

Grafos

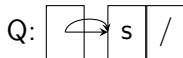
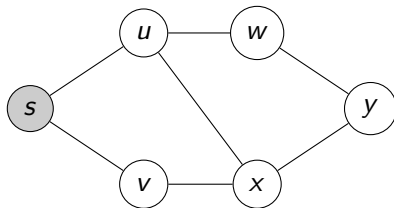
Ricardo Dutra da Silva

Universidade Tecnológica Federal do Paraná

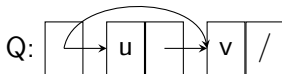
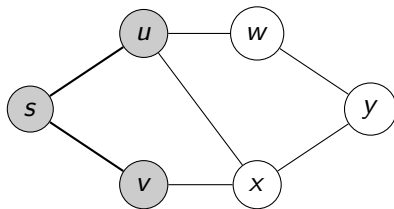
Algoritmo: BuscaLargura($G = (V, E)$, s)

```
1  para  $v \in V$  faça
2     $v.visitado \leftarrow \text{falso}$ 
3   $s.visitado \leftarrow \text{verdadeiro}$ 
   /* Fila auxiliar Q. */
4  Insere( $Q$ ,  $s$ )
5  enquanto  $Q \neq \emptyset$  faça
6     $v \leftarrow \text{Remove}(Q)$ 
7    para  $(v, w) \in E$  faça
8      se  $w.visitado = \text{falso}$  então
9         $w.visitado \leftarrow \text{verdadeiro}$ 
10       Insere( $Q$ ,  $w$ )
```

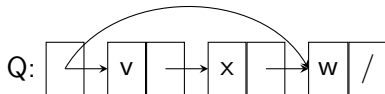
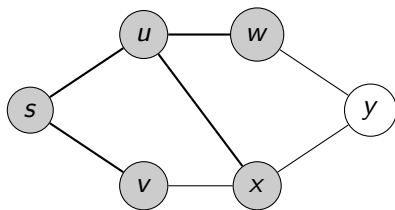
Busca em Largura



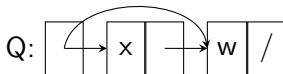
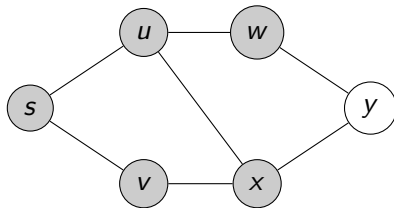
Busca em Largura



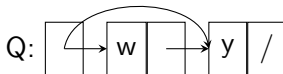
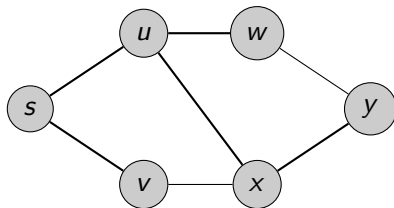
Busca em Largura



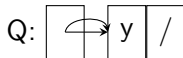
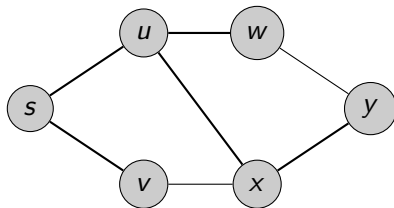
Busca em Largura



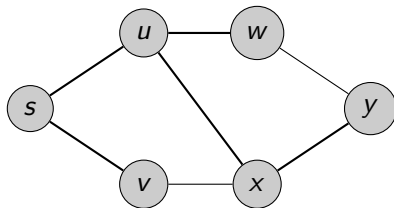
Busca em Largura



Busca em Largura



Busca em Largura



Q: /

Análise da complexidade do algoritmo (supondo um grafo conexo):

- linhas 1 e 2 executadas $\mathcal{O}(n)$ vezes;
- linhas 3 e 4 executadas uma vez ($\mathcal{O}(1)$);
- linhas 5 e 6 executadas uma vez para cada vértice, $\mathcal{O}(n)$ vezes;
- linhas 7 e 8 executadas duas vezes para cada aresta, $\mathcal{O}(m)$ vezes;
- linhas 9 e 10 têm execução limitada pelas linhas 7 e 8, $\mathcal{O}(m)$ vezes.

Portanto, o algoritmo tem complexidade $\mathcal{O}(n + m)$.

