CSE 101-02 3-14-23

- SETS: Due Sunday 3/19 11:59 PM
- final exam

  101-01 : Wed 3/22 9-11 am

  101-02 : Tue 3/21 4-6 PM

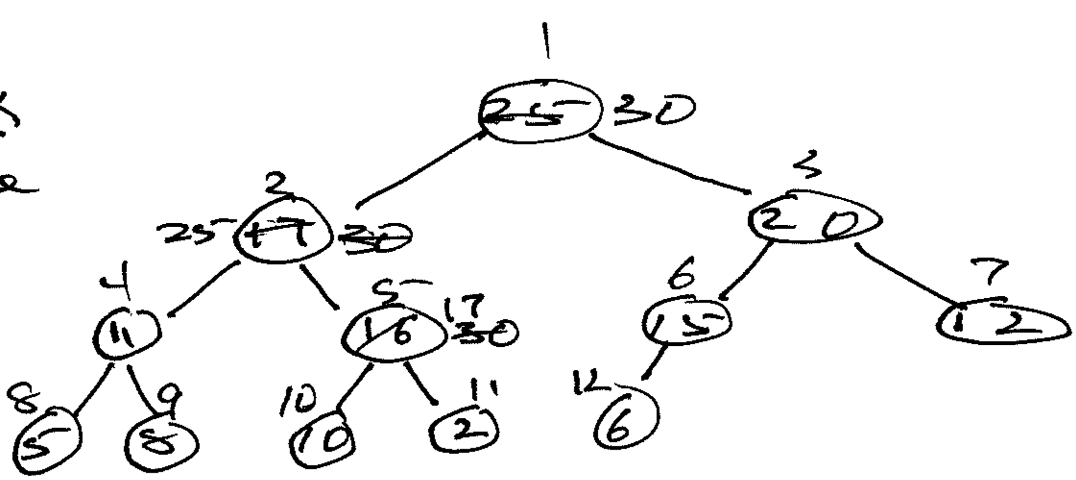
- Pa8: ext. 3 days to Fri 3/17

HeapIncreaseKey(A, i, k):

Pre: $1 \le i \le heapSize[A]$

Ex

Tree

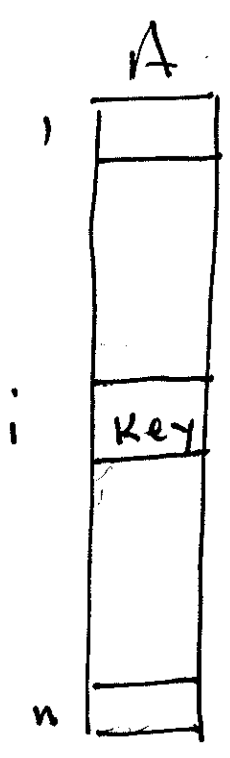


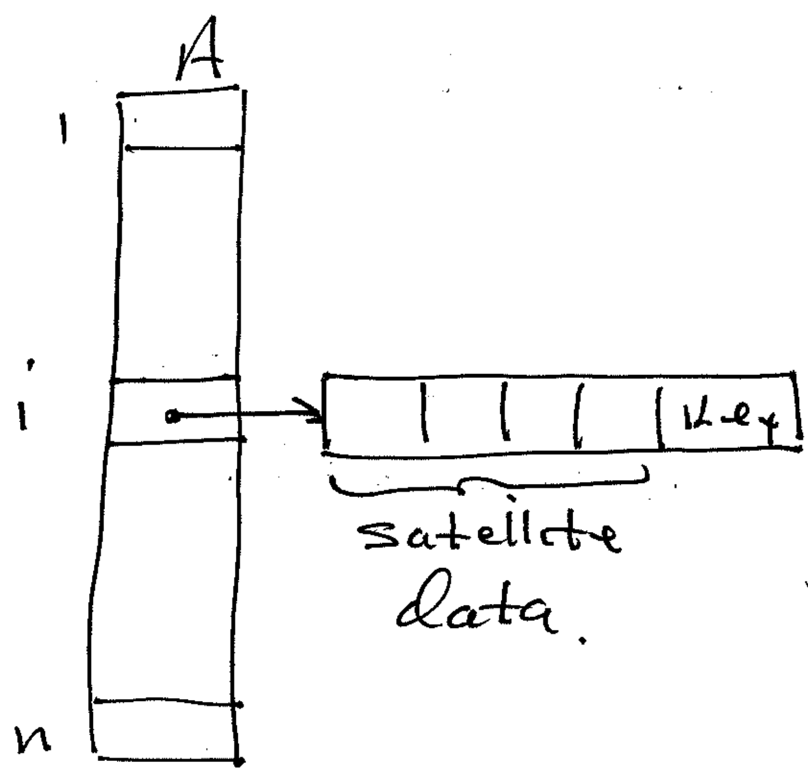
Array:

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
A	~~25~~	~~17~~	20	11	~~16~~	15	12	5	8	10	2	6	·	·	·	·
	30	~~30~~ 25			~~30~~ 17											

HeapIncreaseKey(A, 5, 30)

<u>Our Picture</u> <u>General Picture</u>





SSSP in a Weighted Graph

(chap 24 of CLRS)

Defn

- Weighted graph: $G = (V, E, w)$
  where
  $w : E \to \mathbb{R}$ weight-fcn

- Path weight: $P: v_0, v_1, v_2, \ldots, v_k$

$$w(P) = \sum_{i=1}^{k} w(v_{i-1}, v_i)$$

- Weight of a subgraph H

$$w(H) = \sum_{e \in E(H)} w(e) .$$

- <u>shortest path weight</u>

$$\delta(x,y) = \begin{cases} \min\{w(p) \mid p \text{ is an } x\text{-}y \text{ path}\} \\ \infty \text{ if } y \text{ not reachable from } x \end{cases}$$

- <u>shortest path</u>:

an x-y path p such that

$$w(p) = \delta(x,y).$$

- <u>SSSP</u>:

Given a weighted graph $G=(V,E,w)$, and $s \in V$. Determine

(1) $\delta(s,x)$ for all $x \in V$.

