

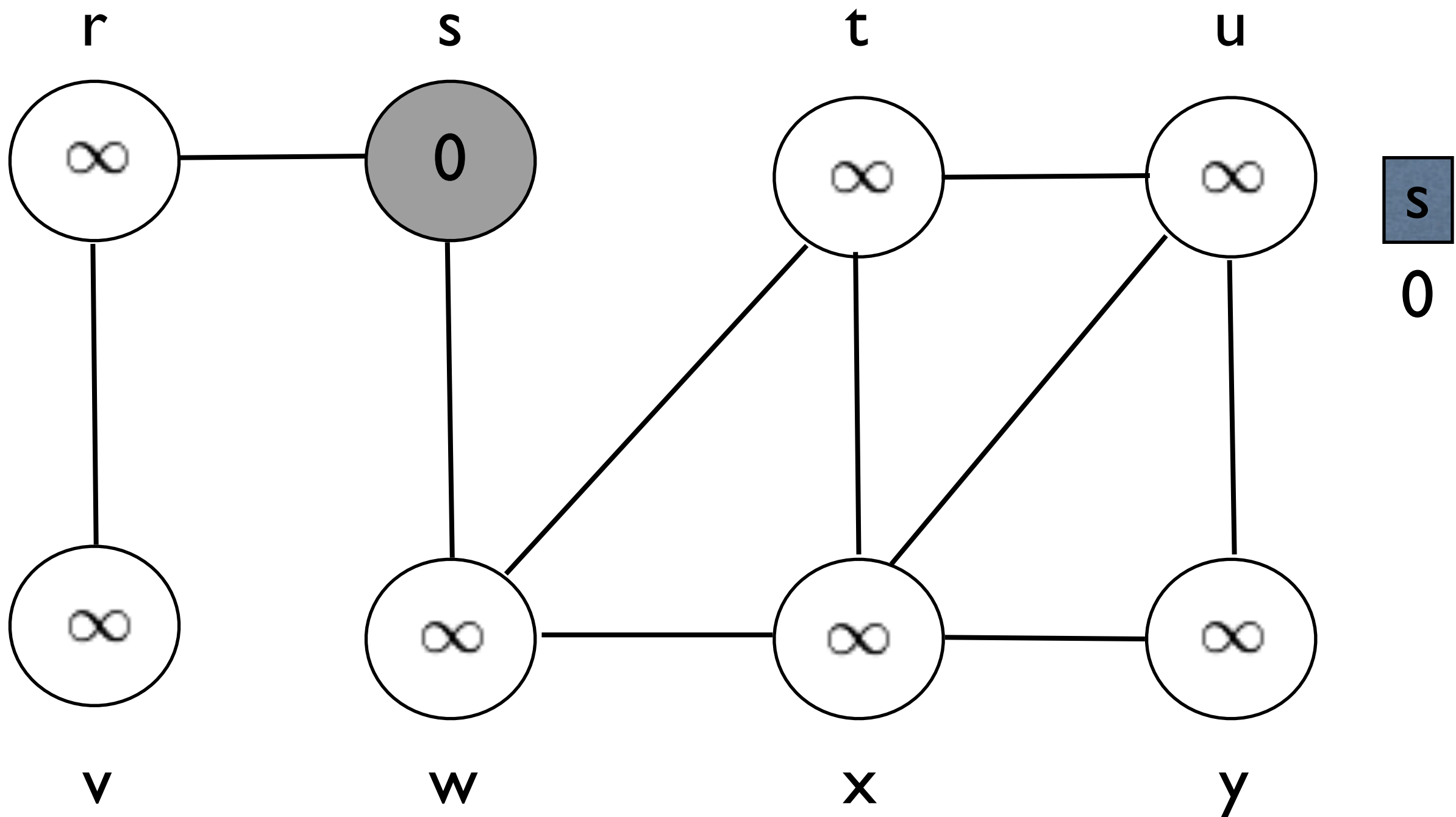
# Algoritmos Elementares

Leandro G. M. Alvim

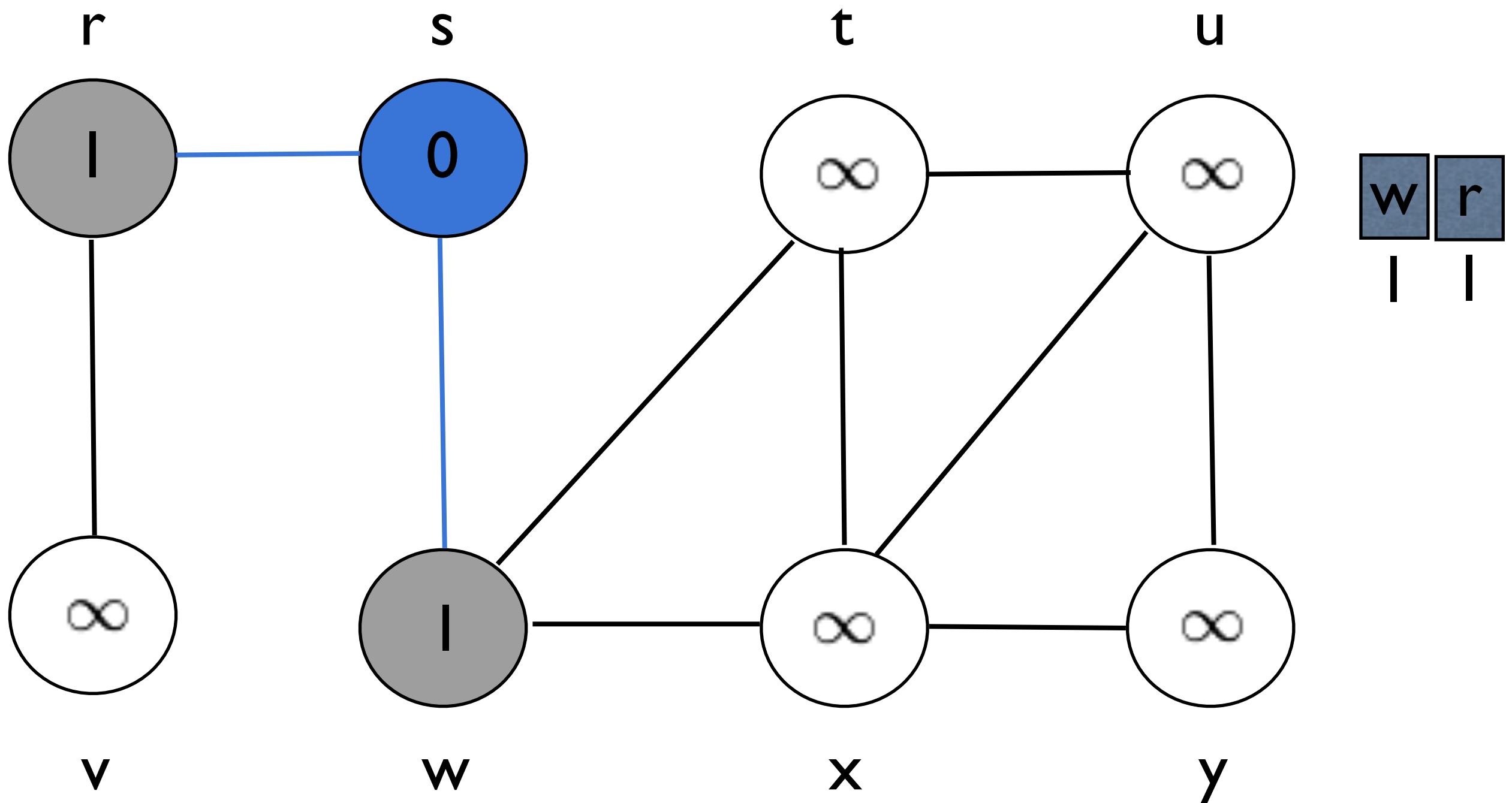
# Busca em Largura

- Caminho Mínimo
  - s até todos os outros
  - Grafo com pesos unitários

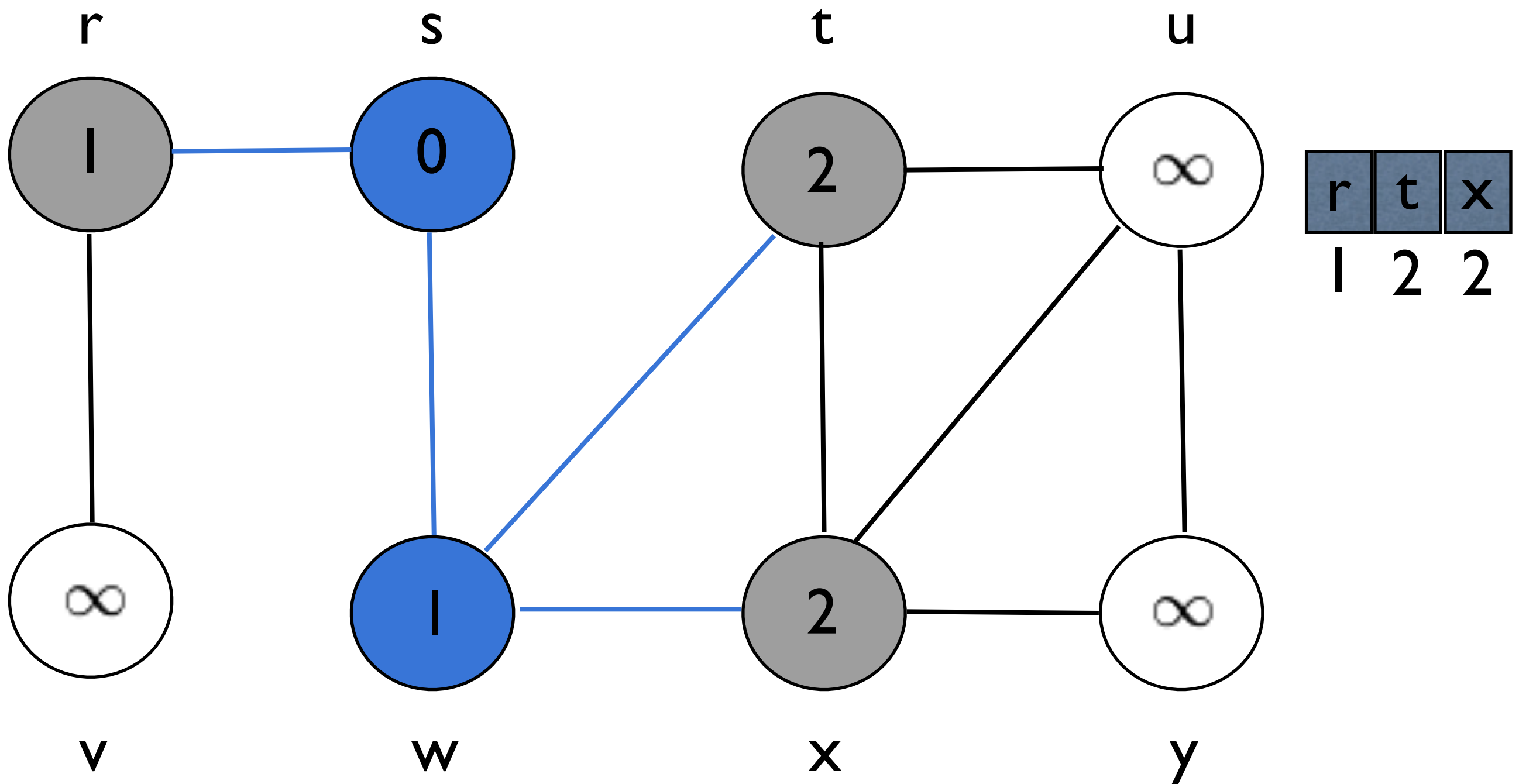
# Busca em Largura



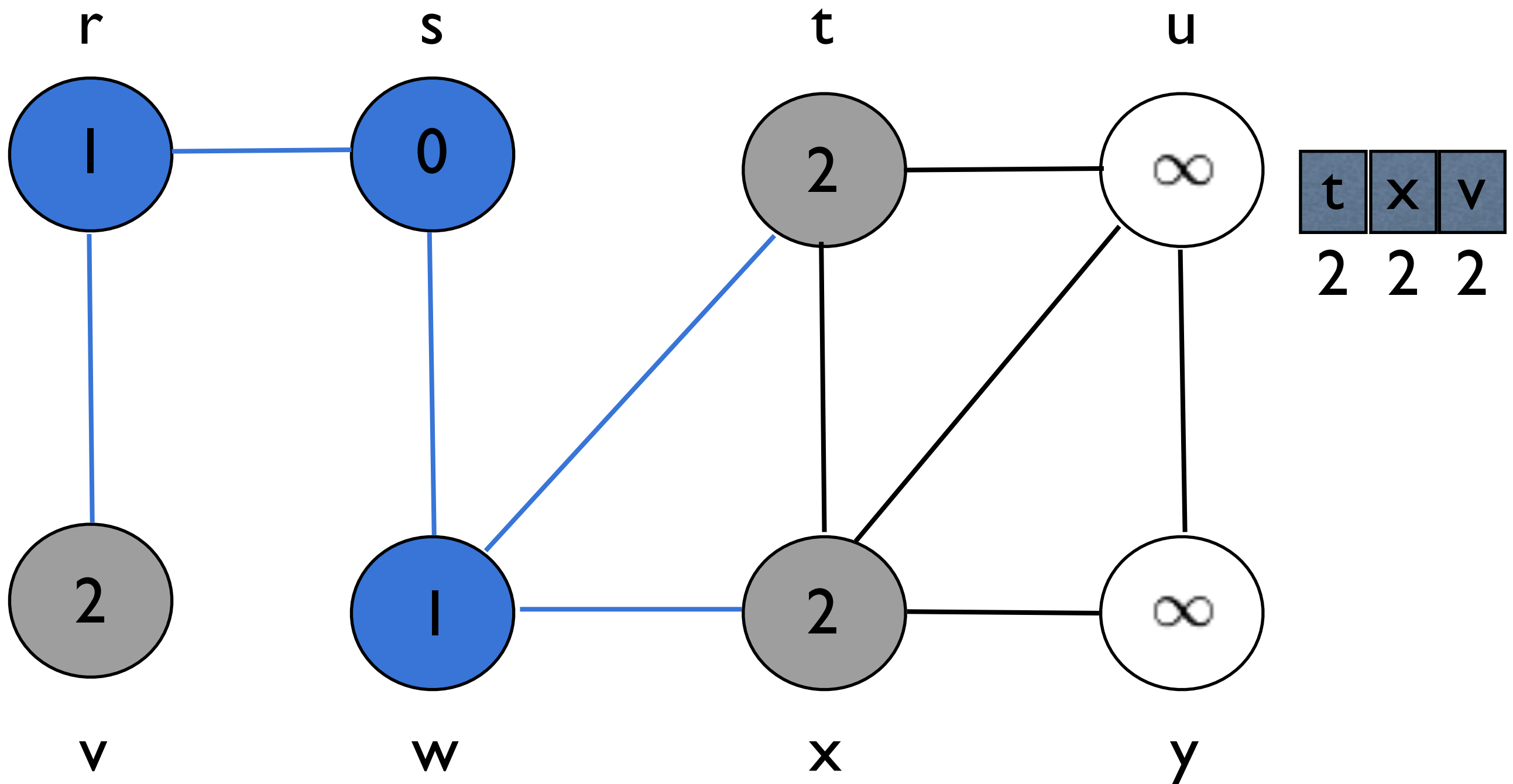
