

Busca em Profundidade

Profa. Sheila Moraes de Almeida

DAINF-UTFPR-PG

junho - 2018

- Iniciando em um vértice s , escolhe-se um vértice adjacente t não visitado.
- Visita-se t e escolhe-se um vértice adjacente u não visitado, e assim por diante.
- Quando todos os vértices adjacentes a t tiverem sido visitados, toma-se um próximo adjacente a s não visitado e prossegue-se o percurso.
- Se restarem quaisquer vértices não visitados, então um deles será selecionado como vértice de início de uma novo percurso.

- Visita-se o mais recente primeiro.
- Utiliza-se o conceito de pilha.
- Pode ser implementada facilmente com recursão.

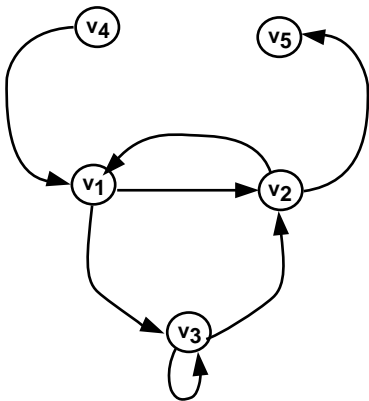
Inicialização:

- Usa o mesmo esquema de cores da BFS.

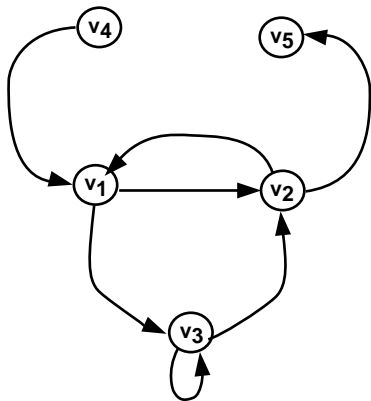
Saída:

- Floresta com árvores de precedência.

Execução do Algoritmo DFS

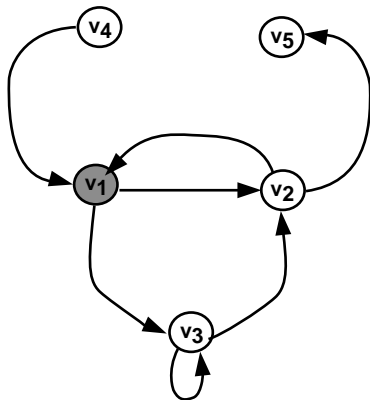


Execução do Algoritmo DFS



pilha

Execução do Algoritmo DFS

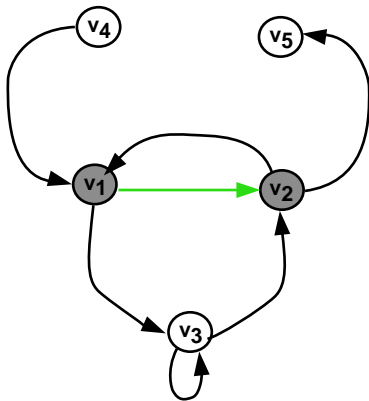


DFS visita v1



pilha

Execução do Algoritmo DFS

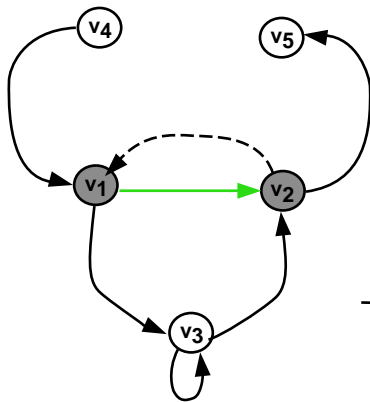


DFS visita v1
↳ DFS visita v2



pilha

Execução do Algoritmo DFS



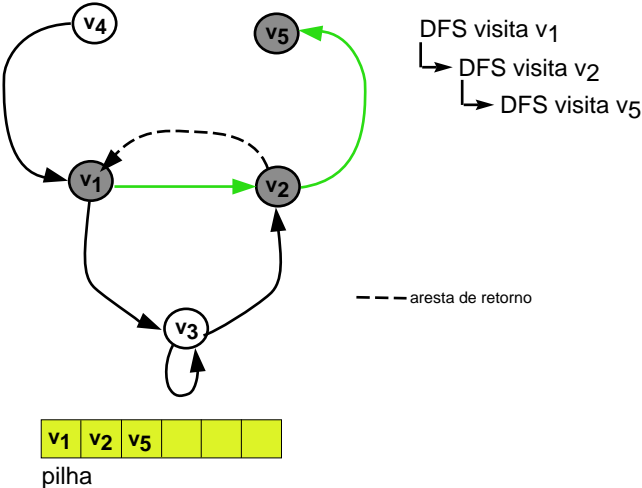
DFS visita v1
└─ DFS visita v2
 └─ DFS visita v1

