

Algoritmos - Aula 8

Fernando Raposo

Vamos ver

- Contextualização
 - Web Crawlers
 - Que tipo de busca fazem?
- Busca em Largura
 - Aplicações práticas

Web Crawler

- O que são Web Crawlers?
 - Você sabe quem são estes dois abaixo?



Web Crawler

- O que são Web Crawlers?
 - Uma dica...
 - Larry Page e Sergey Brin

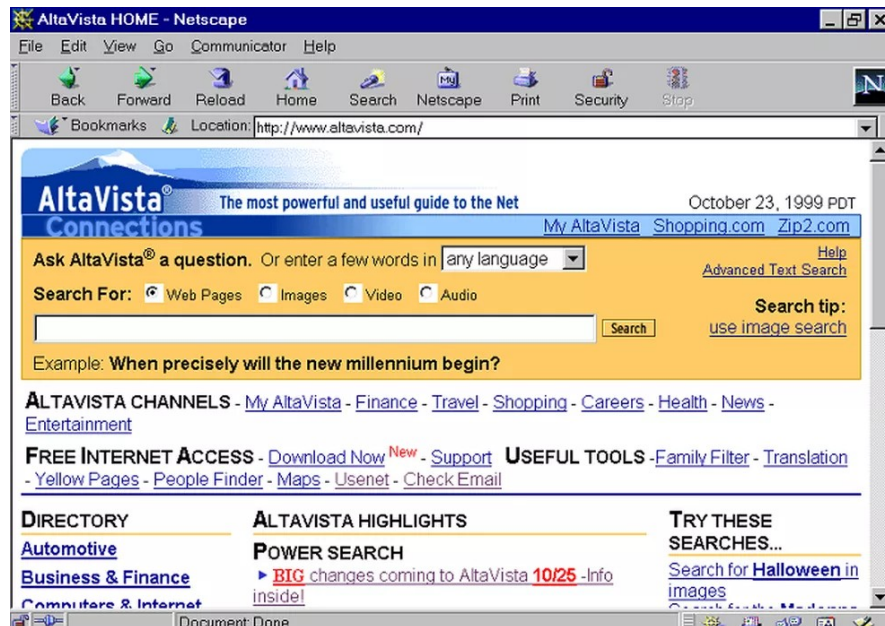


Web Crawler

- O **Google** começou na garagem de seus fundadores, quando ainda eram estudantes de Stanford.
- A missão do Google seria “*Organizar toda a informação do mundo e torná-la universalmente acessível*”.
- Web Crawler:
 - Mecanismo poderoso (robôs) de leitura de indexação de websites.
- O Google nada mais era em seu início que um índice de sites que buscava informação na Web de forma sem igual comparado aos competidores existentes.

Alguém lembra?

YAHOO!



Como Funcionam Web Crawlers?

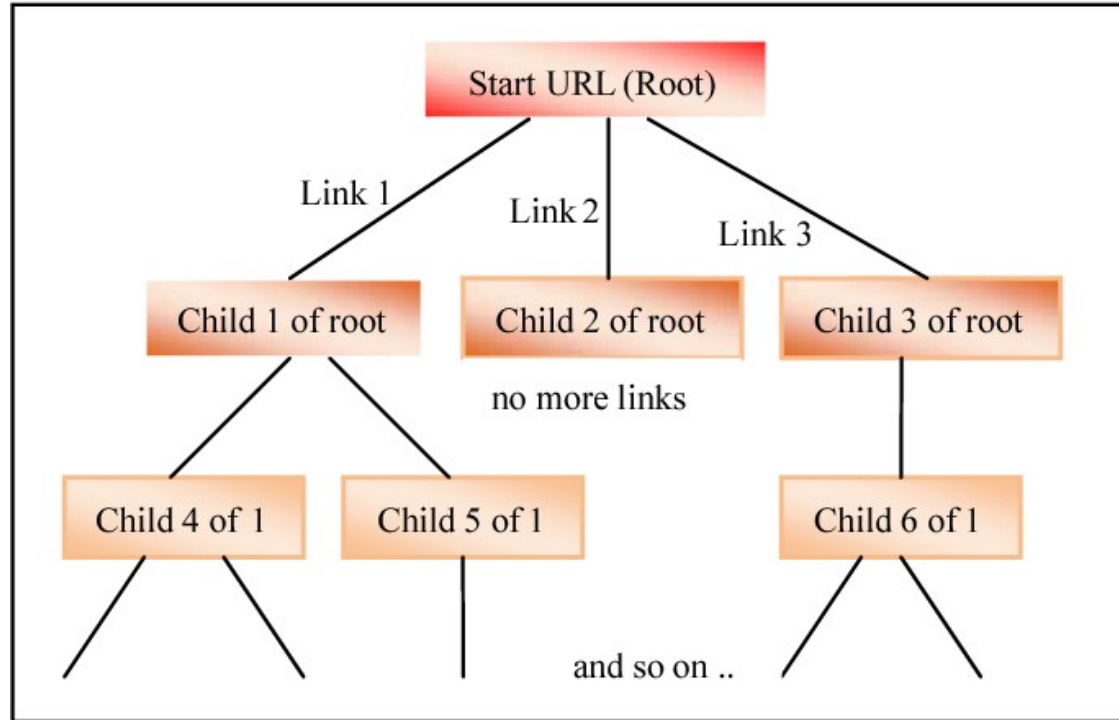


Figure 2. Web Site as a Tree-Structure

Como é feita a busca em um Web Crawler?