# Busca em Largura



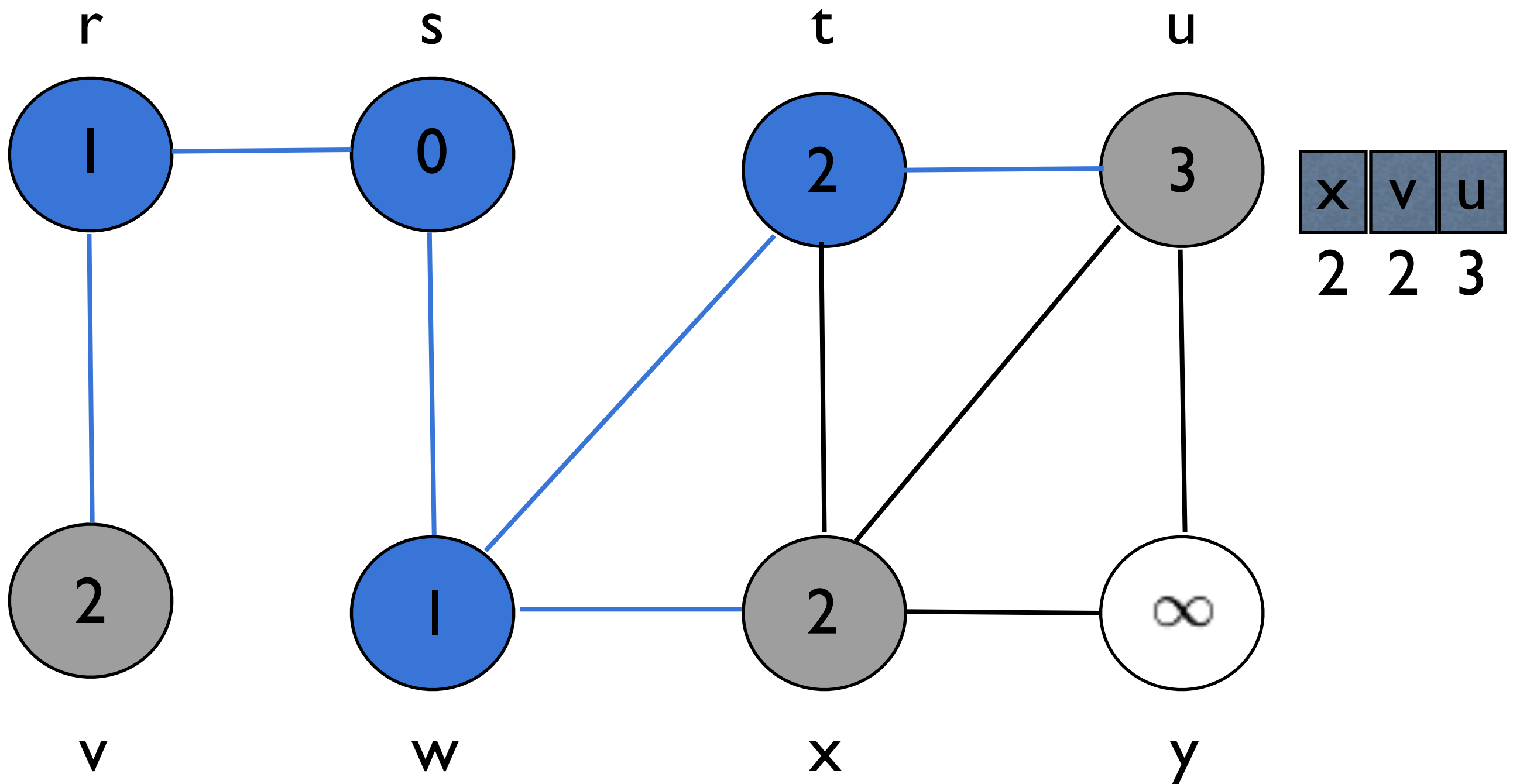
# Busca em Largura



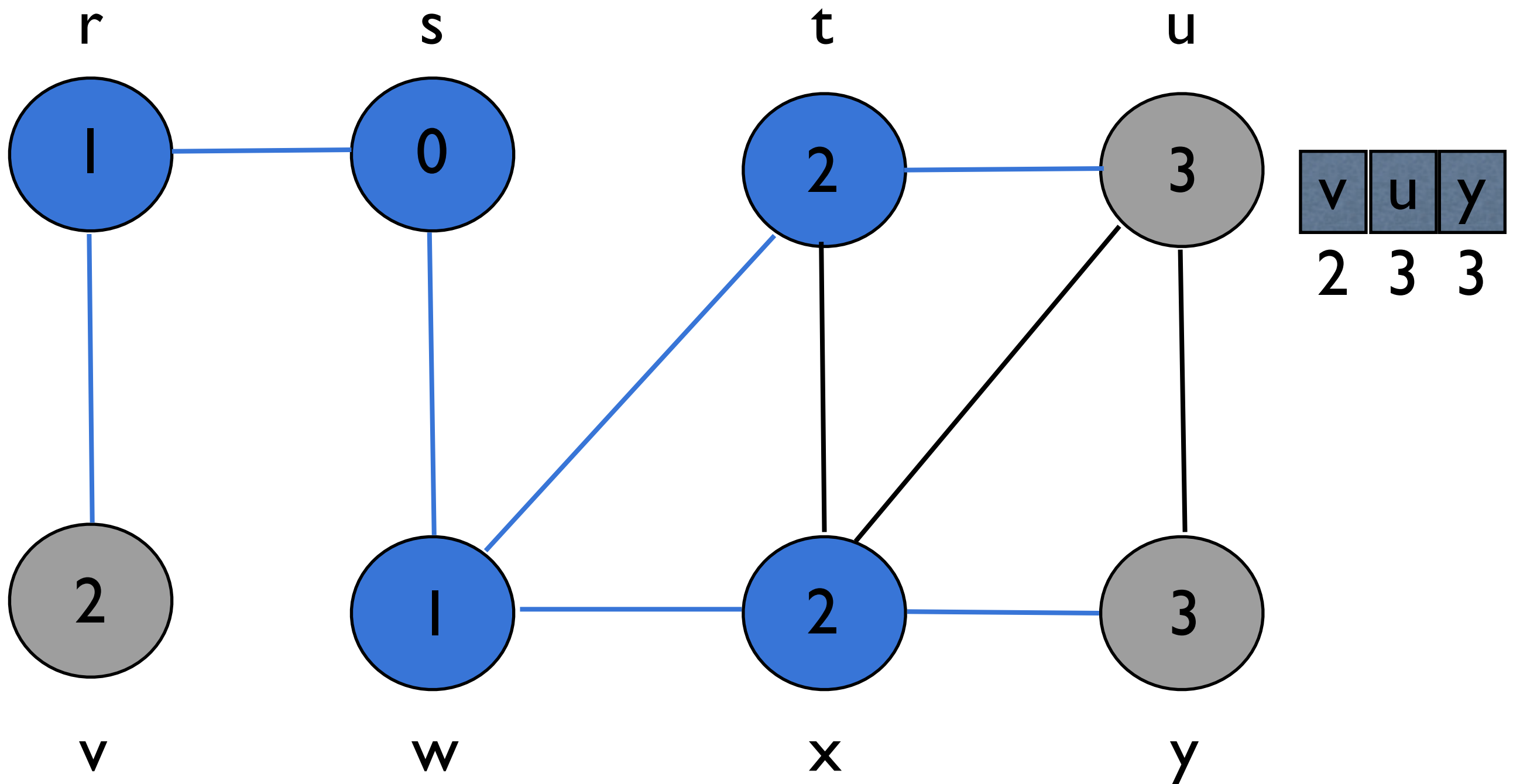
# Busca em Largura



# Busca em Largura

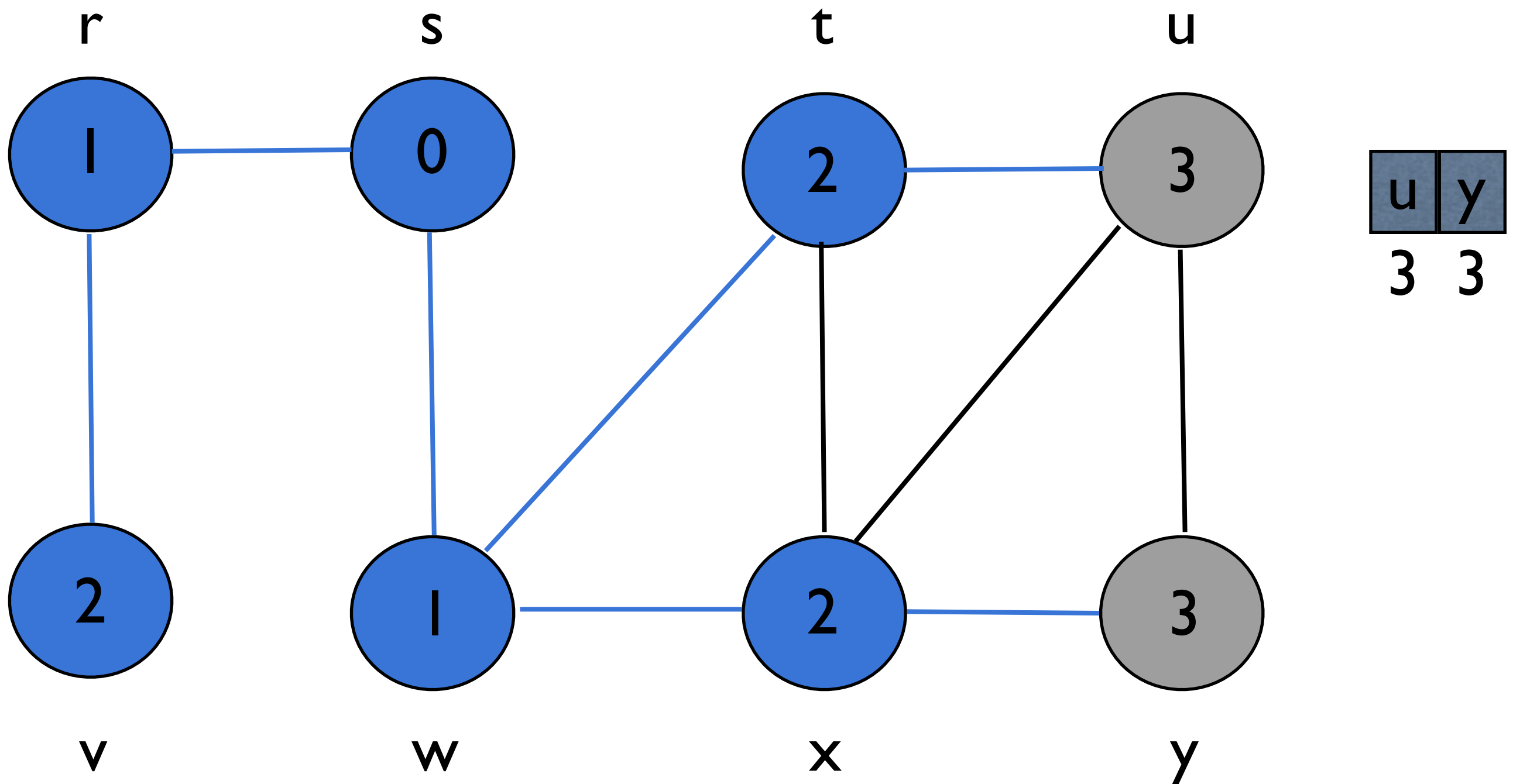


# Busca em Largura

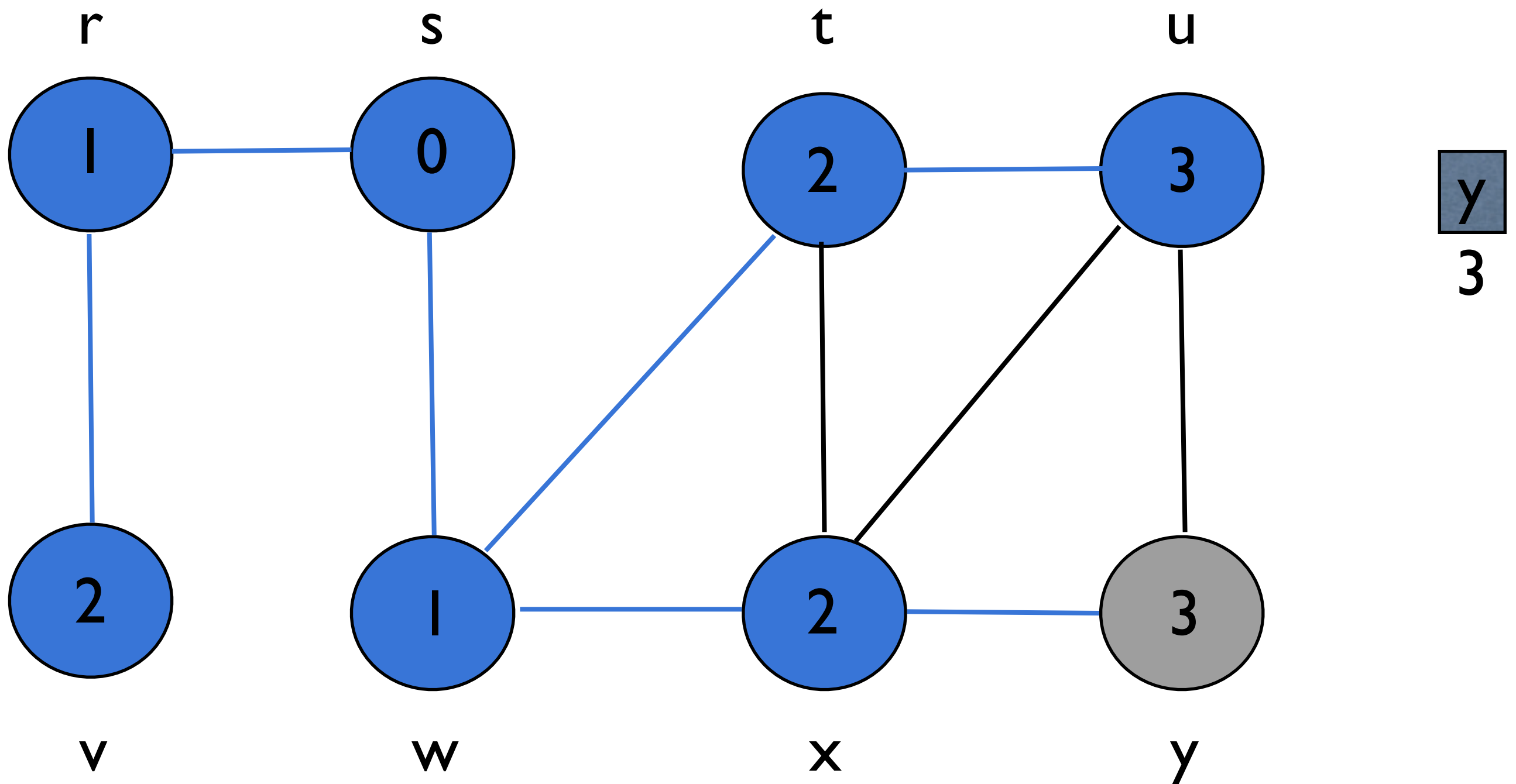




# Busca em Largura

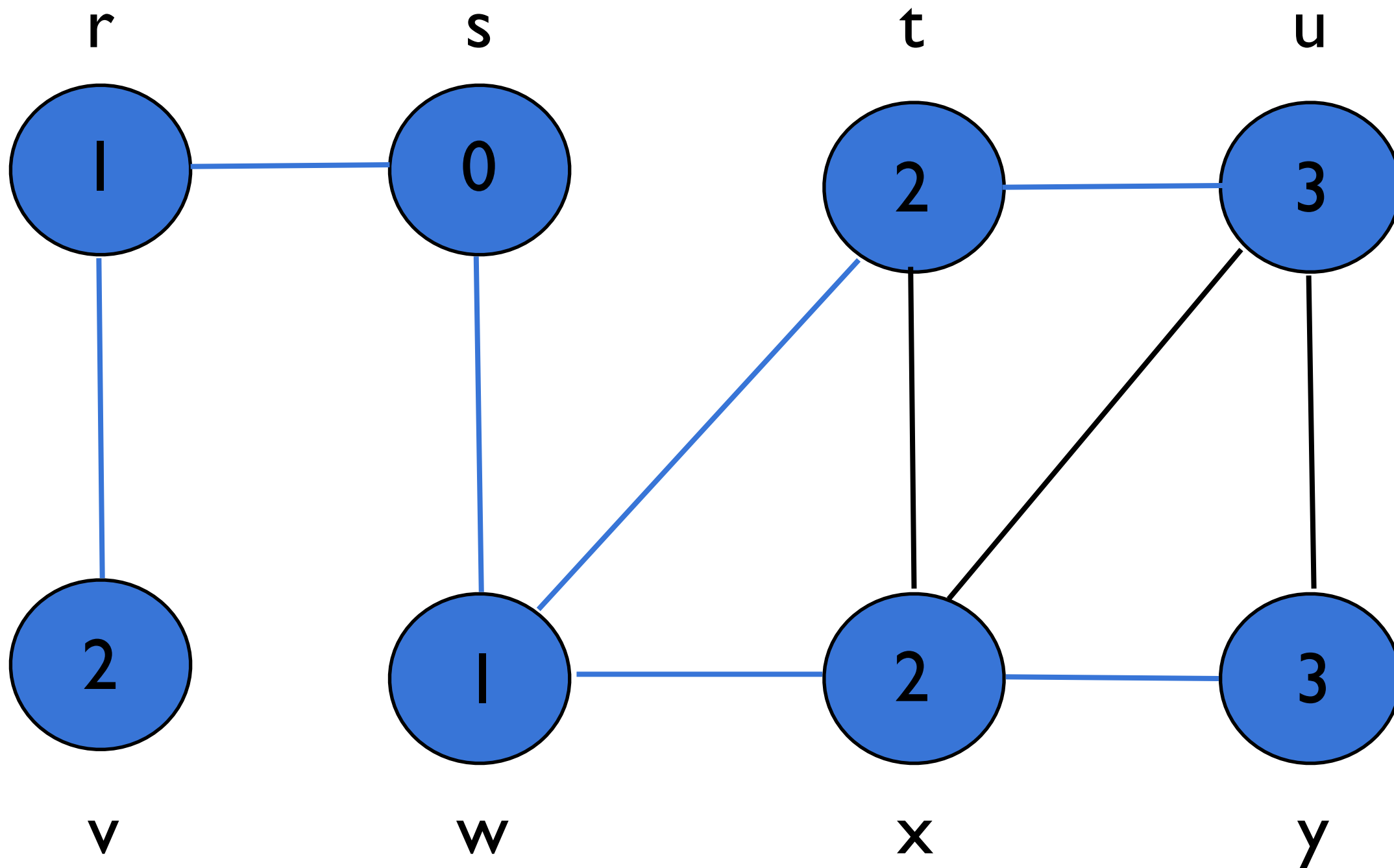


# Busca em Largura



# Busca em Largura

Árvore de Busca em Largura



# Árvore de Busca em Largura

- $G_{\text{pred}} = (V_{\text{pred}}, E_{\text{pred}}),$
- $V_{\text{pred}} = \{v \in V \mid \text{pred}[v] \neq \text{nil}\} \cup \{s\}$
- $E_{\text{pred}} = \{(\text{pred}[v], v) \mid v \in V_{\text{pred}} - \{s\}\}$