(2) for each x with $\delta(s,x) < \infty$, find a shortest s-x path.

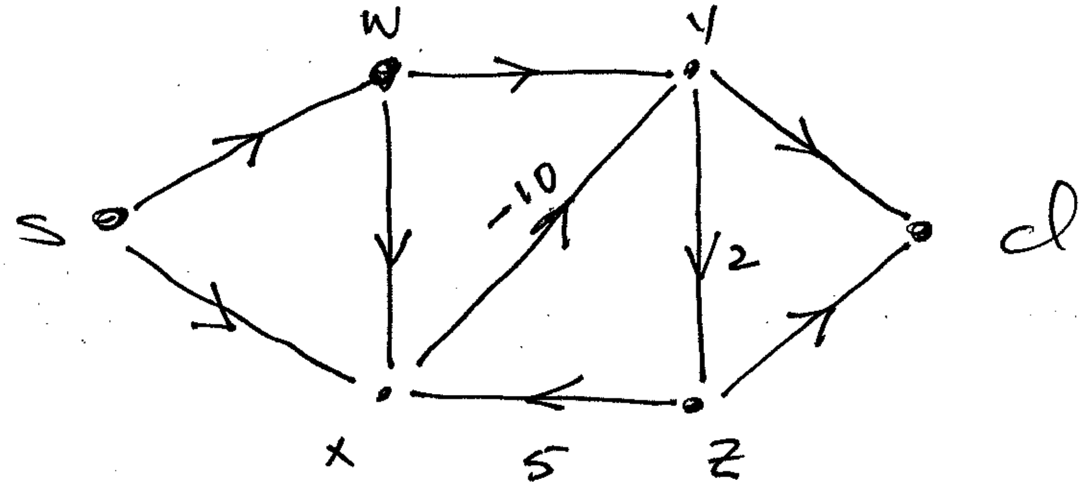
Two Algorithms

- Dijkstra: requires non-negative weights, fast
- Bellman Ford: slow, works with pos & neg. edges.

Both actually find minimum weight walks, rather than paths.

Problem: what if there is a negative weight cycle reachable from the source?

Ex.



$w(x, y, z, x) = -3$

we can always find a "shorter" walk by traversing (x, y, z, x) more times.

Common Infrastructure

$P[x]$: Parent, encodes a shortest Paths tree

$d[x]$: estimate of $\delta(s, x)$

$P[\cdot]$ defines:

$$V_p = \{x \in V \mid P[x] \neq nil\} \cup \{s\}$$
$$= \{\text{vertices reachable from } s\}$$

$$E_p = \{(P[x], x) \mid P[x] \neq nil\}$$

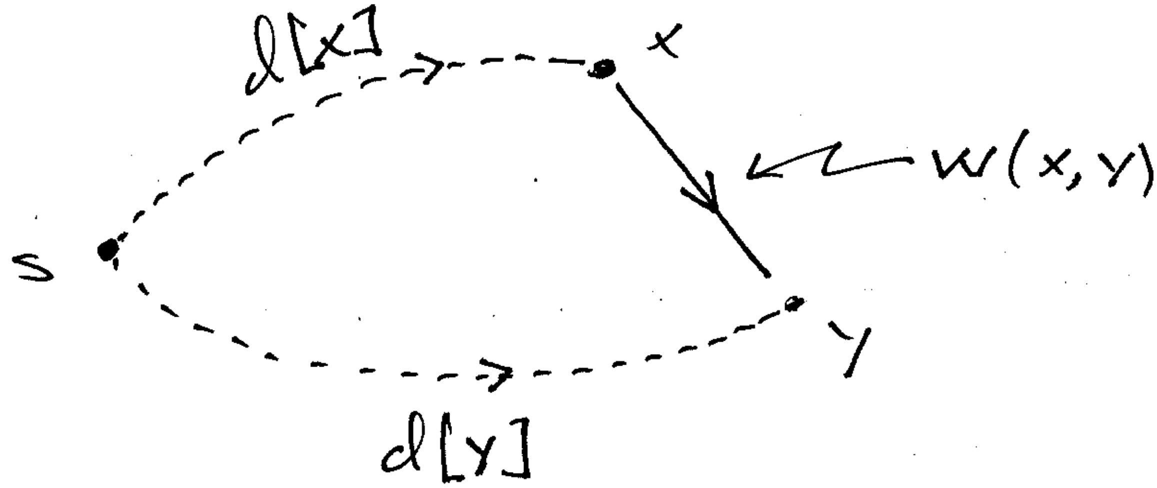
Predecessor Subgraph: (V_p, E_p)

to get shortest Paths, use PrintPath(G, s, x)

Two subroutines!

- Initialize()

- Relax()
  - Picture:



<u>note</u>:

- Relax changes only values of y
- after Relax(x,y), then
  $$d[y] \le d[x] + w(x,y)$$
  is true.

Logical infrastructure :

Lemma 1 :

let $x \in V(G)$ and supp. after Initialize(G, s), some seq. of calls to Relax$(\cdot, \cdot)$ results in $d[x] < \infty$. Then G contains an s-x path of weight $d[x]$.