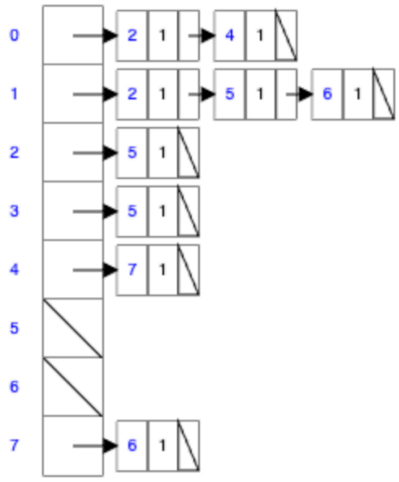
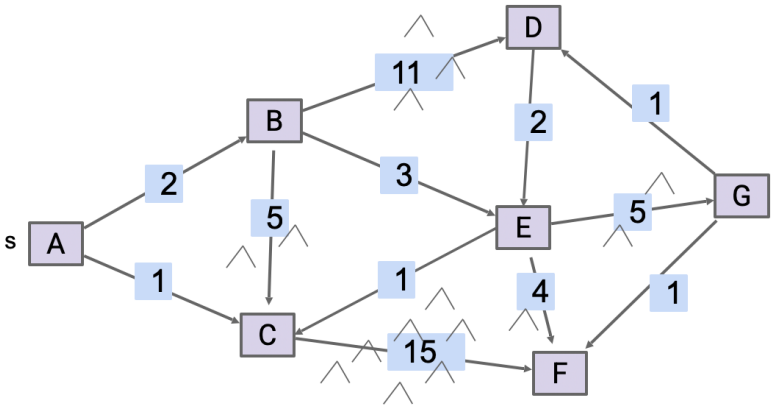
CS62: Fall 2025 | Lecture #21 (Shortest Paths) worksheet | Prof. Li

Last time: Given the following adjacency list, visualize the resulting digraph and run BFS on it starting at vertex 0. In what order did you visit the vertices? (Ignore the black 1 values in the list, they're vertex-value pairs) Is every vertex reachable from 0?

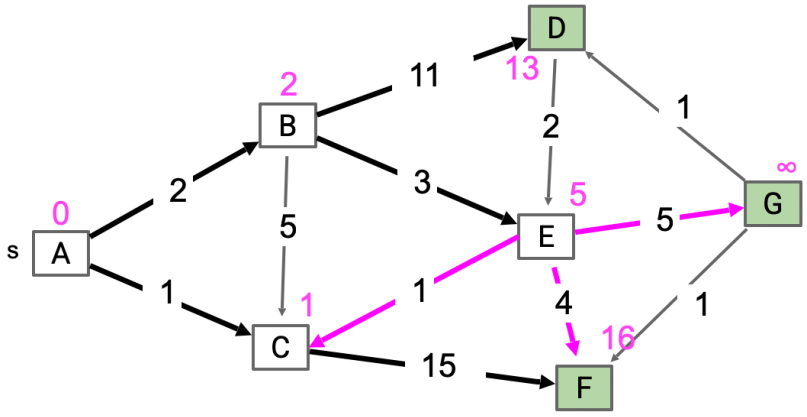


1. Find the single source shortest path from s: the shortest paths from source s to every other vertex.
   a. What data structure does your shortest paths look like?
   b. How many edges, as a function of V, are in your shortest paths?



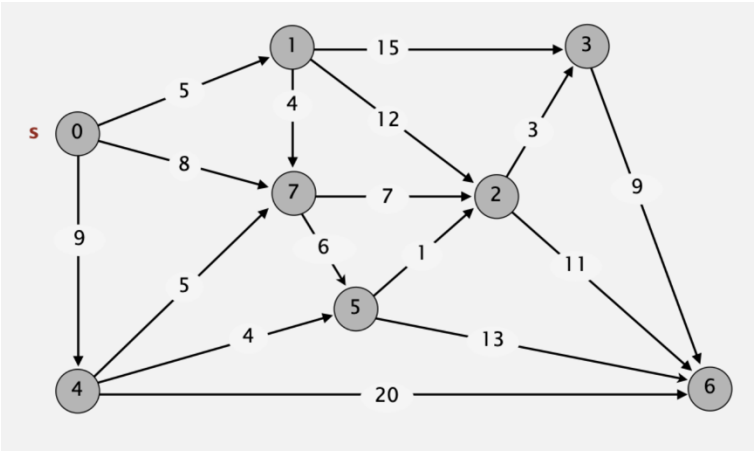
2. Show distTo, edgeTo, and fringe after relaxing on edge E.

Node	distTo	edgeTo
A	0	-
B	2	A
C	1	A
D	13	B
E	5	B
F	16	C
G	∞	-



Fringe: [(D: 13), (F: 16), (G: ∞)]

3. Run Dijkstra's algorithm to generate the shortest path tree from s below. Also keep track of distTo and edgeTo.



v	distTo	edgeTo
0		
1		
2		
3		
4		
5		
6		
7		