# Caminho Mínimo

**Para cada**  $u \in V[G] - \{s\}$  **faça**

$cor[u] = \text{branca}$

$dist[u] = \infty$

$pred[u] = \text{nil}$

$cor[s] = \text{azul}$

$dist[s] = 0$

$pred[s] = \text{nil}$

$fila = \{s\}$

**Enquanto**  $fila \neq \{\}$  **Faça**

$u = \text{cabeça}(fila)$

**Para cada**  $v \in Adj[u]$  **Faça**

**If**  $cor[v] == \text{branca}$  **Então**

$cor[v] = \text{cinza}$

$d[v] = d[u] + l$

$pred[v] = u$

$\text{enfila}(fila, v)$

$\text{desenfila}(fila)$

$cor[u] = \text{azul}$

# Imprimindo Caminho Mínimo

Imprime\_caminho( $G, s, v$ )

**Se**  $v = s$  **então**

imprimir  $s$

**Senão Se**  $\text{pred}[v] = \text{nil}$  **então**

imprimir “Sem caminho de ”,  $s$ , “ para ”,  $v$

**Senão**

imprime\_caminho( $G, s, \text{pred}[v]$ )

imprimir  $v$

# Análise de Desempenho

- Inicialização
  - Marcar os vértices  $O(|V|)$
- Fila
  - Enfilar, Desenfilar  $O(1)$
- Vizinhos
  - Estrutura de Adjacência  $O(|E|)$
  - Matriz de Adjacência  $O(|V|^2)$

# Análise de Desempenho

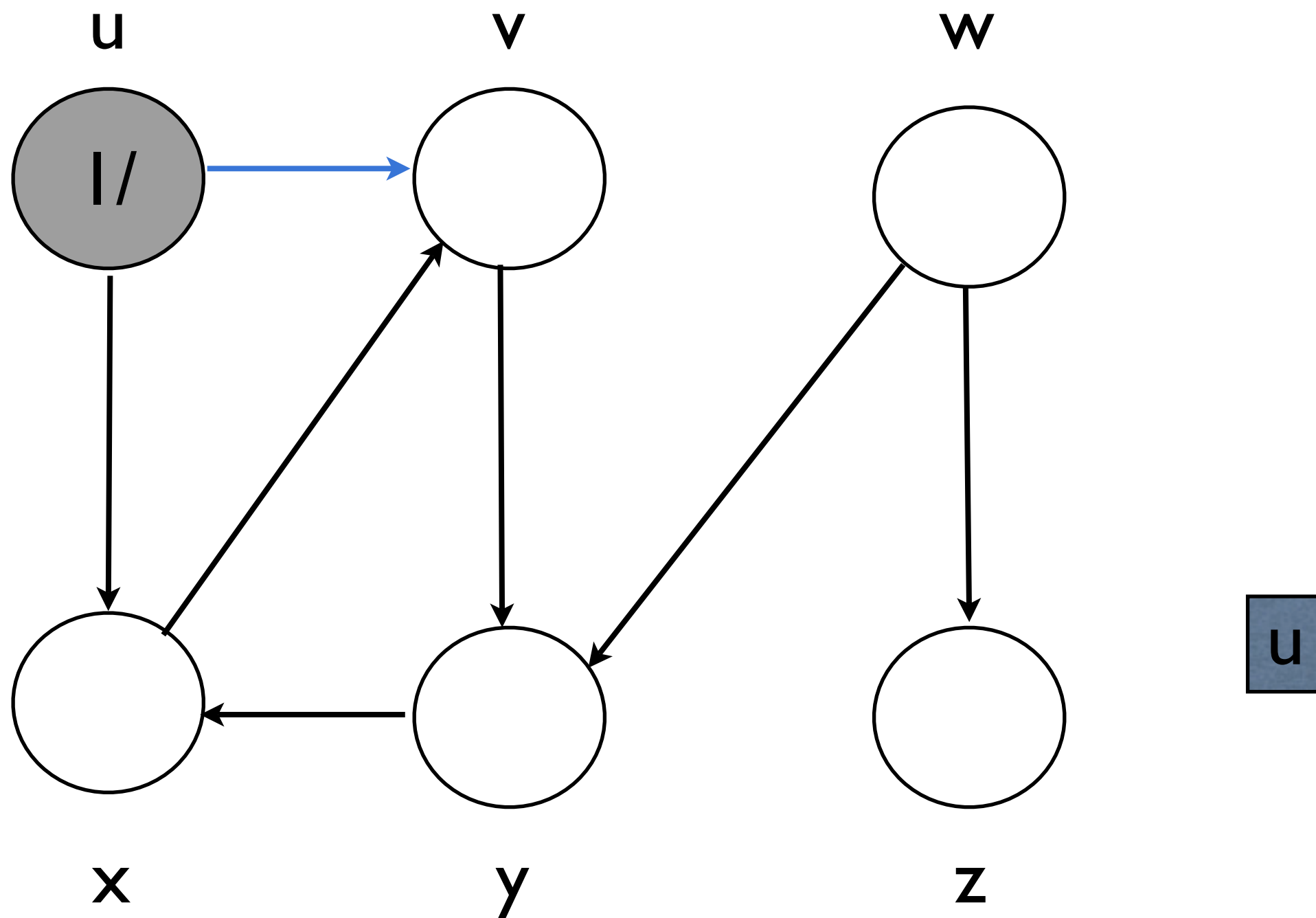
- Estrutura de Adjacência  $O(|V|+|E|)$
- Matriz de Adjacência  $O(|V|^2)$



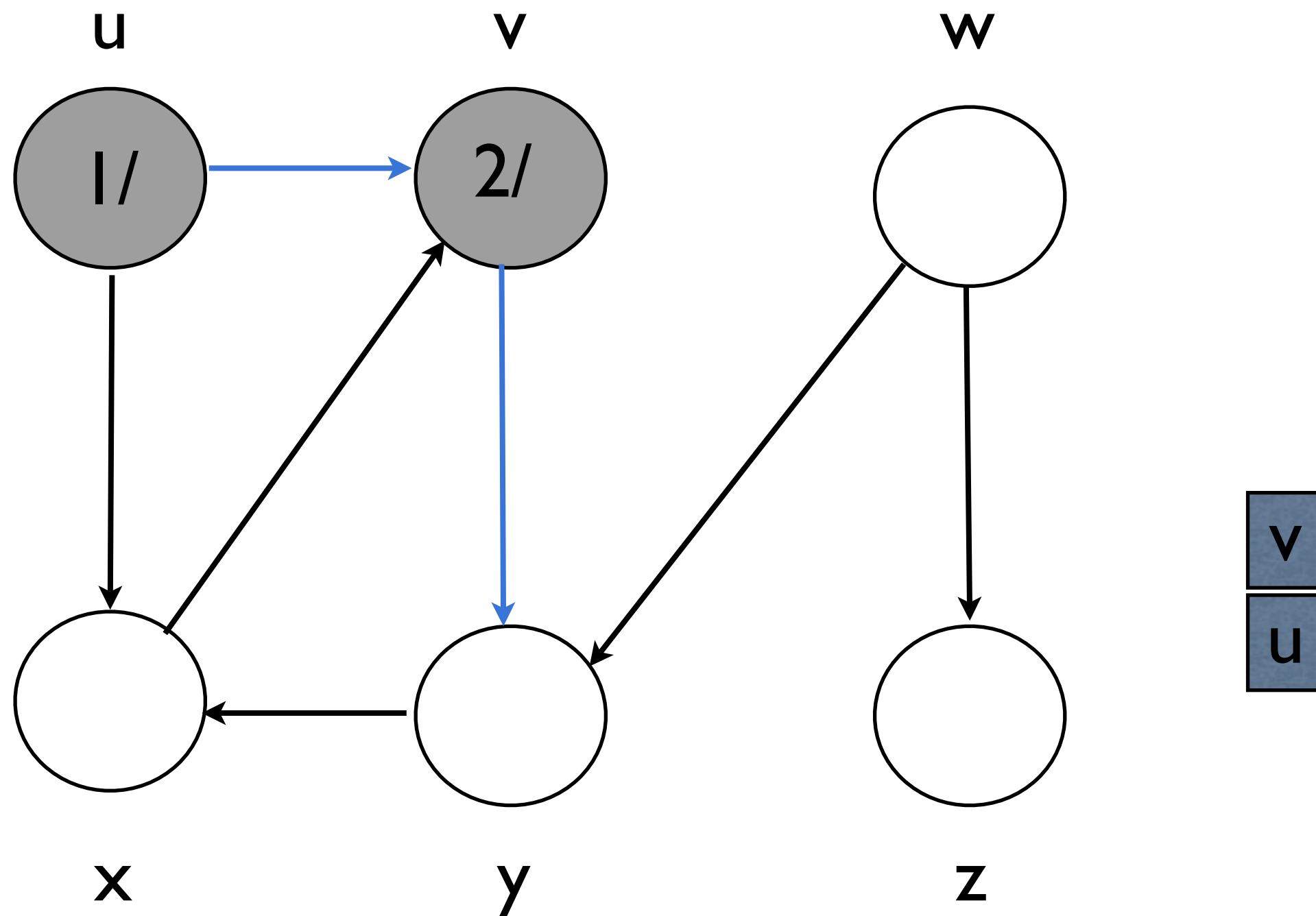
# Problemas

- Componente conexa pertencente a  $v$
- Caminho Mínimo (pesos unitários)
- Busca por chave

