAFIA

- N. Ziviani. Projeto de Algoritmos, Thomson, 2a. Edição, 2004.
- T. H. Cormen, C. E. Leiserson and R. L. Rivest. Introduction to Algorithms, MIT Press, 2nd Edition, 2001.

