

Algoritmos De FLOYD-WARSHALL & BELLMAN-FORD

FLOYD-WARSHALL

- O algoritmo de Floyd-Warshall encontra o caminho mais curto entre todos os pares de nós em um grafo ponderado.
- Funciona com pesos negativos, mas não com ciclos negativos.
- Usa programação dinâmica, atualizando progressivamente as distâncias entre os pares de nós.

Passo a passo do algoritmo de FLOYD-WARSHALL

- Inicialize a matriz:
 - $\text{dist}[i][j]$ = peso da aresta de i para j se existir, senão infinito.
 - $\text{dist}[i][i] = 0$ para todos i.
- Atualize com cidades intermediárias:
 - Para cada par de cidades (i, j), verifica se passar por uma cidade intermediária k oferece um caminho mais curto.
 - Se $\text{dist}[i][k] + \text{dist}[k][j] < \text{dist}[i][j]$, então atualizamos $\text{dist}[i][j]$ para $\text{dist}[i][k] + \text{dist}[k][j]$.
- Verificação de ciclos negativos:
 - Se, após as atualizações, $\text{dist}[i][i] < 0$ para qualquer i, então há um ciclo negativo no grafo.
- Resultado
 - Após executar o algoritmo, a matriz dist contém as distâncias mais curtas entre todas as pares de cidades.
 - Se $\text{dist}[i][j]$ ainda for infinito, significa que não há caminho entre i e j.

BELLMAN-FORD

- O algoritmo de Bellman-Ford encontra o caminho mais curto de um único nó para todos os outros em um grafo ponderado.
- Funciona mesmo com pesos negativos.
- Pode detectar ciclos negativos no grafo.
- Atualiza as distâncias repetidamente, processando todas as arestas $|V| - 1$ vezes.
- Se após essas iterações ainda for possível melhorar uma distância, isso indica a presença de um ciclo negativo.

Passo a passo do algoritmo de BELLMAN-FORD

- Inicialize as distâncias:
 - $\text{dist}[\text{source}(\text{fonte})] = 0$
 - Para todos os outros nós, $\text{dist}[u] = \infty$
- Processamento das arestas:
 - O algoritmo processa todas as arestas do grafo repetidamente.
 - Repita esse processo $|V| - 1$ vezes, onde $|V|$ é o número de nós no grafo.
- Atualização de distâncias:
 - Em cada iteração, para cada aresta (u, v) com peso w, verifique se o caminho atual até v pode ser melhorado passando por u.
 - Se $\text{dist}[u] + w < \text{dist}[v]$, então atualize $\text{dist}[v]$ para $\text{dist}[u] + w$.
- Verificação de ciclos negativos:
 - Após $|V| - 1$ iterações, execute uma verificação adicional.
 - Se for possível melhorar alguma distância ($\text{dist}[u] + w < \text{dist}[v]$), isso indica a presença de um ciclo negativo no grafo.