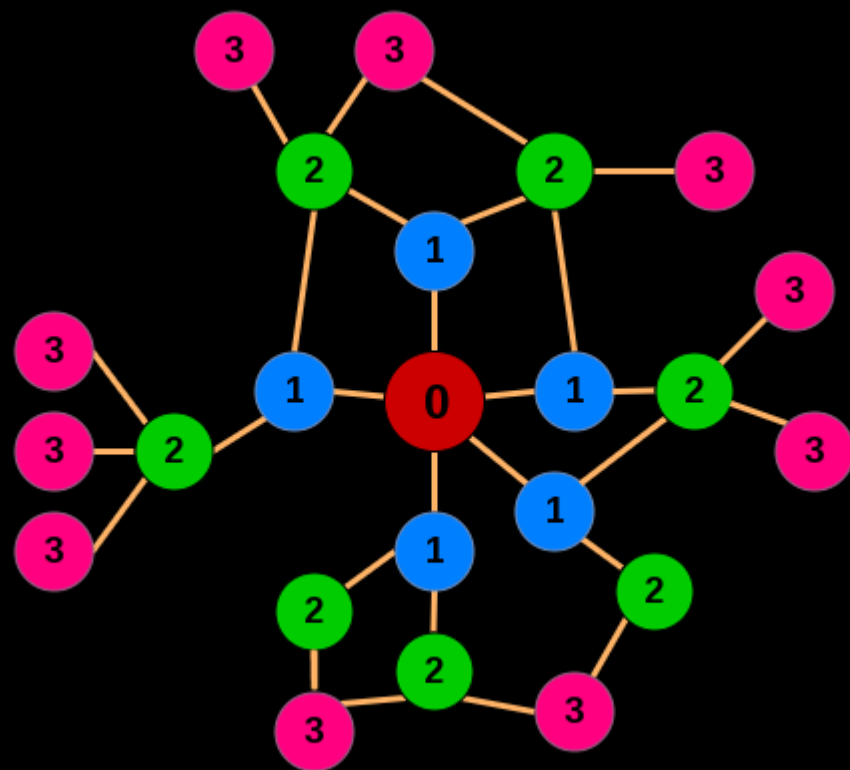
- Web Crawlers utilizam estratégias de busca que recuperam todos os sites filhos referenciados por um site pai e repetem a mesma busca para os sites filhos;
- O tipo de busca que é feita por estes é a chamada **Busca em Largura** (ou breadth-first search);
- As aplicações de busca em largura são inúmeras

Busca em Largura: Aplicações

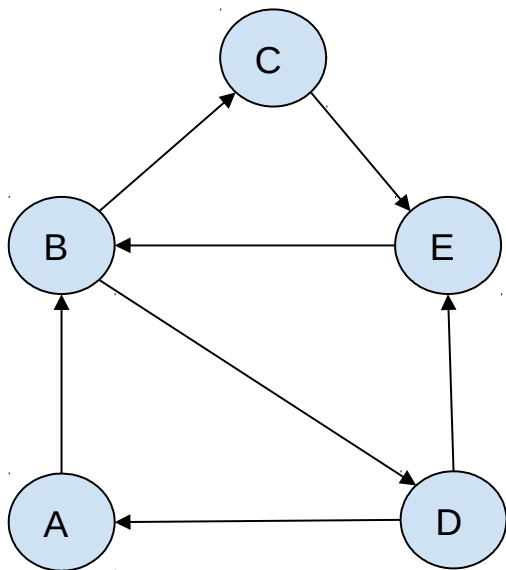
- Aplicativos de Peer-to-Peer (P2P)
 - [BitTorrent](#)
- Engenhos de Redes Sociais
 - Procurar amigos de amigos
- Broadcasting em Redes
- Achar caminho em um Grafo (ou o menor caminho)
- Sistemas de GPS
 - Para achar vizinhanças

Busca em Largura

- Estratégia:
 1. Inicie a Busca em um vértice x ;
 2. Visitar vértice x
 3. Visitar todos os vértices vizinhos de x ;
 4. Visitar todos os vértices vizinhos dos vizinhos de x e assim sucessivamente;