--- aresta de retorno

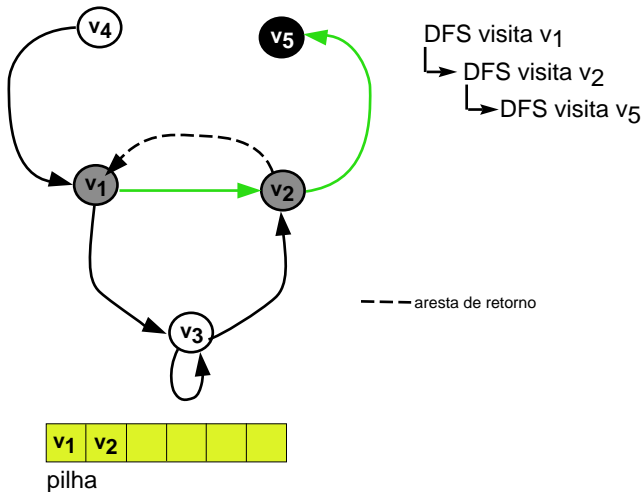


pilha

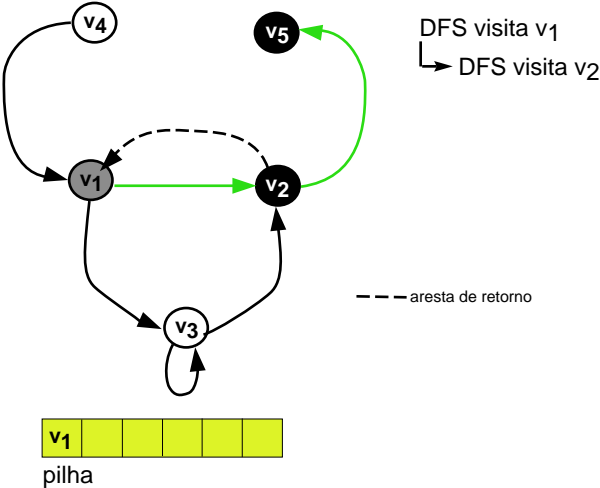
Execução do Algoritmo DFS



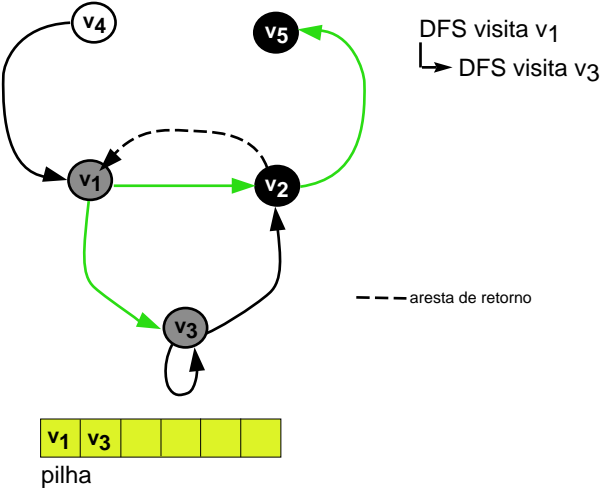
Execução do Algoritmo DFS



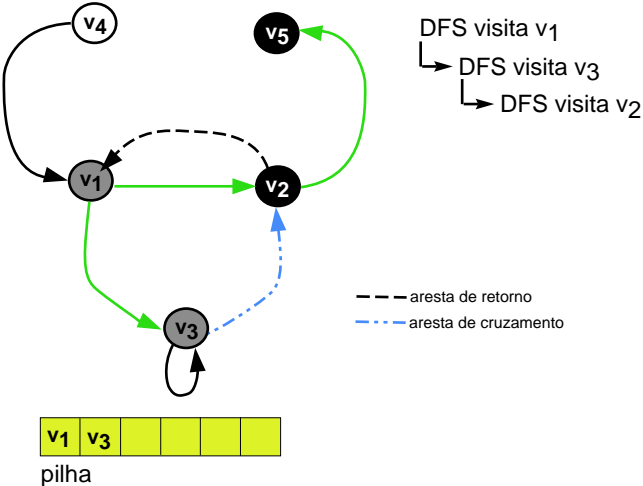
Execução do Algoritmo DFS



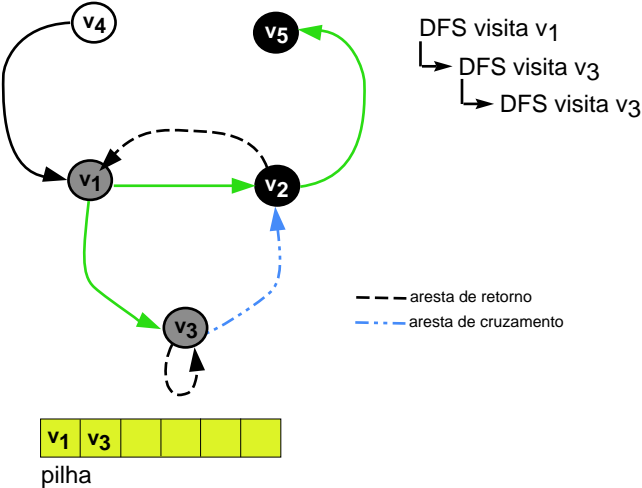
Execução do Algoritmo DFS



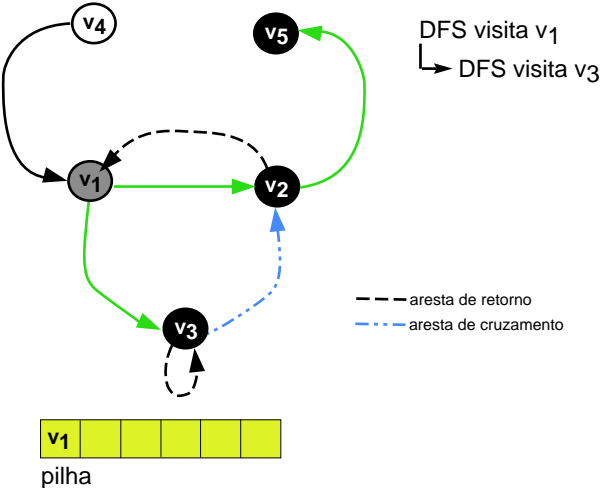
Execução do Algoritmo DFS



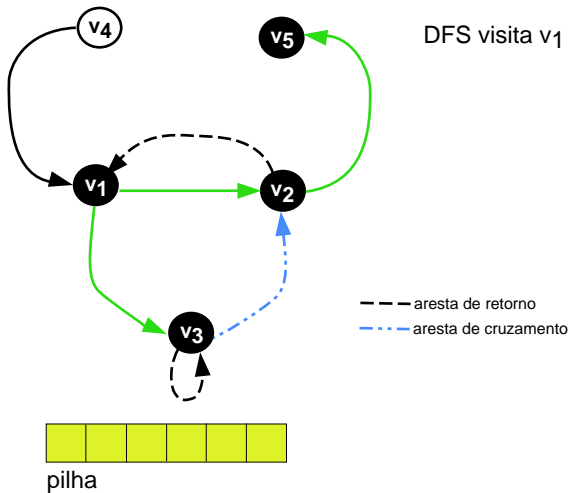
Execução do Algoritmo DFS



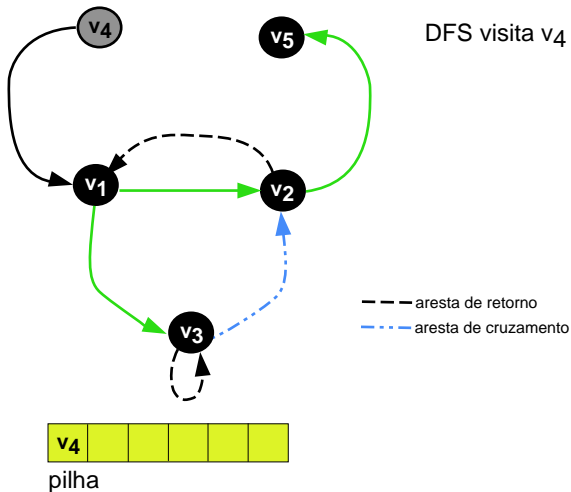
Execução do Algoritmo DFS



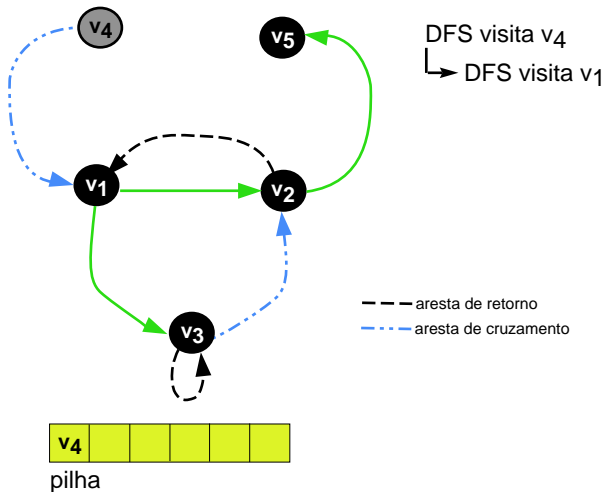
Execução do Algoritmo DFS