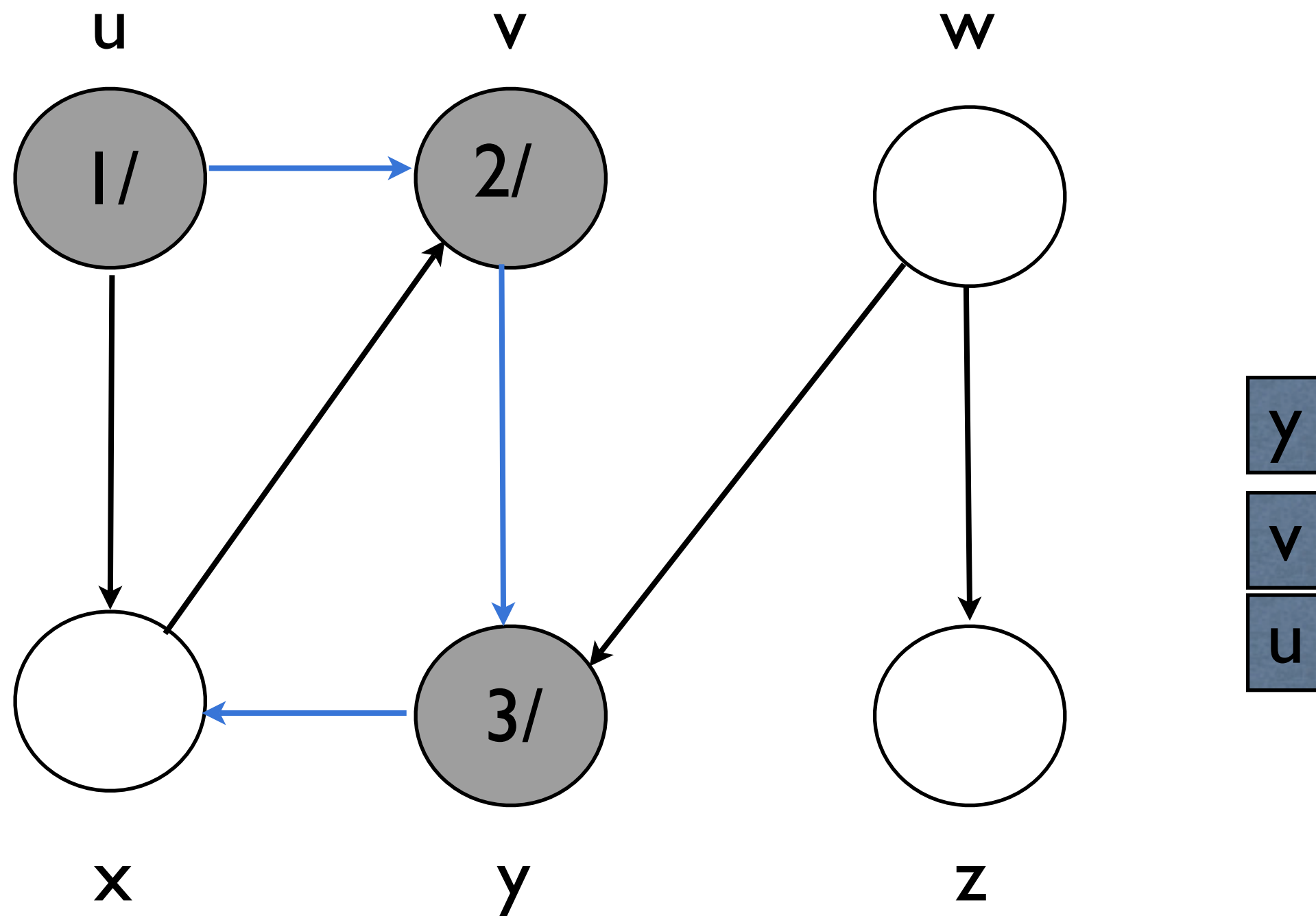
# Busca em Profundidade



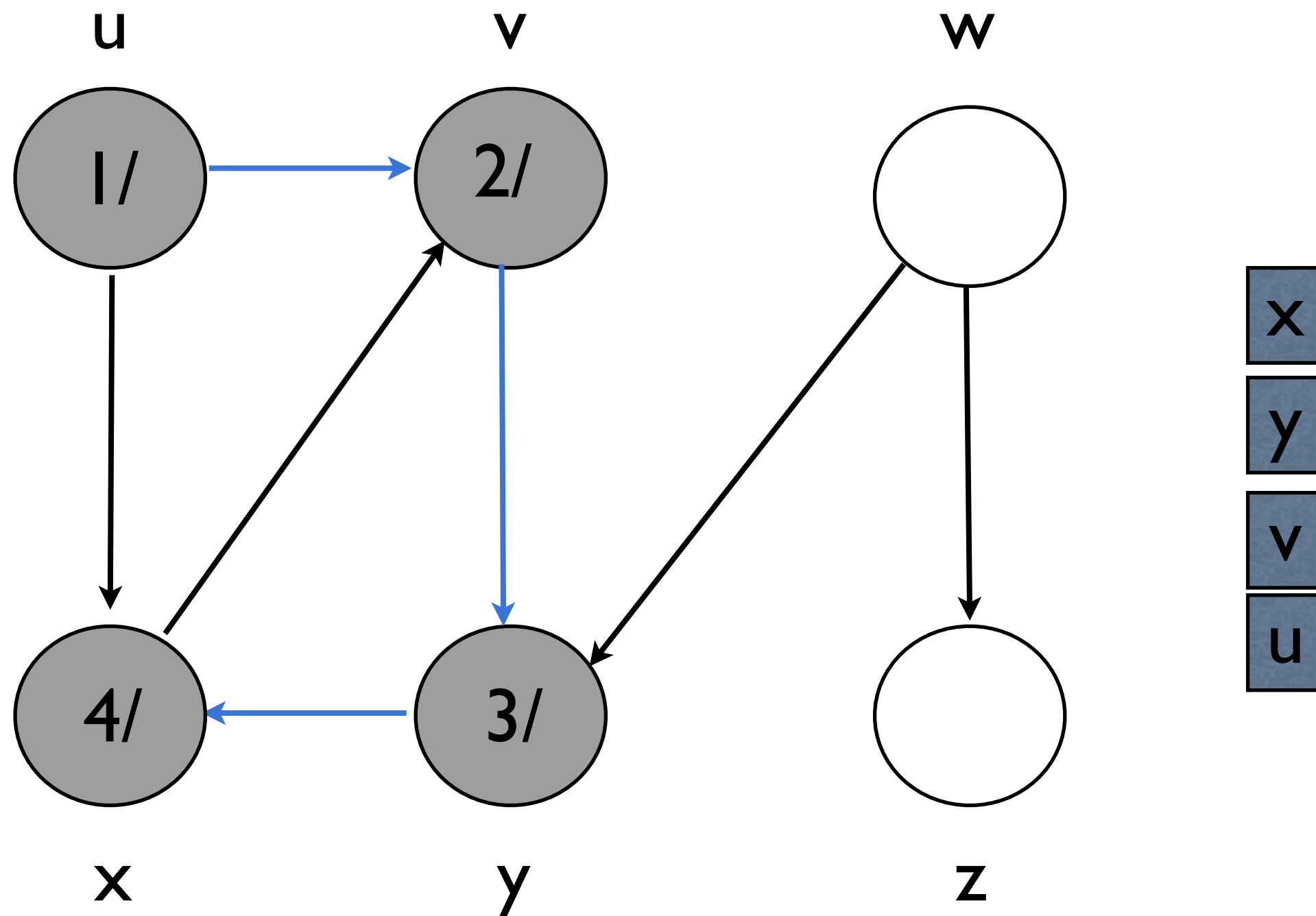
# Busca em Profundidade



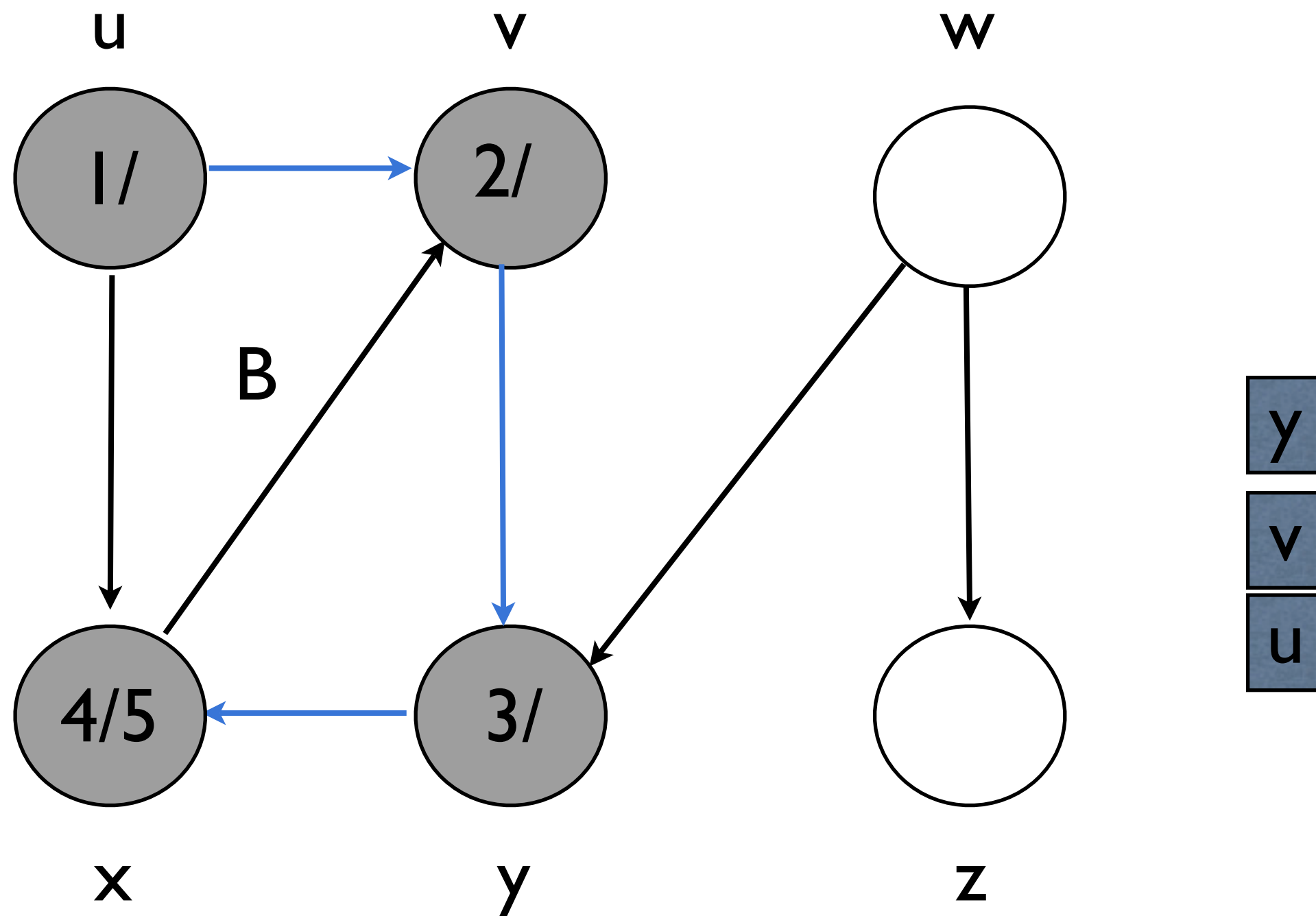
# Busca em Profundidade



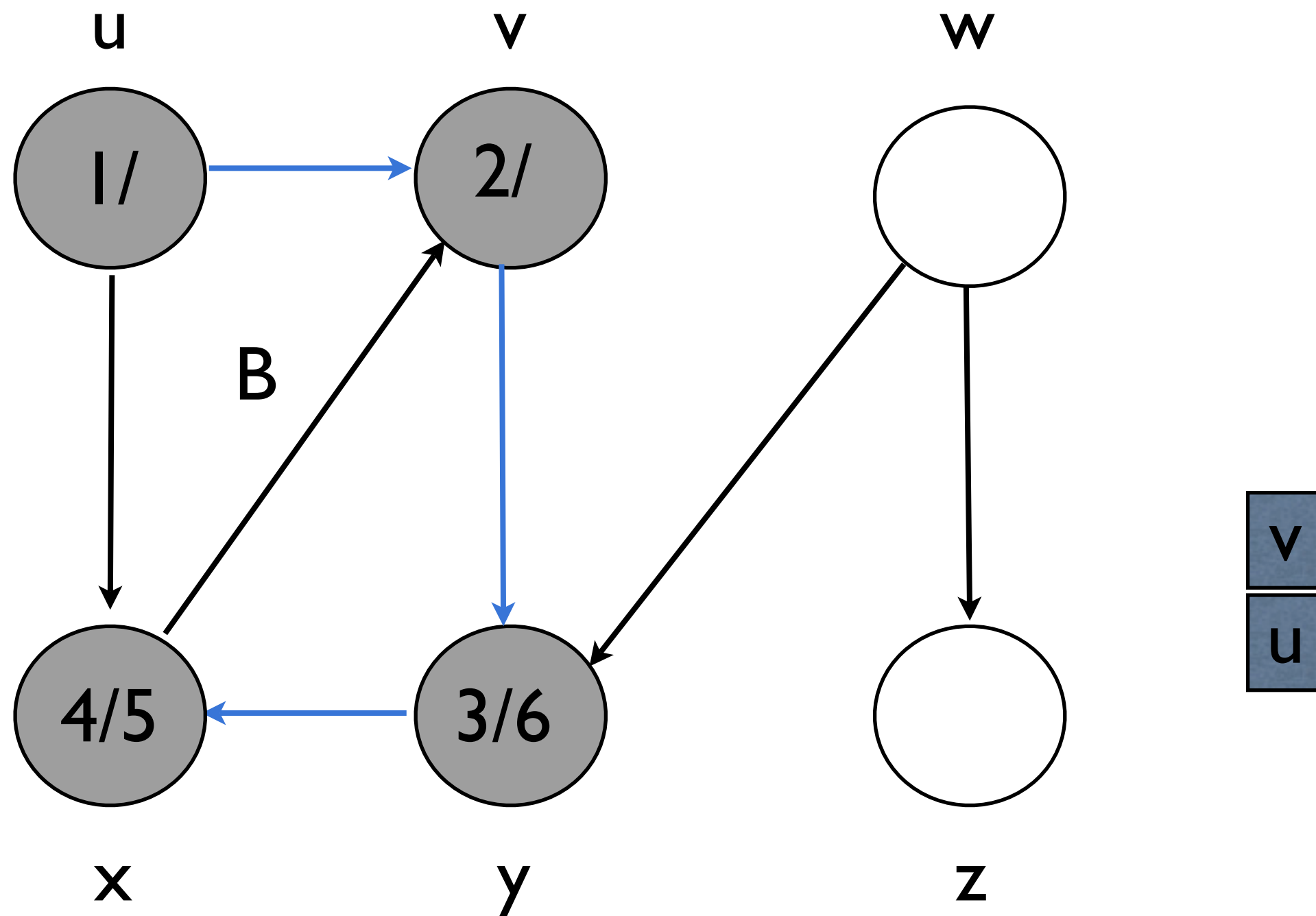
# Busca em Profundidade



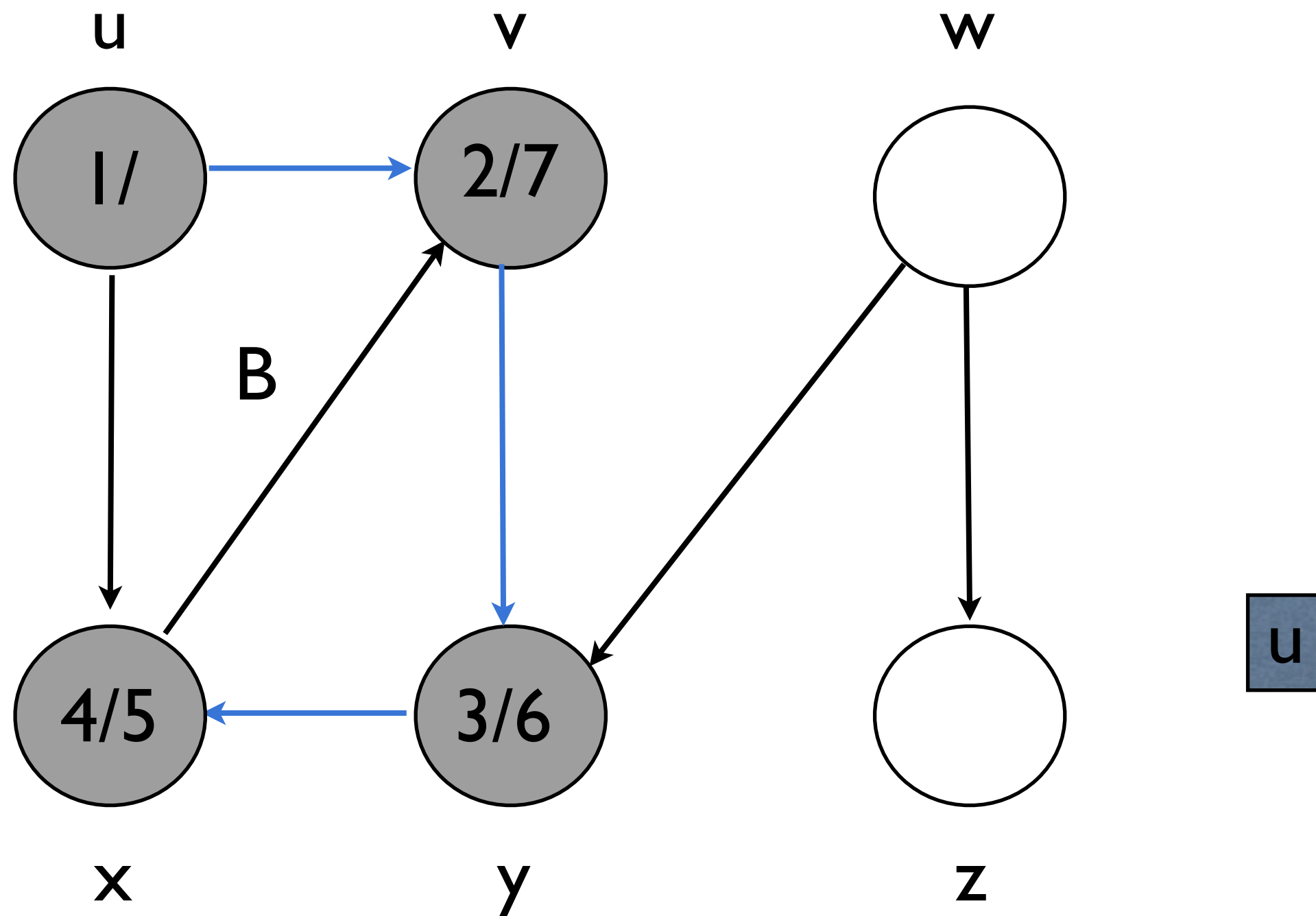
# Busca em Profundidade



# Busca em Profundidade

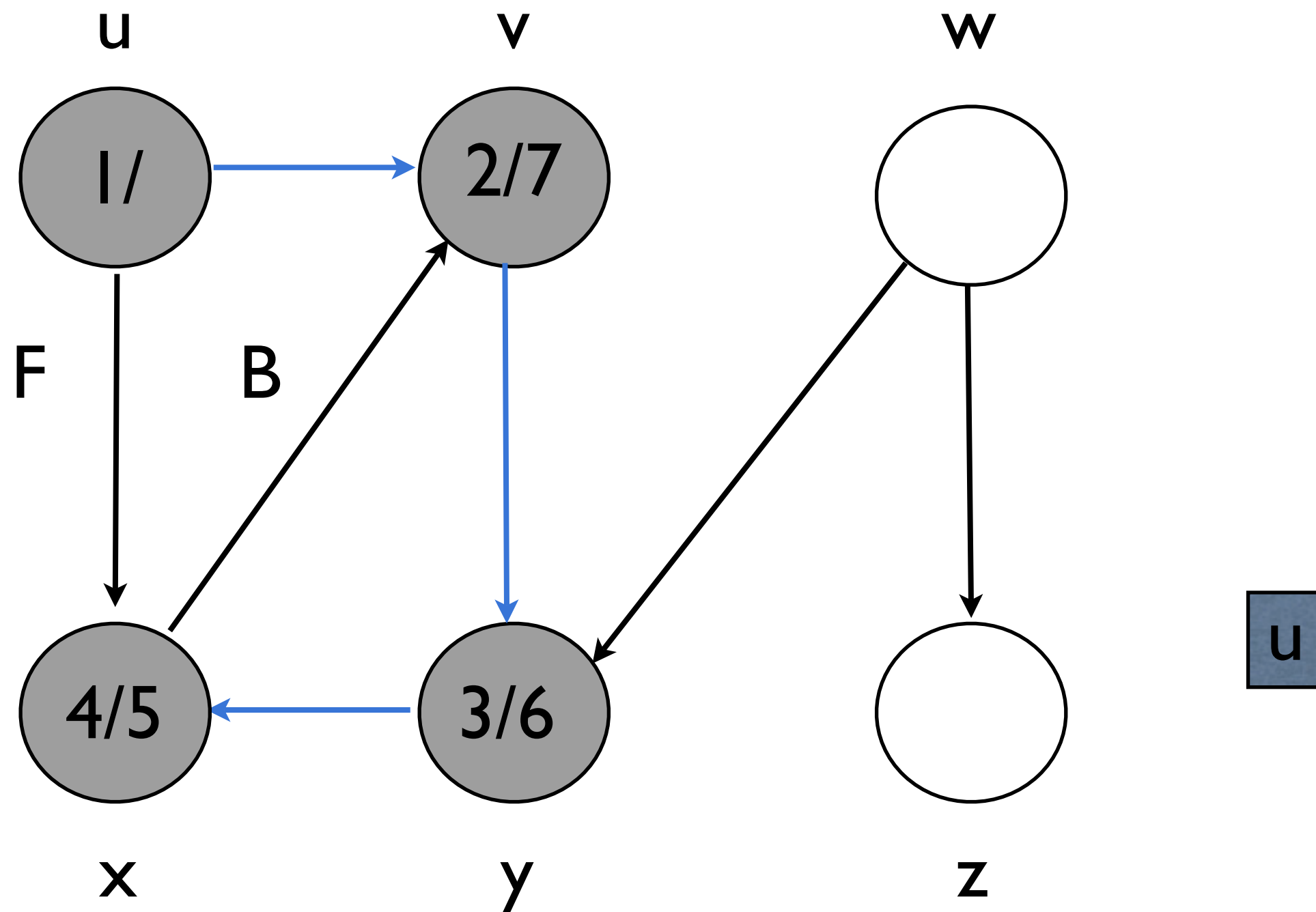


# Busca em Profundidade

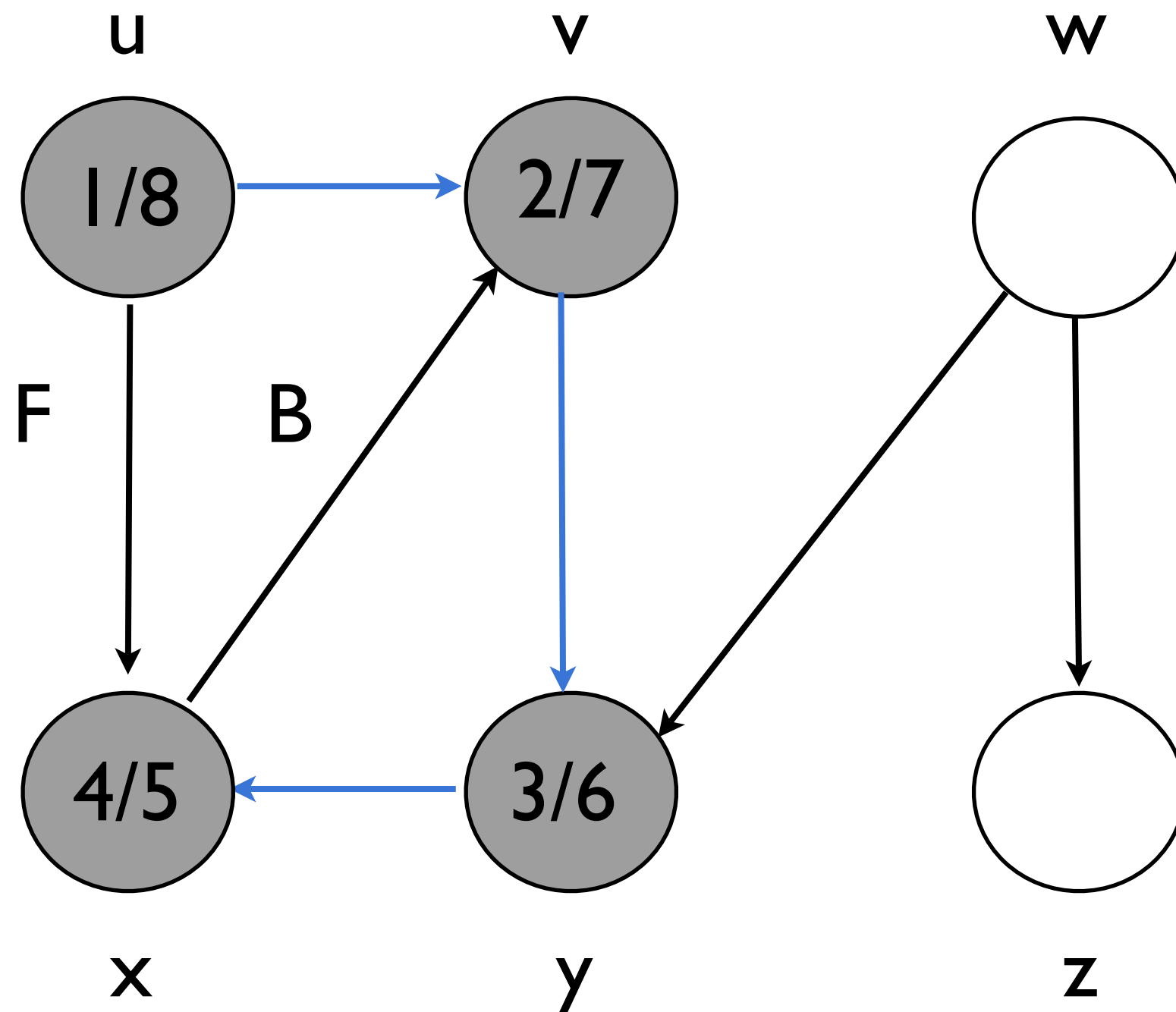




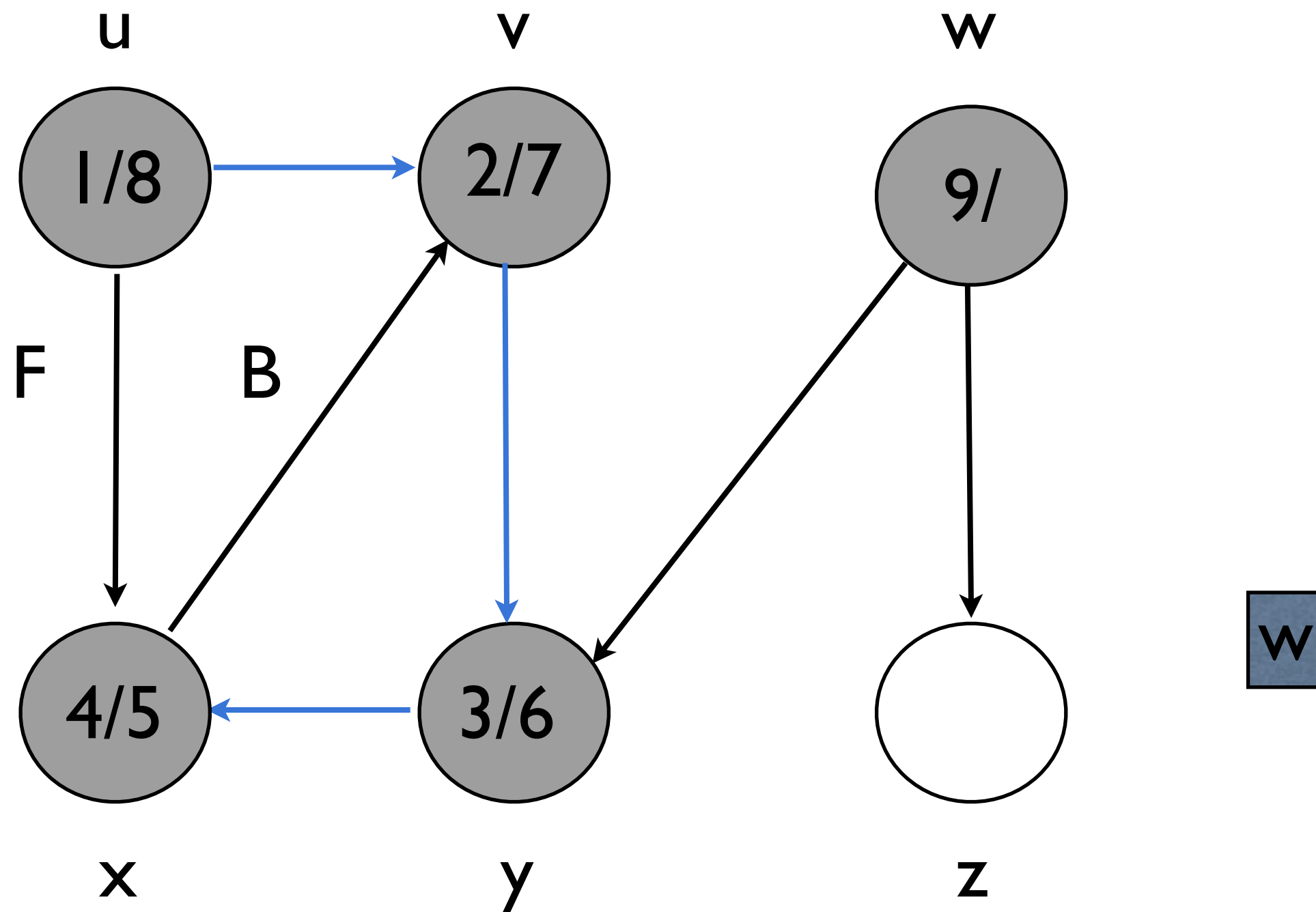
# Busca em Profundidade



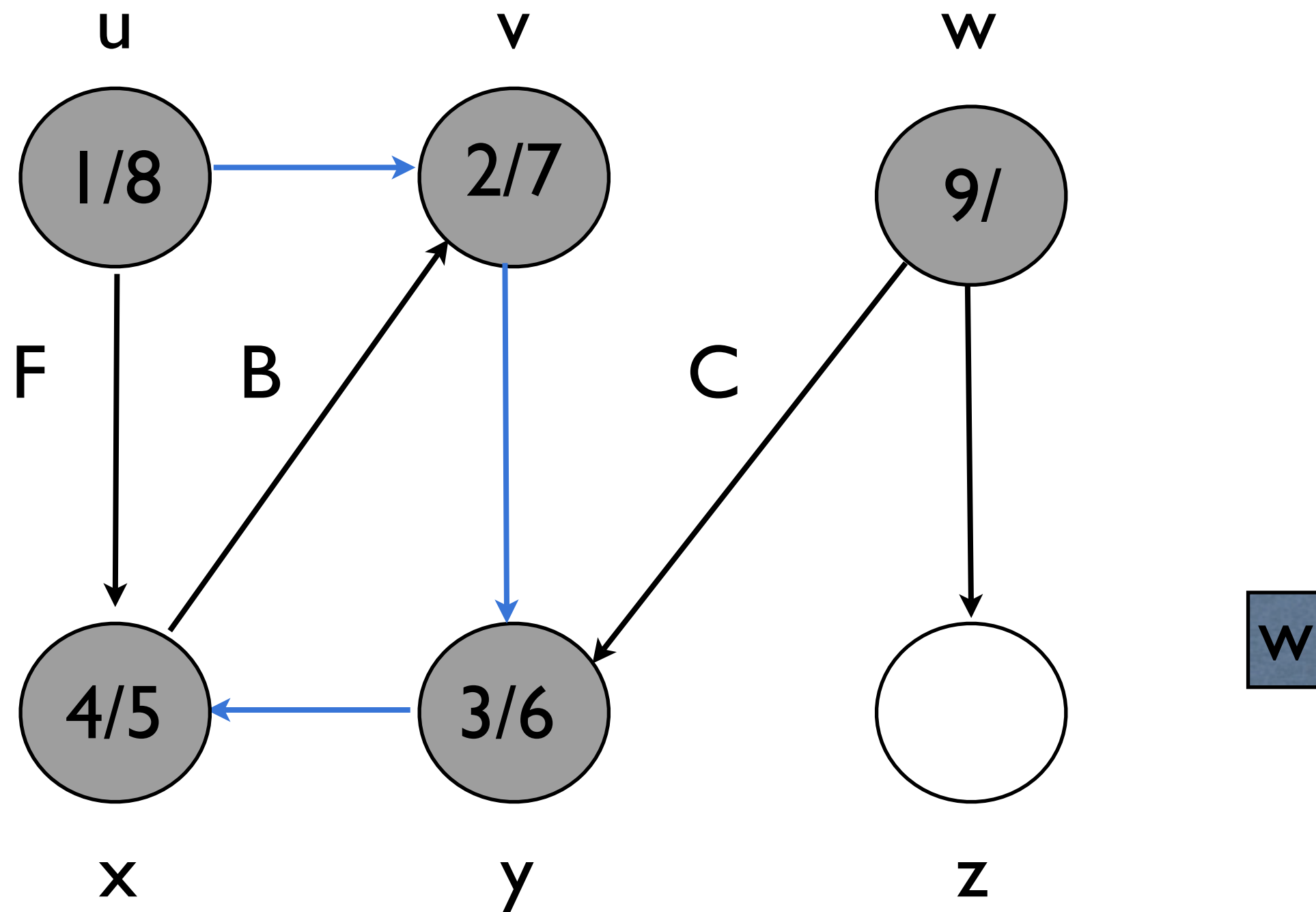