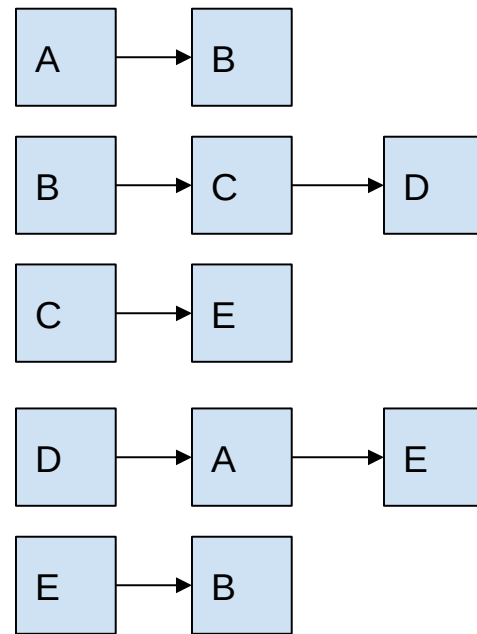
Busca em Largura



Grafo



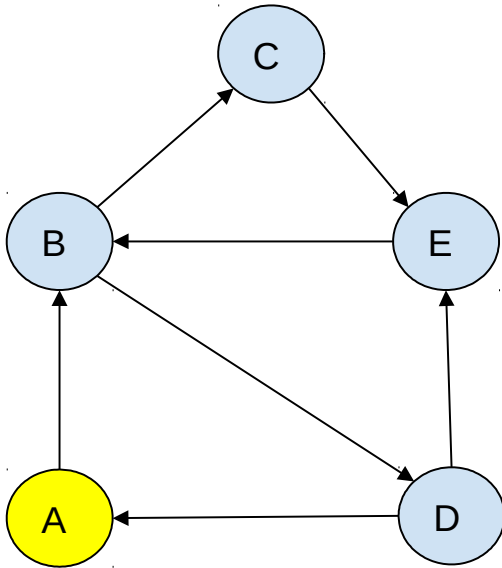
Fila



Lista de Adjacências

Busca em Largura

- **Início:** Marcamos o primeiro vértice que escolhemos como visitado, podemos marcar a trilha percorrida também.

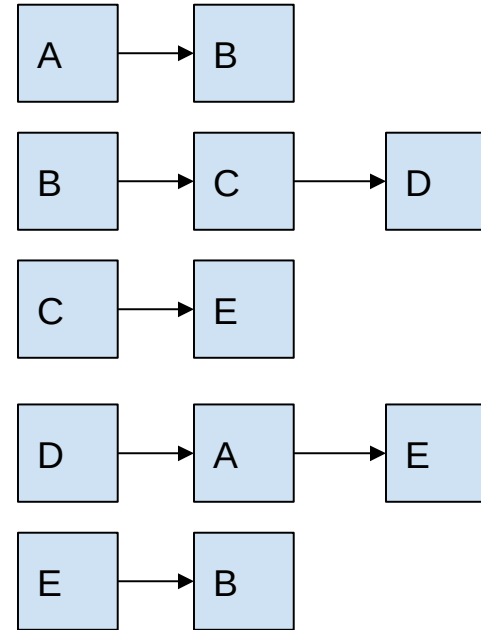


Grafo

Trilha:
A



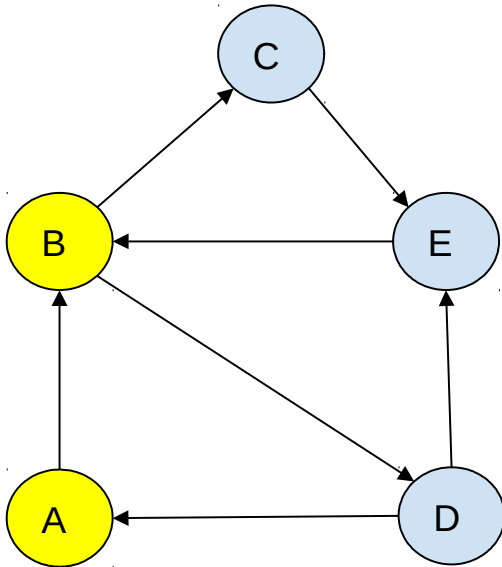
Fila



Lista de Adjacências

Busca em Largura

- Visitamos todos vizinhos de “A” que ainda não foram visitados, e os colocamos na Fila. Vamos aproveitar para registrar a trilha.

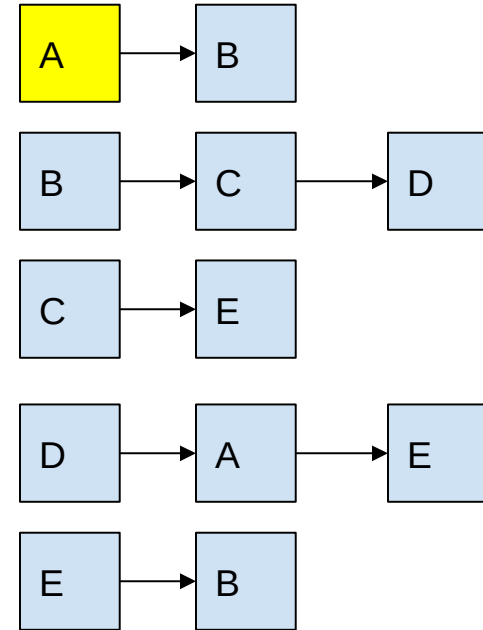


Grafo

Trilha:
A|B



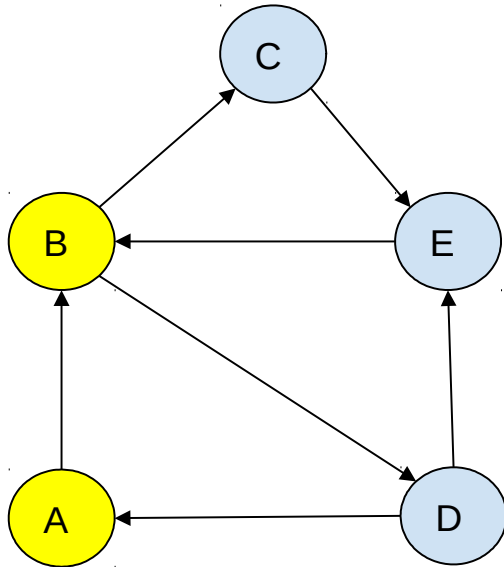
Fila



Lista de Adjacências

Busca em Largura

- ... “A” só tem “B” como vizinho, como terminei de visitar todos os vizinhos de “A”, volto-me para a minha fila e passo a trabalhar com o primeiro elemento