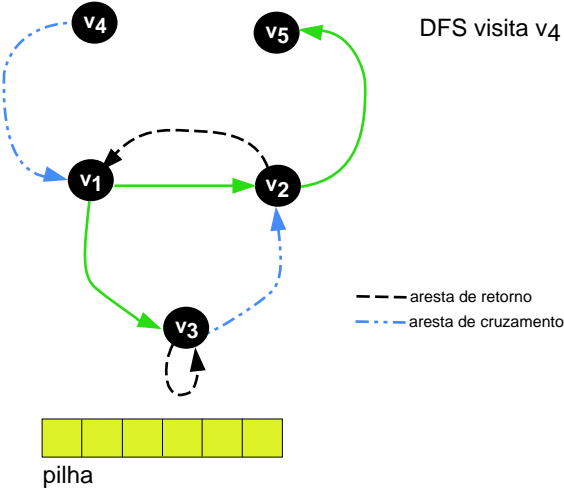
Execução do Algoritmo DFS



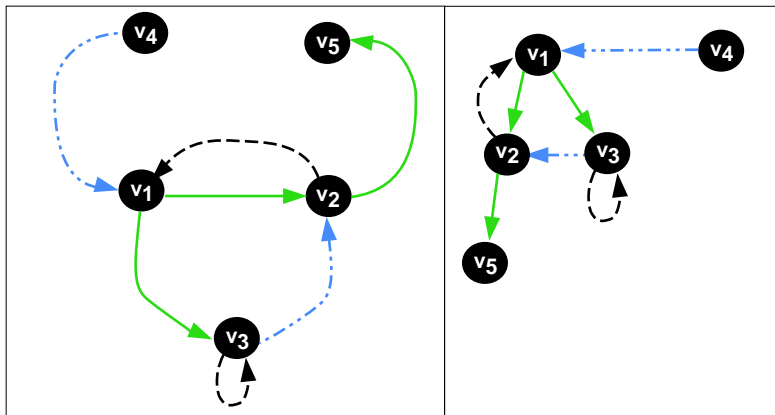
Execução do Algoritmo DFS



Execução do Algoritmo DFS



Execução do Algoritmo DFS



Propriedades da DFS

- O resultado da DFS é um grafo de precedência.
- Pode-se identificar ciclos no grafo (usando as arestas de retorno (B)).

Algoritmo DFS

Algoritmo DFS(Adj[]) // Adj[] é a lista de adjacências de G

para cada vértice $u \in V(G)$ faça:

$cor[u] = \text{branco}$;

$pred[u] = \text{nulo}$; // predecessor de u

para cada vértice $u \in V[G]$ faça:

 se $cor[u] = \text{branco}$ então

 DFS-Visita(u);

DFS-Visita($u, \text{Adj}[]$)

$\text{cor}[u] = \text{cinza};$

