Lecture 19

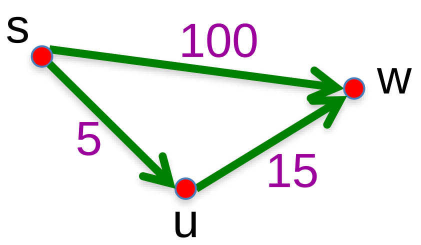
CSE 331

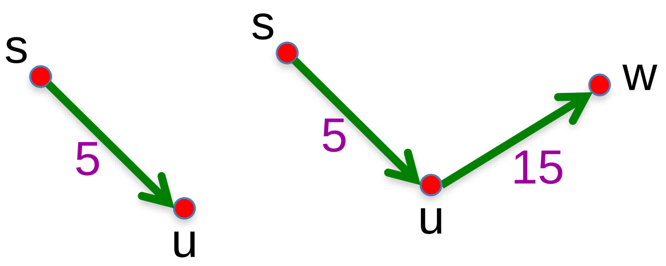
Shortest Path problem

Input: *Directed* graph G=(V,E)

Edge lengths, l_e for e in E

"start" vertex s in V



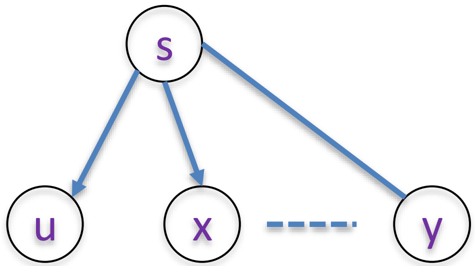


Output: Length of shortest paths from s to all nodes in V

Towards Dijkstra's algo: part one

Determine d(t) one by one

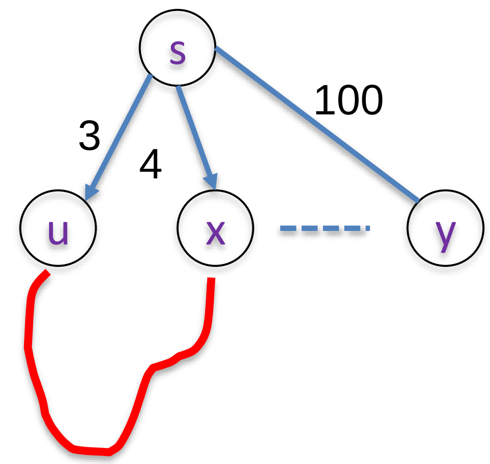
$d(s) = 0$



Towards Dijkstra's algo: part two

Determine $d(t)$ one by one

Let u be a neighbor of s with smallest $l_{(s,u)}$



$$d(u) = l_{(s,u)}$$

Length of is $\geq$ 0

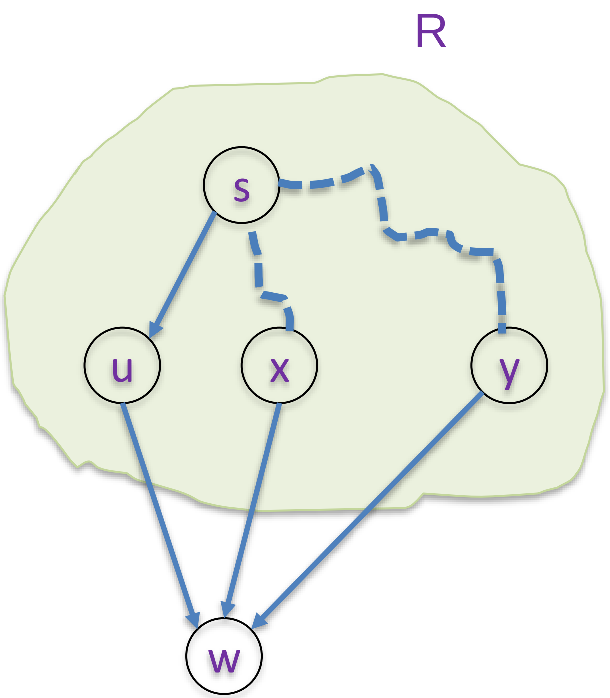
Not making any claim
on other vertices

Towards Dijkstra's algo: part three

Determine d(t) one by one

Assume we know d(v) for every v in R

Compute an upper bound d'(w) for every w not in R



$$d(w) \leq d(u) + l_{(u,w)}$$

$$d(w) \leq d(x) + l_{(x,w)}$$

$$d(w) \leq d(y) + l_{(y,w)}$$

$$d'(w) = \min_{e=(u,w) \text{ in } E,\ u \text{ in } R} d(u)+l_e$$

Reading Assignment

Sec 4.4 of [KT]