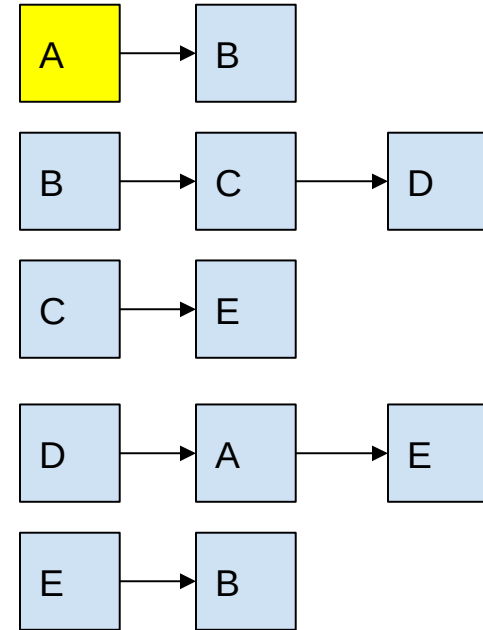


Grafo

Trilha:
A|B



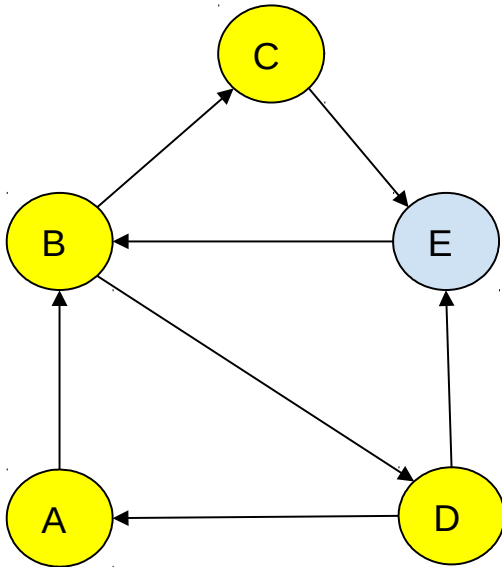
Fila



Lista de Adjacências

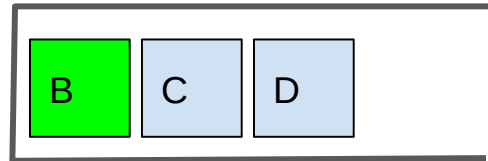
Busca em Largura

- Visitamos todos vizinhos de “B” não visitados, e os colocamos na Fila. Vamos aproveitar para registrar a trilha.

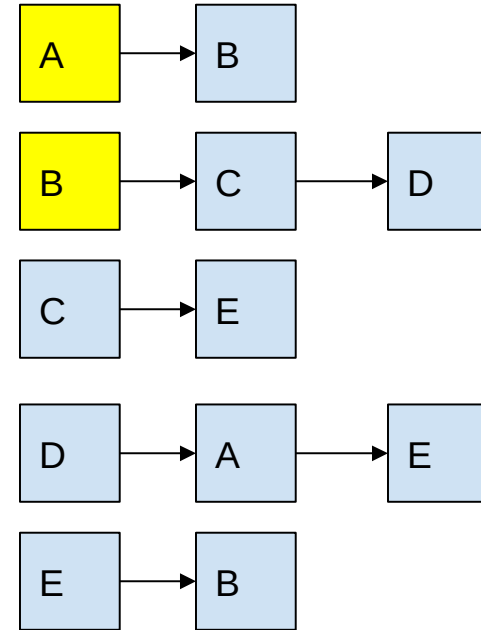


Grafo

Trilha:
A|B|C|D



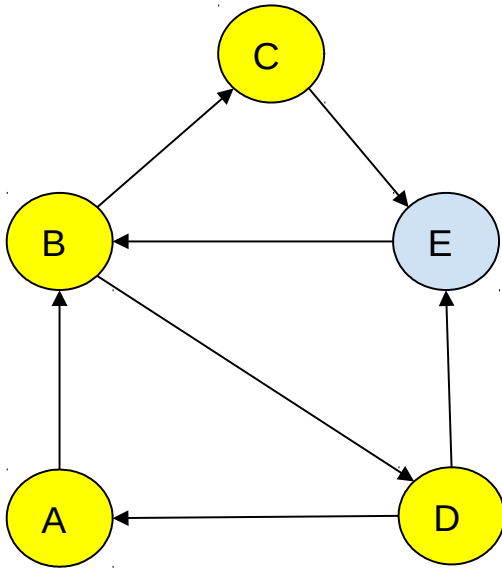
Fila



Lista de Adjacências

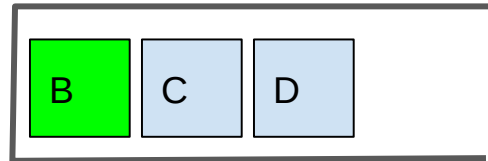
Busca em Largura

- Não há mais o que fazer com “B”, vamos removê-lo da fila e passar para o próximo elemento.