para cada $v \in \text{Adj}[u]$ faça:

 se $\text{cor}[v] = \text{branco}$ então

$\text{pred}[v] = u;$

 DFS-Visita($v, \text{Adj}[]$);

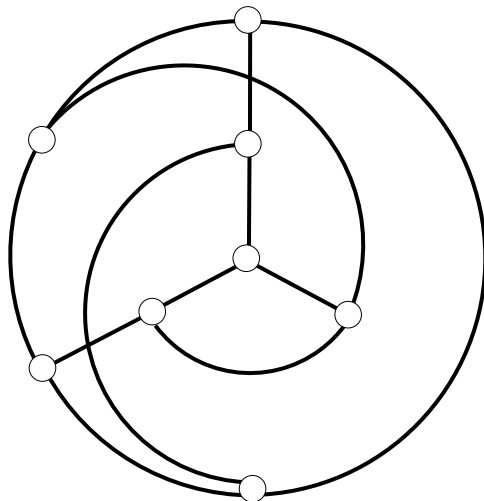
$\text{cor}[u] = \text{preto};$

Tempo de Execução do Algoritmo DFS

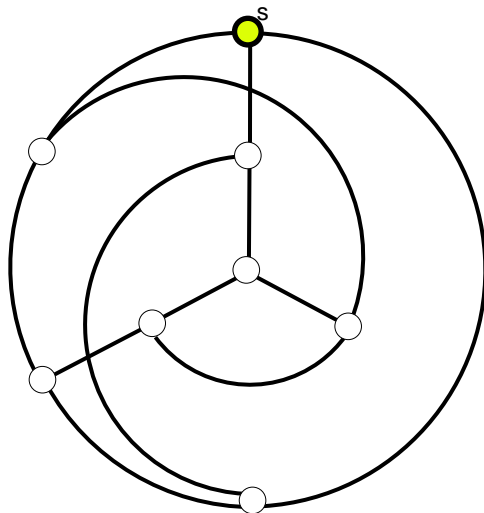
- Os laços da DFS tomam tempo $\Theta(|V|)$ cada, excluindo o tempo de execução de DFS-Visita.
- DFS-Visita é chamado uma vez para cada vértice branco e pinta o vértice de cinza imediatamente.
- Para cada chamada DFS-Visita o laço interage sobre os adjacentes do vértice.
- Assim o custo somado de todas as chamadas de DFS-Visita é $\Theta(|E|)$.

