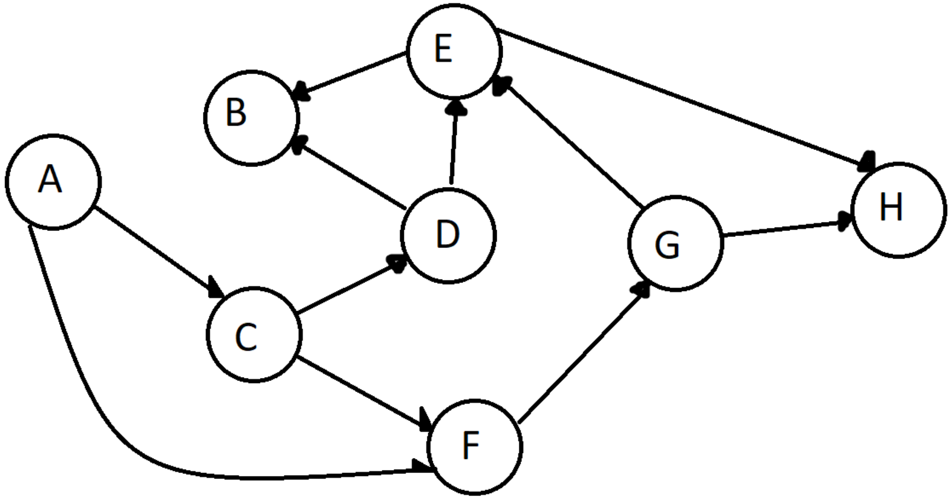
BFS Shortest Path

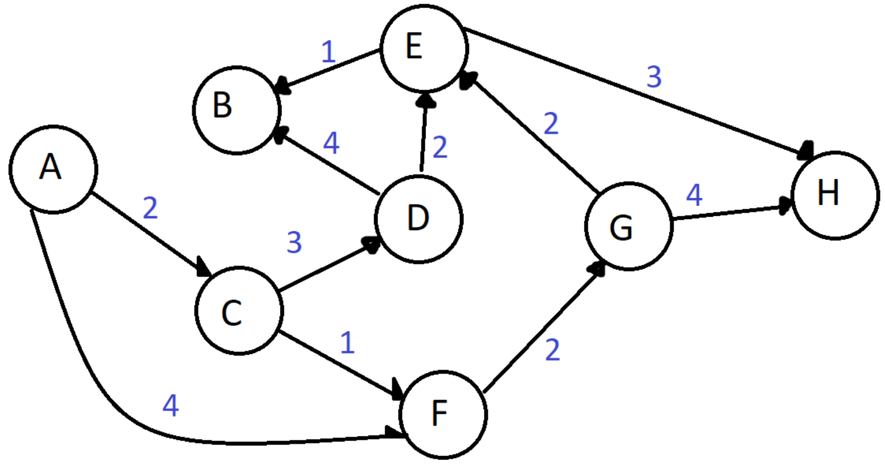


Time : $O(V+E)$

Usecase : No weights, single-sourced

Methodology : Exact same as BFS

Dijkstra Shortest Path



Time : $O(E\log V)$

Usecase : Non-negative weights , single-sourced

Methodology : Use BFS

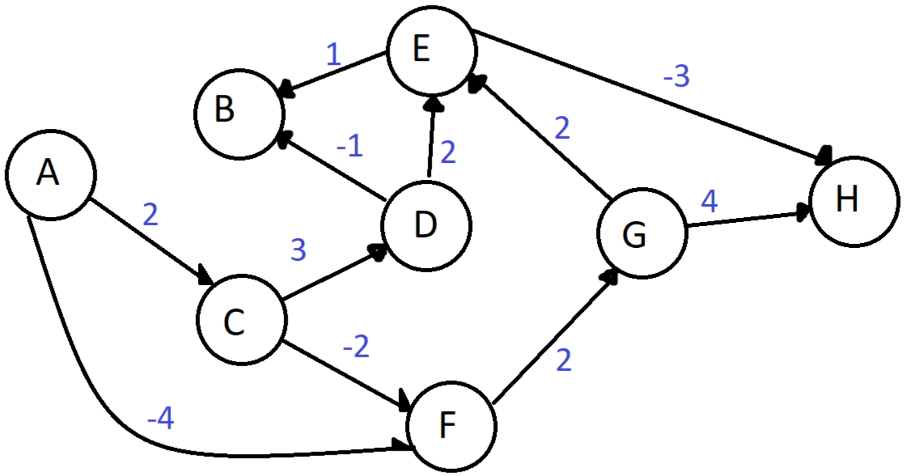
→ MinHeap instead of Queue

Negative Cycles

- For each algorithm , depends if it can detect neg. cycles or not .
- An algorithm CAN have neg. edges ONLY if it can detect neg. cycles

remember you can't fix neg. cycles . Can only detect .

Bellman's Ford



Time: $O(VE)$

Use Case: Negative-weights, single-sourced

Methodology: somewhat DP

1. loop $O(V+1)$ no. of iterations
2. Each iter, loop all edges.

	Base Case i=0	i=1	repeat i=2 ..... V+1
A	0	0	
B	∞	∞	
C	∞	2	
D	∞	∞	
E	∞	∞	
F	∞	-4	
G	∞	∞	
H	∞	∞	

update

i = V i = V + 1

if $V \rightarrow V+1$, there are any changes, then a negative cycle exists

