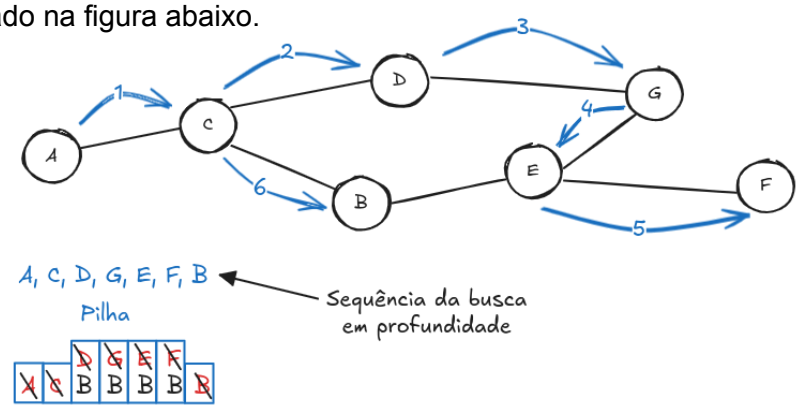
CENTRO UNIVERSITÁRIO DE PATOS DE MINAS – UNIPAM
BACHARELADO EM SISTEMAS DE INFORMAÇÃO
TURMA: 4º PERÍODO – 02/2024
ESTRUTURA DE DADOS E ALGORITMOS
PROFESSOR RAFAEL MARINHO E SILVA

Algoritmos de Busca

A operação de busca consiste em explorar o grafo de uma maneira bem específica. Trata-se de um processo sistemático de como caminhar por seus vértices e arestas, geralmente a partir de um vértice inicial específico. Esse ponto de partida é crucial, especialmente em buscas pelo menor caminho. Dependendo do problema, a busca pode precisar visitar todos ou apenas alguns vértices. Existem vários tipos de busca que podemos realizar em um grafo. Será apresentado neste material os três principais: busca em profundidade, busca em largura e busca pelo menor caminho.

Busca em Profundidade

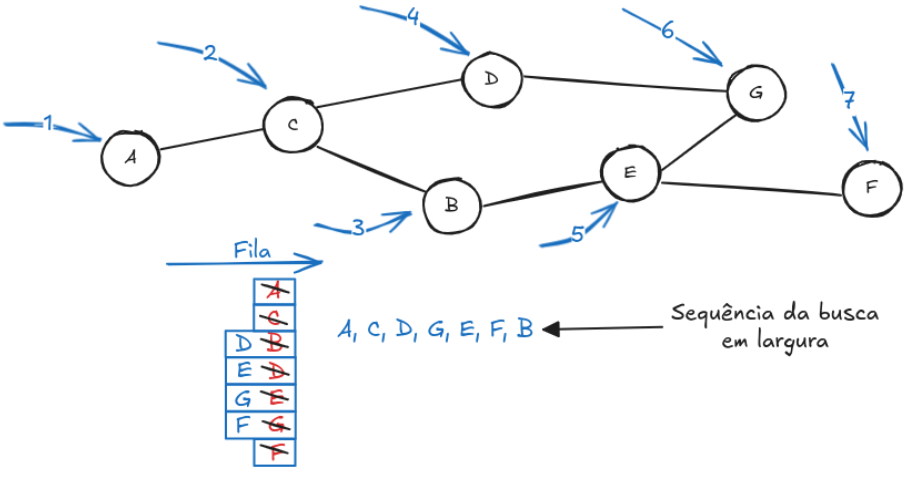
Explora o grafo avançando o máximo possível por um caminho antes de retroceder (backtracking) e explorar caminhos alternativos. Essa abordagem usa uma pilha (ou recursão) para acompanhar o caminho percorrido e pode ser útil para verificar a conectividade do grafo, detectar ciclos e identificar componentes conectados, como ilustrado na figura abaixo.



A busca em profundidade é também uma técnica eficiente para aplicações como a ordenação topológica em grafos direcionados acíclicos, identificação de pontes e pontos de articulação em grafos, e a resolução de labirintos. No entanto, ela pode ser menos eficiente que a busca em largura em grafos densos para encontrar o caminho mais curto, pois ela pode explorar caminhos longos antes de alcançar a solução. Além disso, em grafos muito grandes, a busca em profundidade pode consumir muita memória se a profundidade da recursão for elevada.

Busca em Largura

Explora o grafo por camadas, visitando todos os vizinhos de um vértice antes de seguir para o próximo nível. Utiliza uma fila para garantir essa exploração em níveis e é eficiente para encontrar o caminho com o menor número de arestas entre dois vértices em grafos não ponderados, como ilustrado na figura abaixo.



A busca em largura é amplamente utilizada em problemas que requerem a menor distância entre dois pontos, como em algoritmos de roteamento em redes e na solução de quebra-cabeças que envolvem uma série de movimentos mínimos. Ela também é fundamental para verificar a bipartição de um grafo e detectar componentes conectados em grafos não direcionados. Apesar de suas vantagens, a ela pode consumir muita memória em grafos com alta largura (muitos vértices por nível), o que pode levar a problemas de desempenho em ambientes com memória limitada. Em grafos ponderados, a BFS não garante a solução ótima para encontrar o menor caminho em termos de custo, sendo mais adequada para grafos onde todas as arestas têm peso igual.

Busca pelo Menor Caminho

Visa encontrar o caminho de menor custo entre dois vértices, considerando os pesos nas arestas. Algoritmos como Dijkstra e Bellman-Ford são amplamente utilizados para esse tipo de busca em grafos ponderados. Além de serem aplicados em redes de transporte e roteamento de redes de computadores, esses algoritmos são essenciais em sistemas de navegação e otimização de recursos, como na análise de rotas logísticas e planejamento financeiro.

Embora eficientes, esses algoritmos possuem limitações: Dijkstra, por exemplo, não funciona corretamente em grafos com arestas de pesos negativos, e em grafos dinâmicos, onde pesos e conexões mudam constantemente, ambos podem se tornar ineficazes sem adaptações específicas, como o uso de algoritmos incrementais.