Portanto, o Algoritmo DFS é $\Theta(|V| + |E|)$.

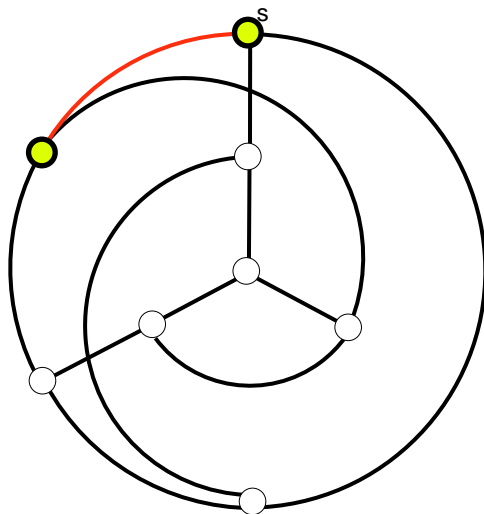
Propriedades - Arestas de Retorno



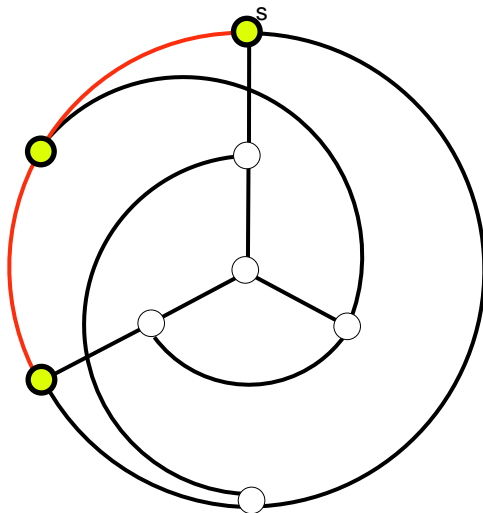
Propriedades - Arestas de Retorno



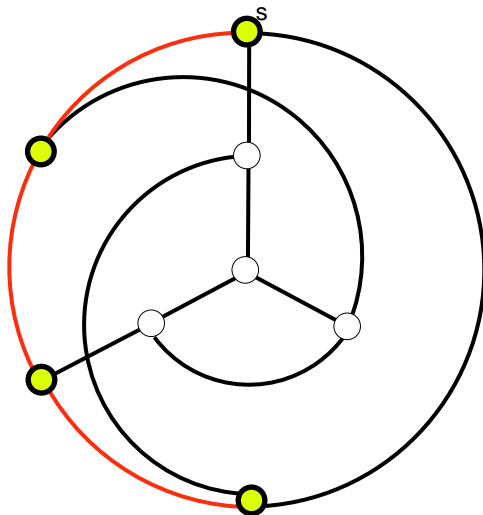
Propriedades - Arestas de Retorno



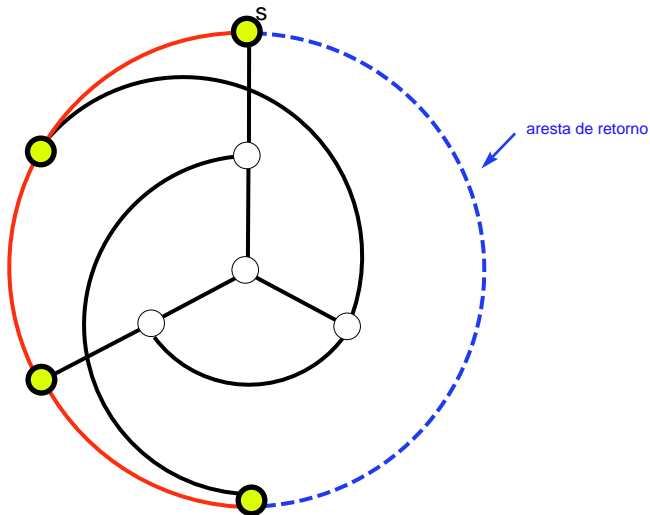
Propriedades - Arestas de Retorno