# Busca em Profundidade



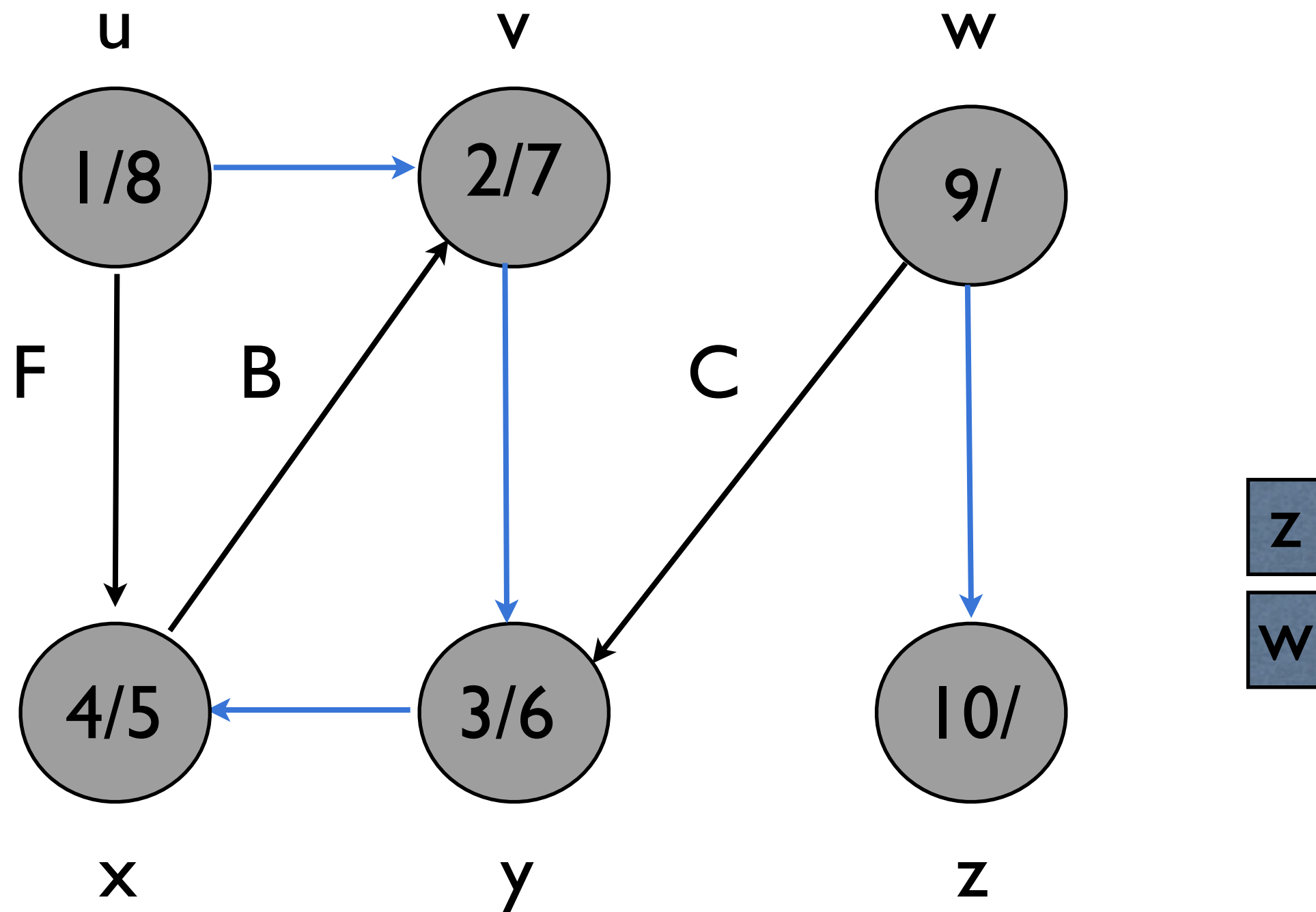
# Busca em Profundidade



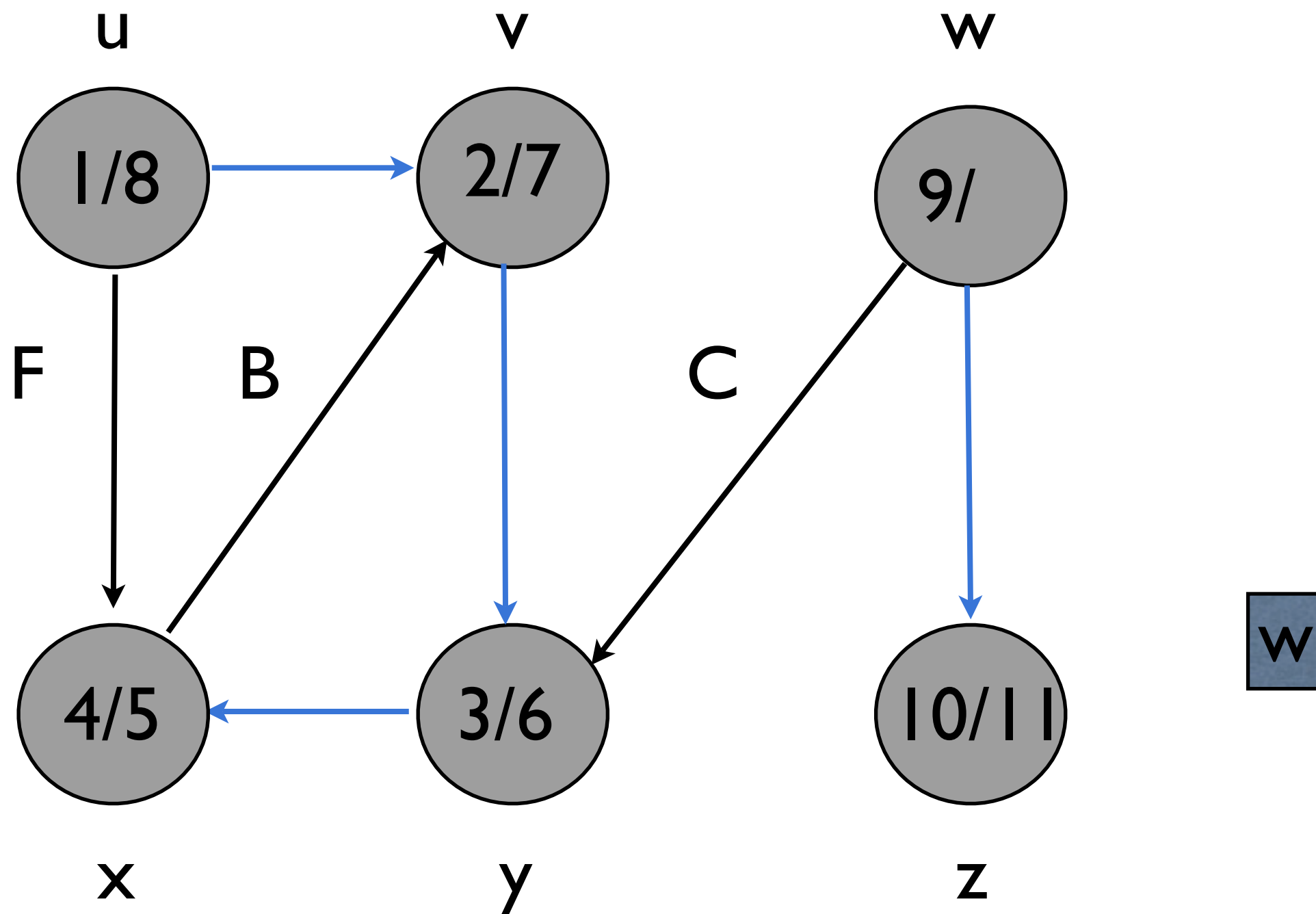
# Busca em Profundidade



# Busca em Profundidade

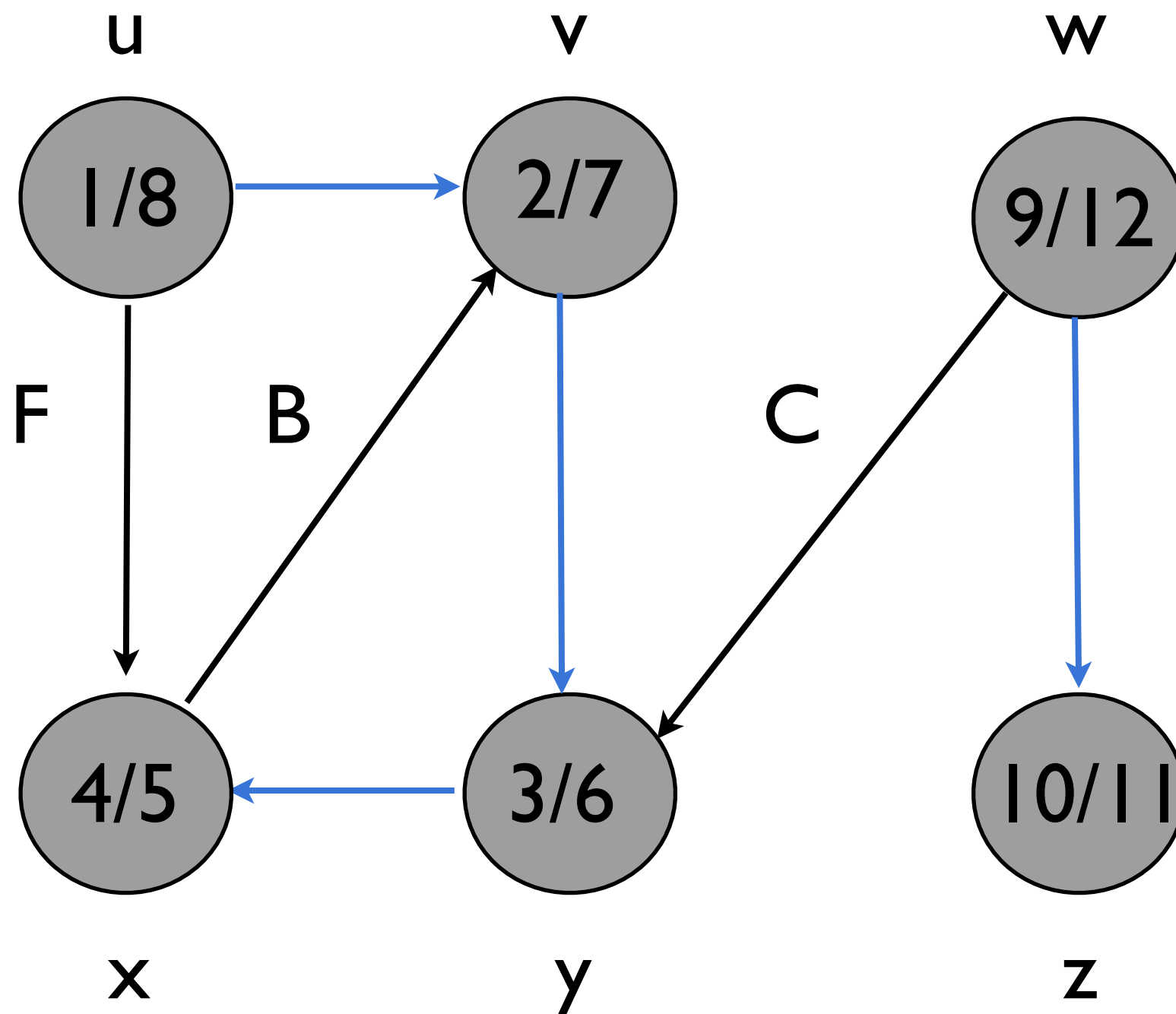


# Busca em Profundidade



# Busca em Profundidade

- Floresta



# Busca em Profundidade

- Floresta
  - $G_{\text{pred}} = (V, E_{\text{pred}}),$
  - $E_{\text{pred}} = \{(pred[v], v) \mid v \in V \mid pred[v] \neq nil\}$



# Busca em Profundidade

- Arestas
  - Forward (F)
    - Arestas que conectam  $v$  a um descendente da árvore
  - Back (B)
    - Arestas que conectam  $v$  a um ancestral da árvore

