

Lista 2 - Resolucao

▼ 1 -

- Dijkstra: Tempo $O(V^2)$ e Espaço $O(V)$
- Bellman-ford: Tempo $O(V^3)$ e Espaço $O(V)$
- Floyd-Warshall: Tempo $O(V^3)$ e Espaço $O(V^2)$

▼ 2 -

- Iniciando pela vantagem do DFS em espaço, o IDS irá manter uma lista de prioridade com no pior dos casos $O(h \times n)$ onde h é a altura da árvore naquele nível e n é o número de filhos médios de cada nó. Por outro lado, irá, assim como no BFS, encontrar a solução mais próxima da raiz, sem analisar profundidades desnecessárias da árvore

▼ 3 -

- Seleção da Fronteira:
 - D, E, A, B, C, E
- Caminho Encontrado:
 - F, G, G, H, H, H
- Custo em Espaço:
 - J, I, J, J, J, I

▼ 4 -

- Para de uma forma tentar prever a distância de um nó até o nó objetivo para auxiliar nas podas, seleção de próximo nó em pesquisas de menor caminho.

▼ 5 -

- O algoritmo de poda de ciclos funciona por verificar se o nó que será visitado já não faz parte do caminho atual. Sendo assim, sua complexidade dependerá do método de pesquisa utilizado. Para o DFS, por exemplo, podemos armazenar em um 'set' os nós do caminho e assim a pesquisa se torna $O(1)$. Para as pesquisas de caminho que possuem múltiplos caminhos (as que possuem

custo de memória exponencial, como BFS, A*, Lowest-cost-first), o custo se torna linear, visto que ele devera pesquisar no caminho parcial atual se o nó faz ou não parte do caminho.

▼ 6 -

- A poda de múltiplos caminhos é feita de forma a manter um conjunto de nós explorados que estão ao final de um caminho previamente expandido. Ao selecionarmos um caminho na fronteira, verificamos se o nó final faz parte do conjunto, caso não faça, adicionamos ele ao conjunto, caso faça, descartamos o caminho.
- Observe que, esse método não garante o descarte do caminho da melhor solução, sendo assim, devemos tomar algumas medidas para evitar a situação mencionada.
 - Podemos ter a certeza que o primeiro caminho encontrado para um nó é o caminho com menor custo até o nó (assim como no Lower-cost-first) e podemos os restantes
 - Caso encontremos um caminho menor que um caminho que está no conjunto, podemos remover da fronteira os caminhos até um nó qualquer que utilizem esse caminho pior. Por exemplo, supondo que temos um caminho p na fronteira $\langle s, \dots, n, \dots, m \rangle$ e encontramos um p' até n que tem custo menor que p , excluimos da fronteira os caminhos que usam p .
 - Supondo o caso acima, podemos substituir nos caminho a porção do caminho p de s a n por p' .