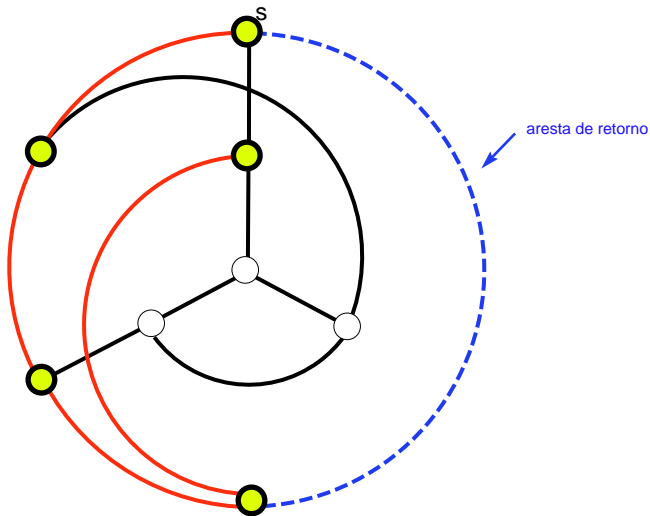
Propriedades - Arestas de Retorno



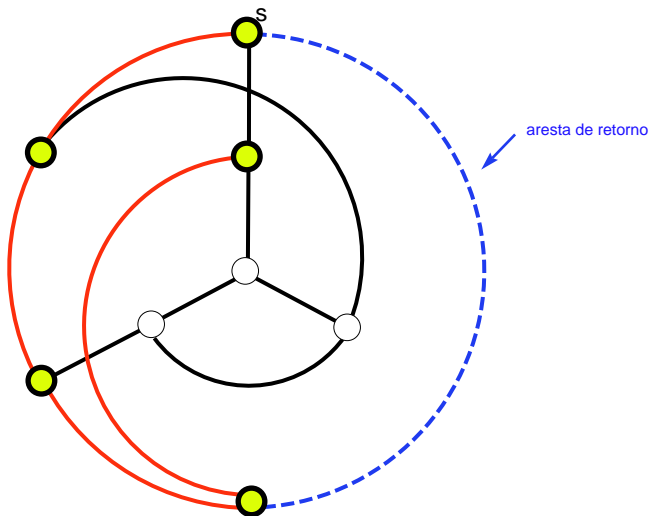
Propriedades - Arestas de Retorno



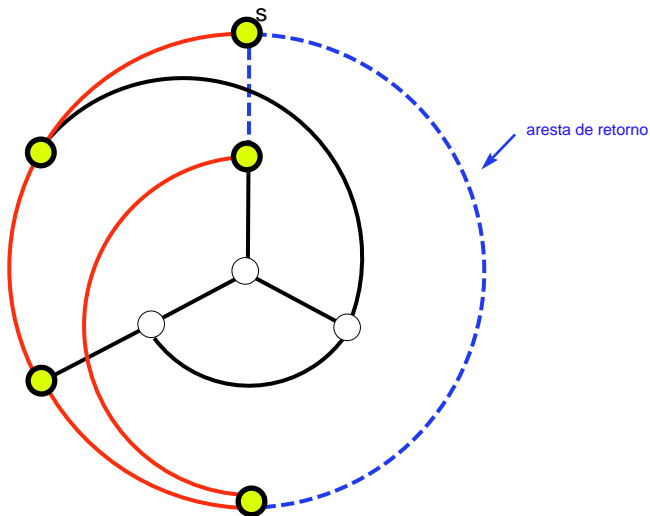
Propriedades - Arestas de Retorno



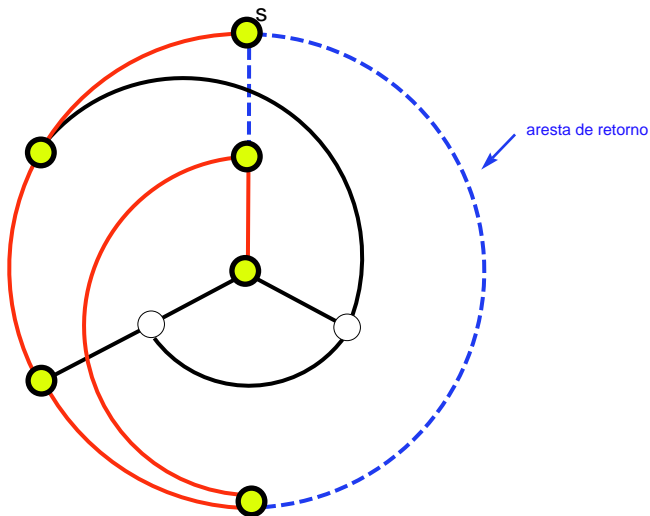
Propriedades - Arestas de Retorno



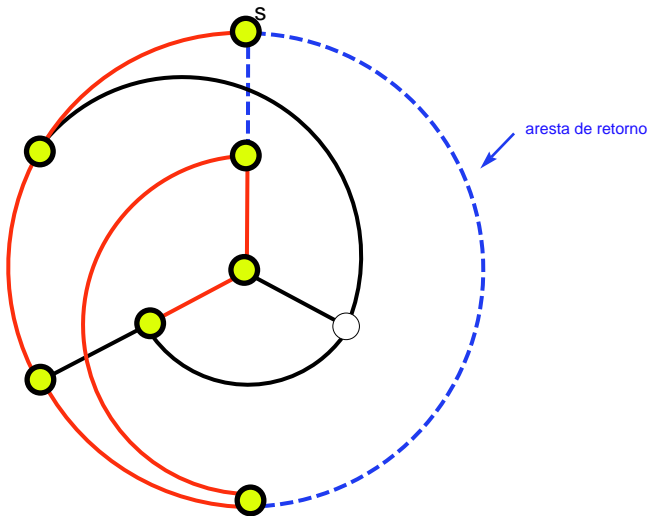
Propriedades - Arestas de Retorno



