

Caminhos mínimos

Algoritmos e Estruturas de Dados - 2025/02

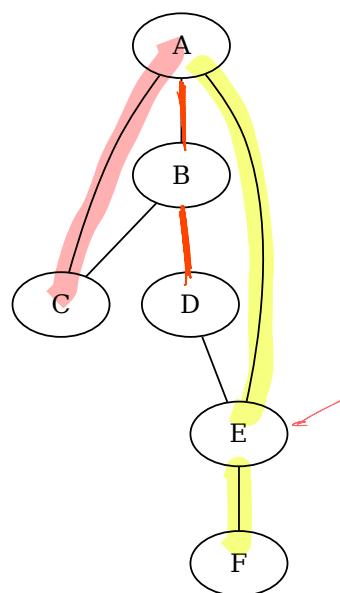
Última atualização: 25/07/2025 11:03

Nesta aula iremos simular a busca em largura (BFS) em dois grafos e compará-la com a busca em profundidade (DFS) para encontrar caminhos entre dois vértices.

Grafo 1

Queremos encontrar o menor caminho entre **A** e **F**

$$\begin{aligned} \text{Fila} &= [\cancel{A}, \cancel{B}, \cancel{C}, \cancel{D}, \cancel{E}, \cancel{F}] \\ \text{pred} &= [0, 0, 0, 1, 0, 3] \\ \text{dist} &= [0, 1, 1, 2, 1, 2] \end{aligned}$$



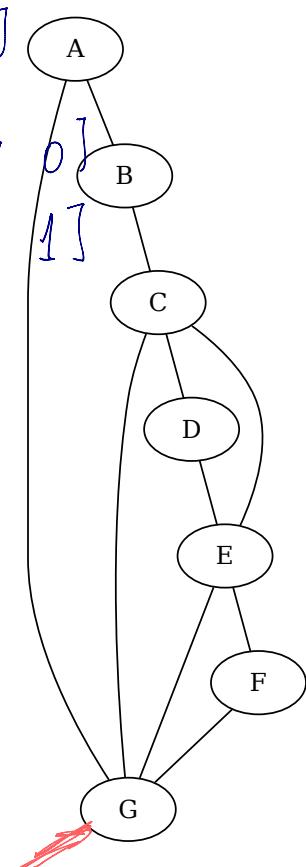
Grafo 2

Vamos encontrar o menor caminho entre **A** e **G**

Fila : [~~A~~ ~~B~~ ~~G~~ ~~C~~]

pred = [0 0 1 -1 -1 -1 0]

dist = [0 1 2 -1 -1 -1 1]



Algoritmo

Escreva abaixo o algoritmo da busca em largura.

 $\text{pred} = [-1, \dots, -1] | G.vi$ $\text{dist} = [-1, \dots, -1] | G.vi$ fila F (vazia) $\text{pred}[st] = st$ $\text{dist}[st] = 0$ add(F, st)while (vazia(F) = False) do atual = primeiro(F) if atual = dst thenA Fila circular FIM

Faremos essa parte juntas. Use o espaço abaixo para anotações.

 (G, st, dst) for cada vizinho v de atual do if $\text{dist}[v] = -1$ then add(F, v) $\text{dist}[v] = \text{dist}[\text{atual}] + 1$ $\text{pred}[v] = \text{atual}$

end

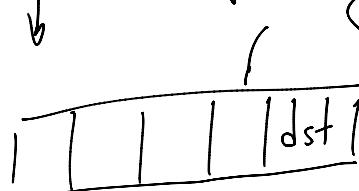
 if $\text{pred}[dst] = -1$ then

não tem caminho

else

 tem caminho \rightarrow usar pred

end

 $\text{pred}(dst)$ $\text{dist}[dst] + 1 = N$  $\text{pred}[\text{pred}[dst]]$