$\boxed{0}\ A \xrightarrow{2} C \quad 0+2 = 2$

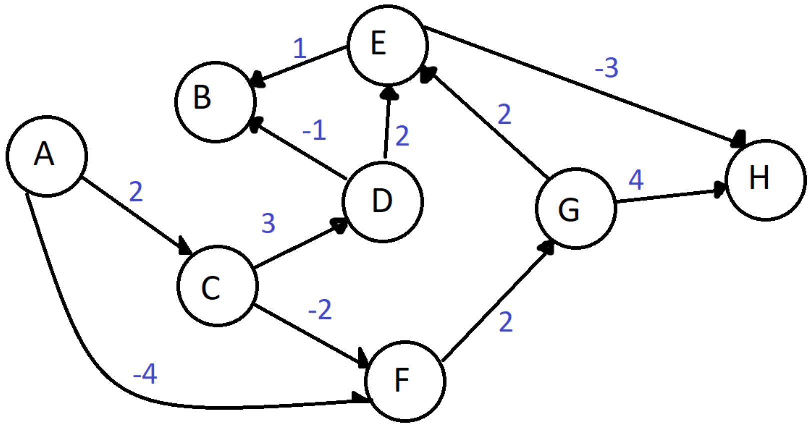
$\boxed{0}\ A \xrightarrow{-4} F \quad 0-4 = -4$

$\boxed{\infty}\ C \xrightarrow{-2} F \quad \infty - 2 = \infty$

⋮

loop all edges

Floyd Warshall's



Basically repeatedly form edges.

if $A \xrightarrow{2} B \xrightarrow{3} C \xrightarrow{1} D$, then

$A \xrightarrow{2} B \xrightarrow{3} C$ = $A \xrightarrow{5} C$ [A][C] = 5

if $A \xrightarrow{5} C \xrightarrow{1} D$, then

$A \xrightarrow{6} D$ [A][D] = 6

⋮

repeat for $O(v^3)$ times.

Time : $O(v^3)$

Use Case : negative-weights, all-pairs

Methodology :
- 2D-matrix
- transitive relationship
- brute force

Neg. Cycles

if

	A	B	C	D
A	-2	x	x	x
B	x	0	x	x
C	x	x	0	x
D	x	x	x	0

Diagonal, if not 0, then negative cycle exists, if node A travels to itself is -2, then repeating it gives [A][A] = $\underbrace{-4, -8}_{\text{negative cycles}}$, -16 etc....