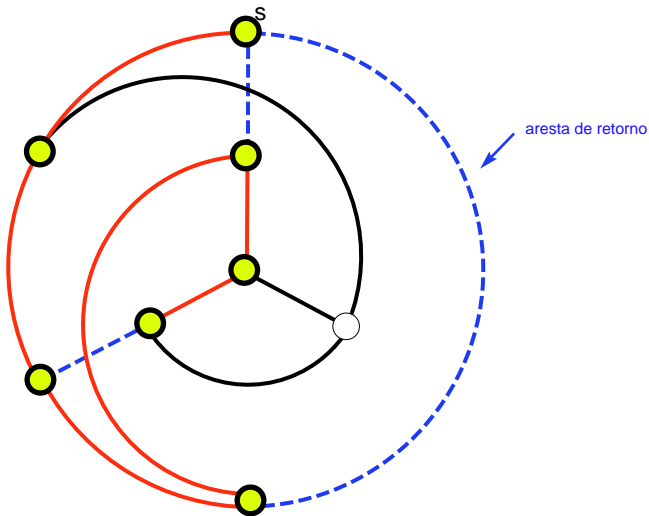
Propriedades - Arestas de Retorno



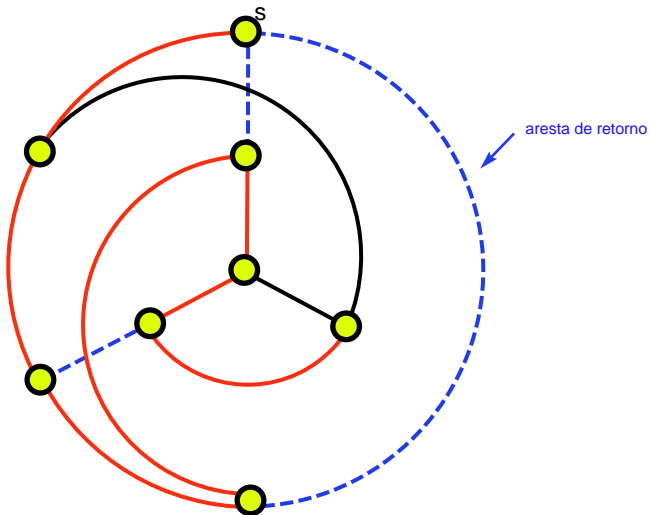
Propriedades - Arestas de Retorno



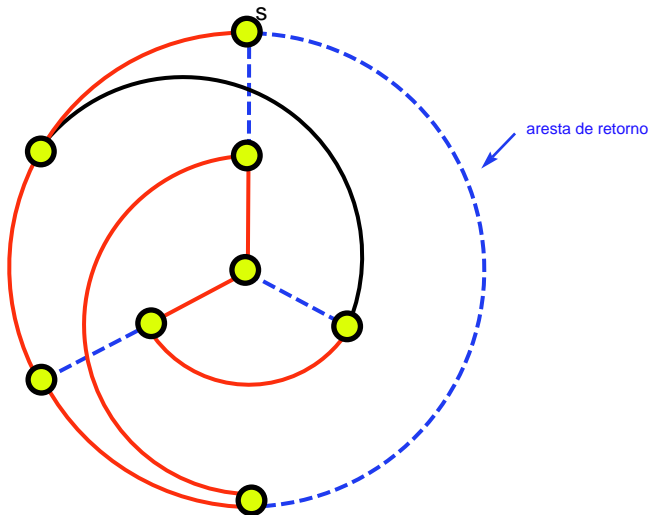
Propriedades - Arestas de Retorno



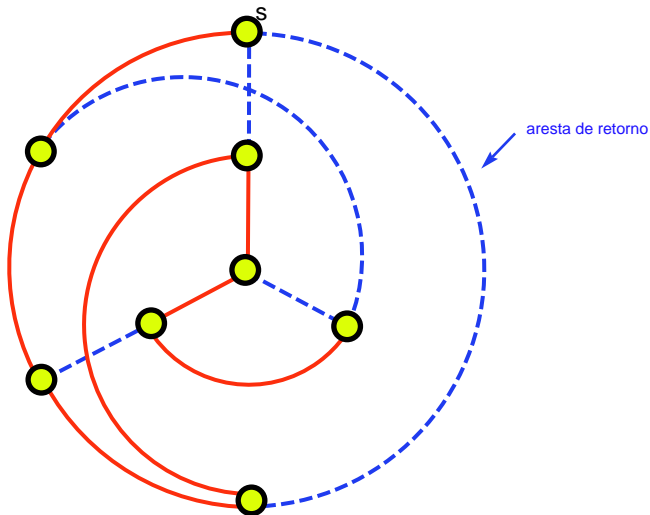
Propriedades - Arestas de Retorno



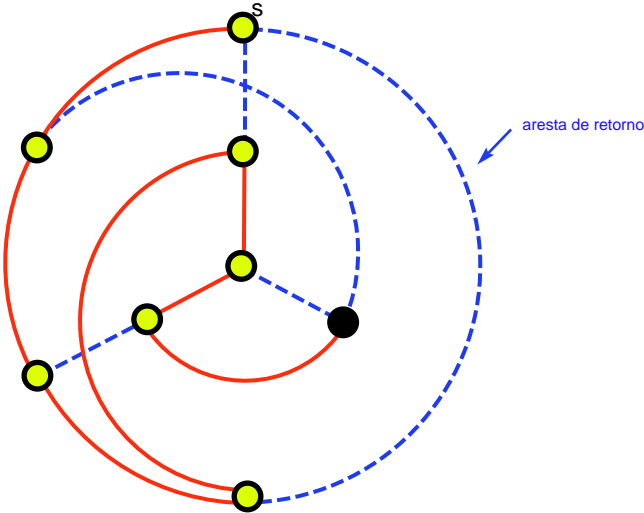