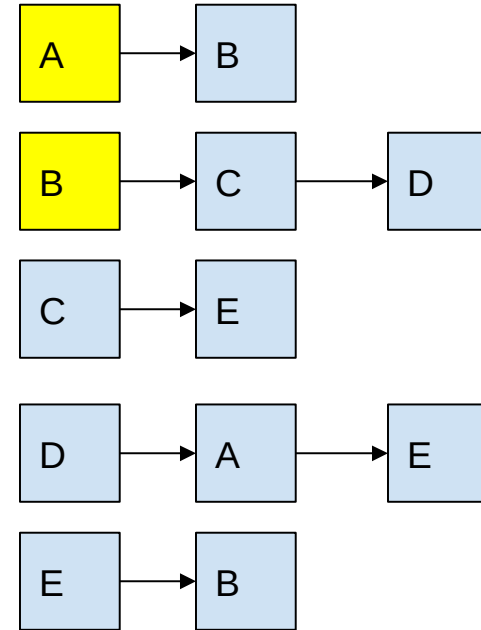


Grafo

Trilha:
A|B|C|D



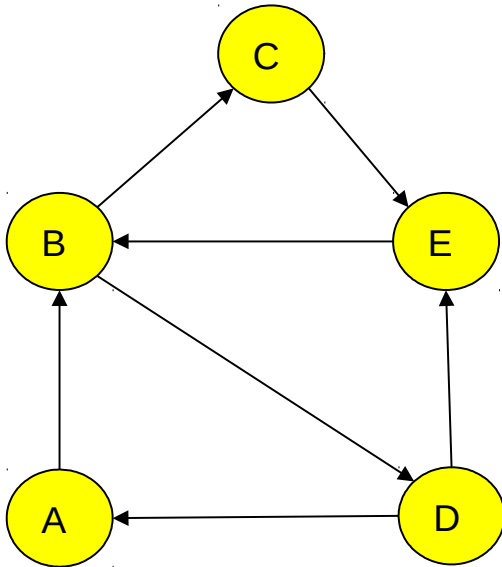
Fila



Lista de Adjacências

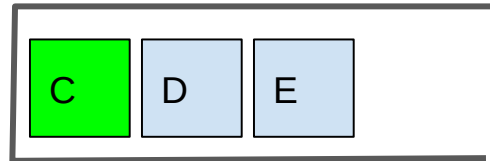
Busca em Largura

- Visitamos todos vizinhos de “C” não visitados, e os colocamos na Fila. Vamos aproveitar para registrar a trilha.

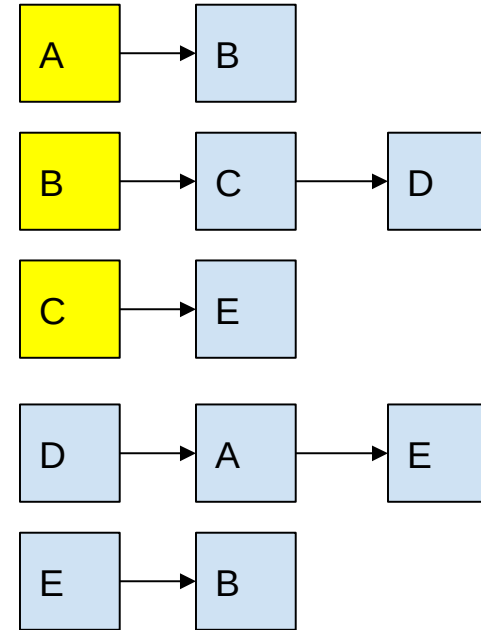


Grafo

Trilha:
A|B|C|D|E



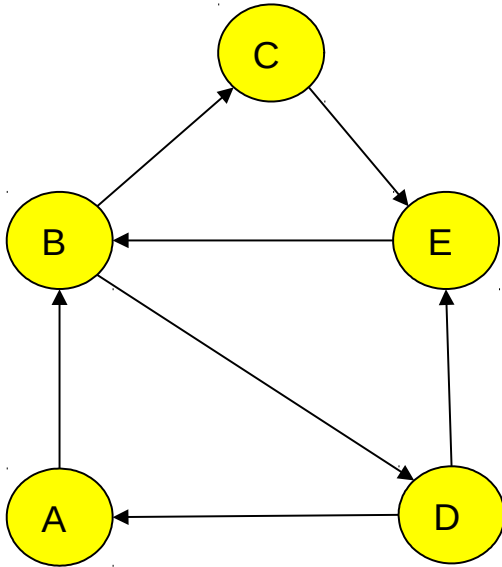
Fila



Lista de Adjacências

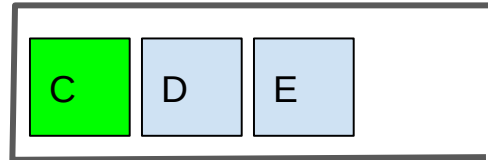
Busca em Largura

- Não há mais o que fazer com “C”, vamos removê-lo da fila e passar para o próximo elemento.

