

CAMINHO MÍNIMO

Árvores de caminho mínimo

DIJKSTRA

Usado para encontrar os caminhos de peso mínimo entre um vértice de origem e os demais.

• Gráficos direcionados e ponderados com arestas de peso não-negativo

• Complexidade $O(E + V \cdot \log(V))$ → Complexidade alcançada utilizando prioridade com heap de Fibonacci.

Existe um caminho de menor custo ('atalho') de V_0 para V_3 passando por V_2 .

A operação de atualizar a distância entre dois vértices, passando por outro(s) vértice(s) é chamada de relaxamento.

ÁRVORE BUSCA

Árvore de caminho mínimo resultante.

ALGORITMO

Fila de prioridade

ENQUANTO houver vértices não processados:

SELECIONE u : vértice não processado com menor distância à origem:

MARQUE u como processado

PARA CADA v , adjacente de u , ainda não processado

SE $\text{dist}[u] + \text{peso}(u-v) < \text{dist}[v]$ ENTÃO

$\text{dist}[v] = \text{dist}[u] + \text{peso}(u-v)$

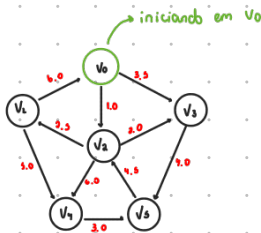
$\text{pred}[v] = u$

n = número de vértices

dist = array de distância de cada vértice ao vértice de origem, iniciando com 0 para a 'origem' e infinito para os demais

pred = array de predecessores

$\text{peso}(u-v)$, peso de u para v .



Arranjo de distâncias						
V_0	V_1	V_2	V_3	V_4	V_5	
0.0	-	-	-	-	-	
0.0	-	1.0	3.0	-	-	
0.0	3.5	1.0	3.0	7.0	-	
0.0	3.5	1.0	3.0	7.0	7.0	

Arranjo de predecessores						
V_0	V_1	V_2	V_3	V_4	V_5	
0	-1	-1	-1	-1	-1	
0	2	0	0	2	-1	
0	2	0	0	2	3	

ÁRVORE