Propriedades - Arestas de Retorno



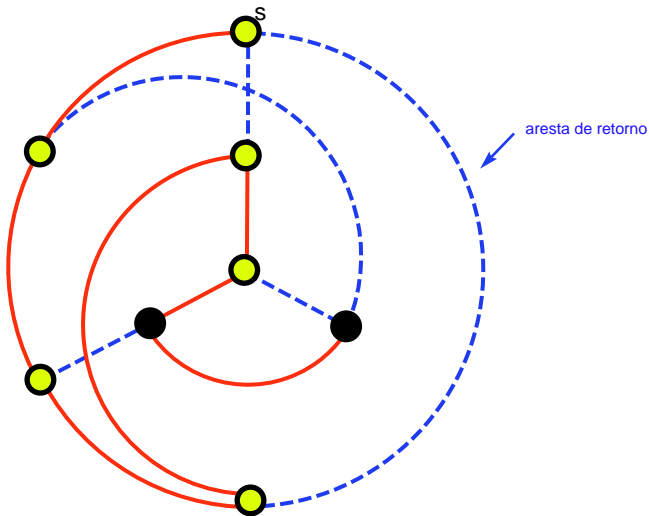
Propriedades - Arestas de Retorno



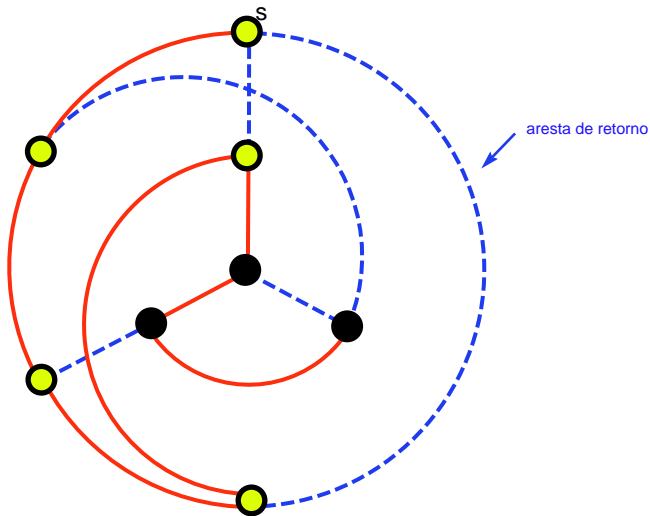
Propriedades - Arestas de Retorno



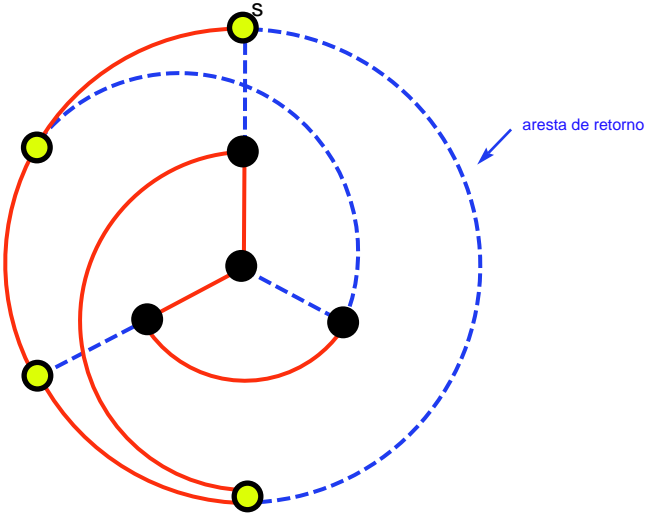
Propriedades - Arestas de Retorno



Propriedades - Arestas de Retorno



Propriedades - Arestas de Retorno



Propriedades - Arestas de Retorno