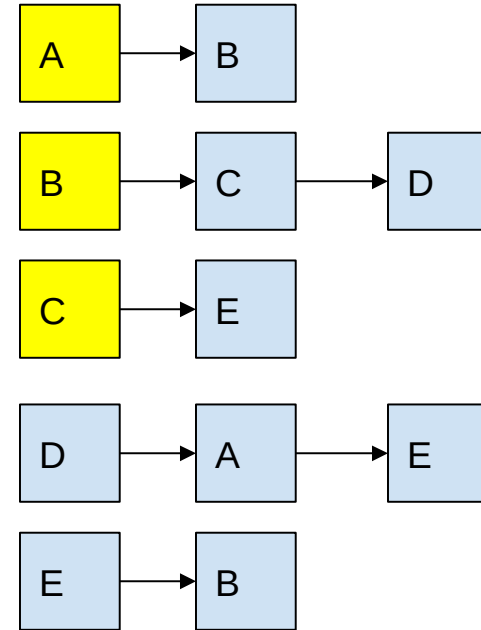


Grafo

Trilha:
A|B|C|D|E



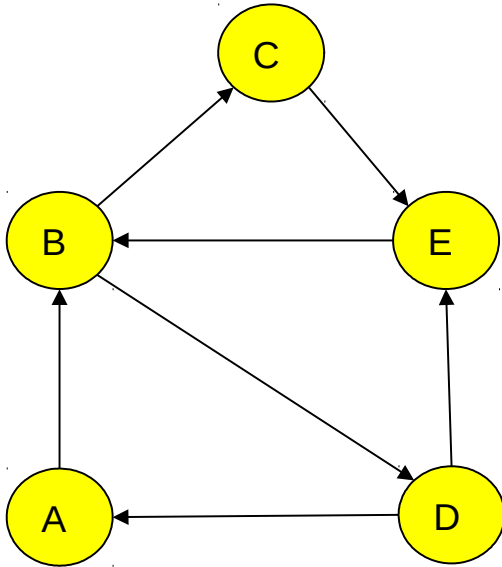
Fila



Lista de Adjacências

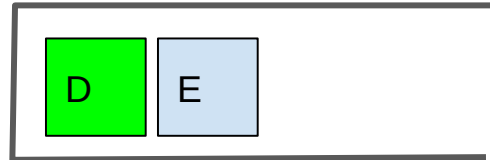
Busca em Largura

- Todos os vizinhos de “D” já foram visitados. Não há mais o que fazer além de retirá-lo da fila

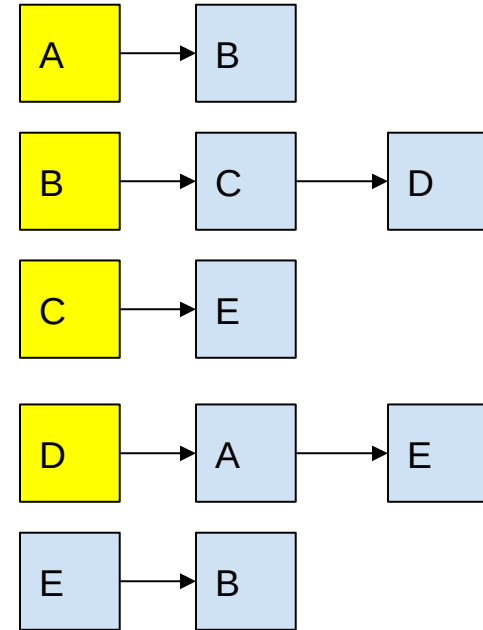


Grafo

Trilha:
A|B|C|D|E



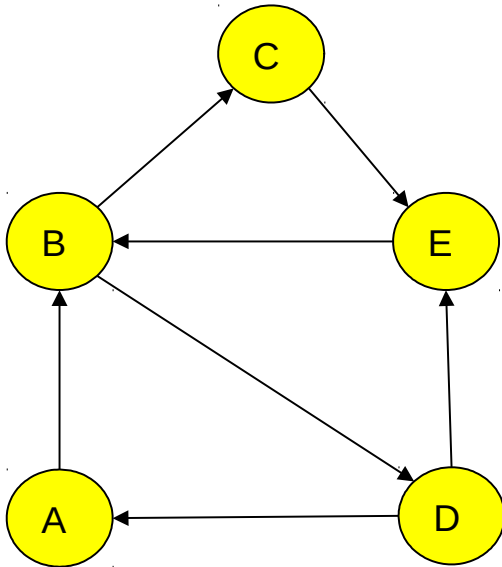
Fila



Lista de Adjacências

Busca em Largura

- Todos os vizinhos de “E” já foram visitados. Não há mais o que fazer além de retirá-lo da fila