# Busca em Profundidade

- Arestas
  - Tree (T)
    - Arestas que pertencem a floresta
  - Cross (C)
    - Demais arestas

# Busca em Profundidade

- Aplicação
  - Um grafo sem ciclos é aquele que não possui arestas do tipo B

# Busca em Profundidade

- Aplicações
  - Estrutura de parênteses
    - Tempo de descoberta e tempo final
  - Ordenação topológica
  - Detecção de ciclos
  - Busca

# Busca em Profundidade

`busca_profundidade(G)`

**Para cada**  $u \in G.v$  **faça**

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

$\text{pred}[u] = \text{nil}$

$\text{tempo} = 0$

**Para cada**  $u \in G.v$  **faça**

**Se**  $\text{cor}[u] = \text{branco}$  **então**

$\text{visita}(G, u)$

# Busca em Profundidade

visita(G,u)

tempo = tempo + 1

d[u] = tempo

cor[u] = cinza

**Para cada** v e G.Adj[v] **faça**

**Se** cor[v] = branco **então**

pred[v] = u

visita(G,v)

cor[u] = preto

tempo = tempo + 1

f[u] = tempo

# Análise de Desempenho

- Inicialização
  - Marcar os vértices  $O(|V|)$
- Vizinhos
  - Estrutura de Adjacência  $O(|E|)$
  - Matriz de Adjacência  $O(|V|^2)$

# Análise de Desempenho

- Estrutura de Adjacência  $O(|V|+|E|)$
- Matriz de Adjacência  $O(|V|^2)$