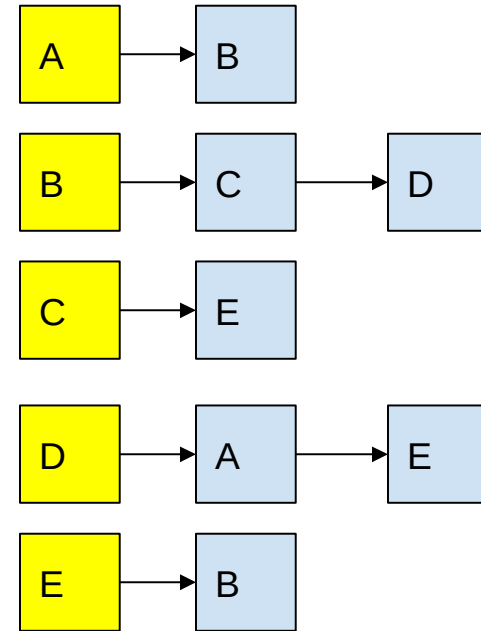


Grafo

Trilha:
A|B|C|D|E



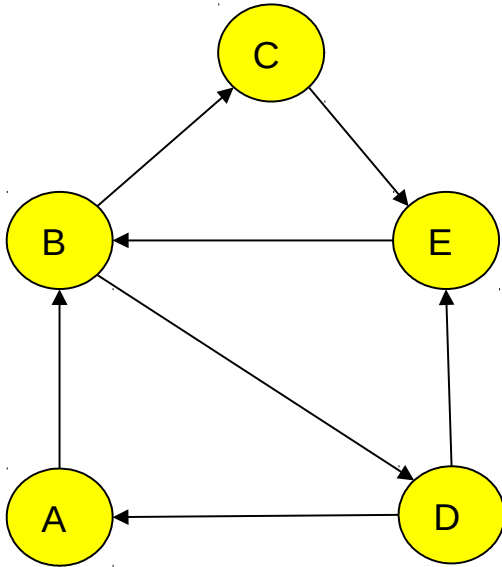
Fila



Lista de Adjacências

Busca em Largura

- Fila vazia e Grafo percorrido em Busca em Largura!

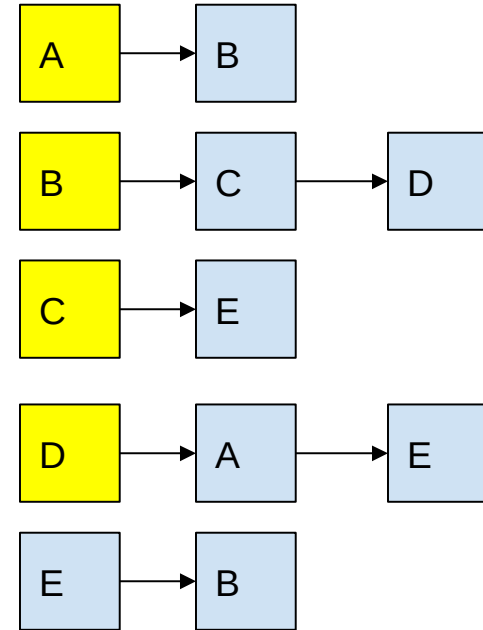


Grafo

Trilha:
A|B|C|D|E



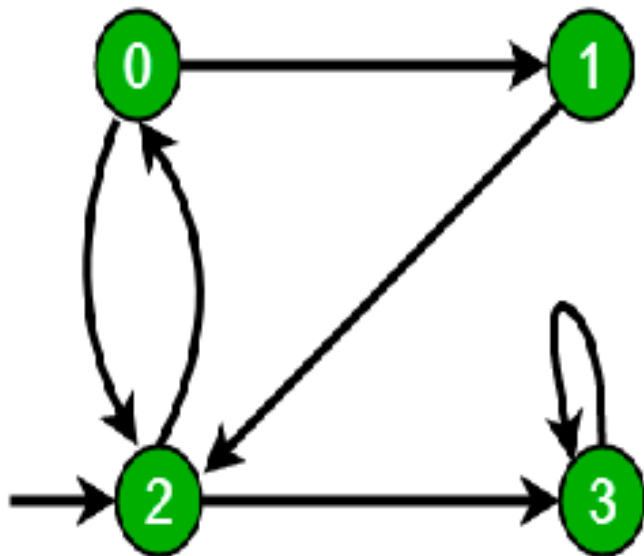
Fila



Lista de Adjacências

Parte Prática

- [Já implementamos o grafo anteriormente...](#)
- Vamos implementar a Busca em Largura do Grafo abaixo?
- **Obs:** Precisamos de um array para marcar os vértices visitados.



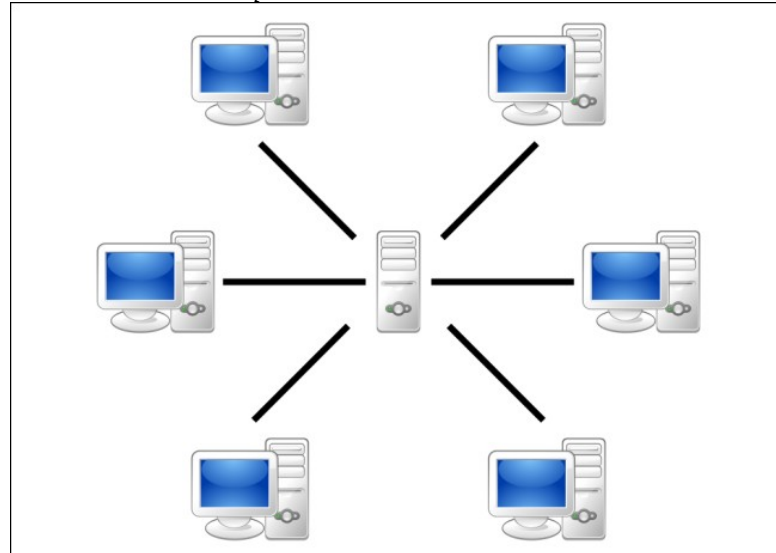
```
adicionaAresta( 0, 1);  
adicionaAresta( 0, 2);  
adicionaAresta( 1, 2);  
adicionaAresta( 2, 0);  
adicionaAresta( 2, 3);  
adicionaAresta( 3, 3);
```

Qual o caminho percorrido?

2 => 0 => 3 => 1

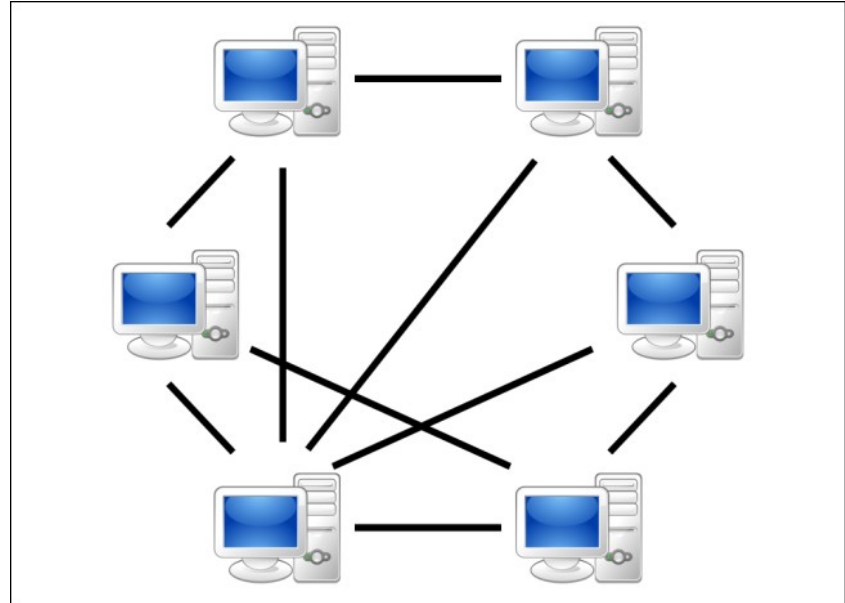
BitTorrent

- O BitTorrent não significa pirataria!
 - A Blizzard Games já utilizou o BitTorrent por exemplo para envio de atualizações de World of Warcraft e Diablo 3.
 - Isso facilitava a distribuição do jogo, pois evitava que todos os jogadores do mundo não fossem ao mesmo tempo tentar baixar a atualização.
- Peer-2-Peer
 - Início com o Napster;
 - Servidor centralizado;
 - Informava quem tinha o quê;
 - Fácil de desligar, basta parar o servidor central.



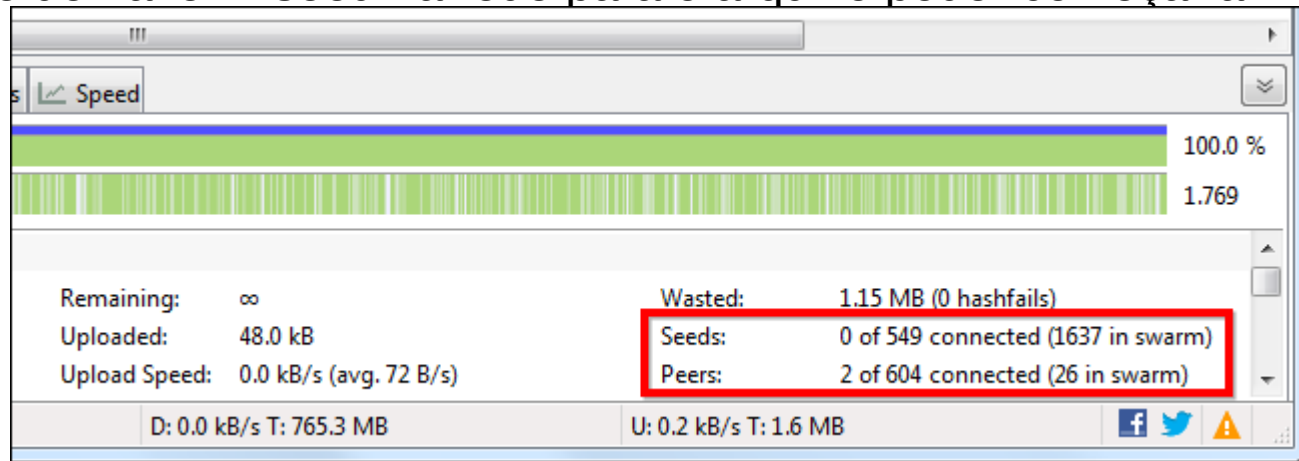
BitTorrent

- Com o passar do tempo, uma nova abordagem apareceu. Os servidores passaram a ficar distribuídos, tornando impossível “desligar” uma rede de P2P bem feita;
- Seed: Alguém na rede de Torrent que tem um arquivo completo;
- Peer: Alguém que está na rede e está baixando e enviando o arquivo;



BitTorrent

- Há a necessidade de haver 1 seed na rede para o arquivo poder começar a ficar distribuído.
- E se o seu peer cai?



- Fácil: Utilizar Busca em Largura para encontrar o peer mais próximo que tenha o arquivo.

