

CSE 101 12-10-20

- Final Exam : wed. Dec. 16
- SETs : closes Sun. Dec. 13 at midnight
- Pat ext 1 last day.

Sec. 24.3

Dijkstra(G, s)

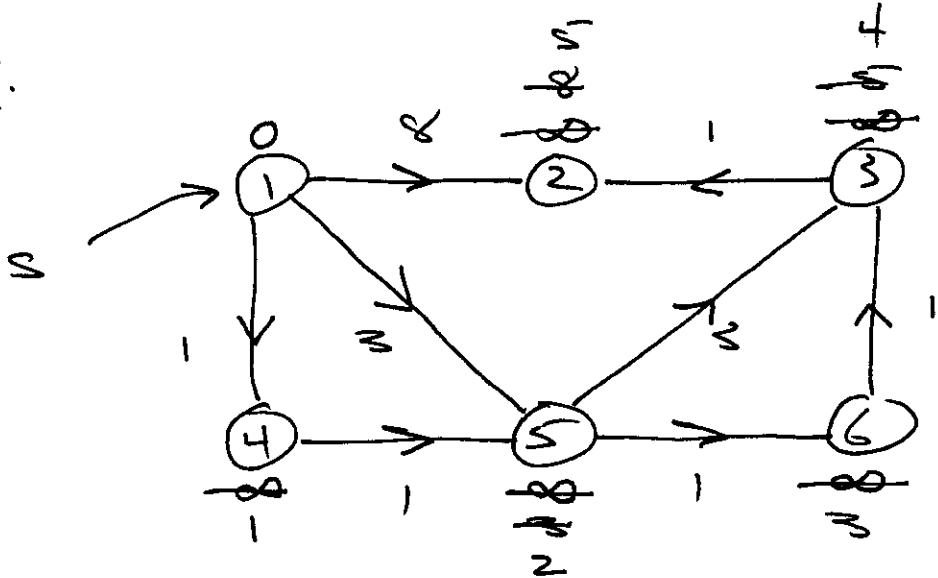
⋮
Pseudo-code

Runtime of Dijkstra

$n = |V|, m = |E|$. If P.Q. implemented as min-heap, then

$$\text{cost} = \Theta((n+m) \log n)$$

Ex.

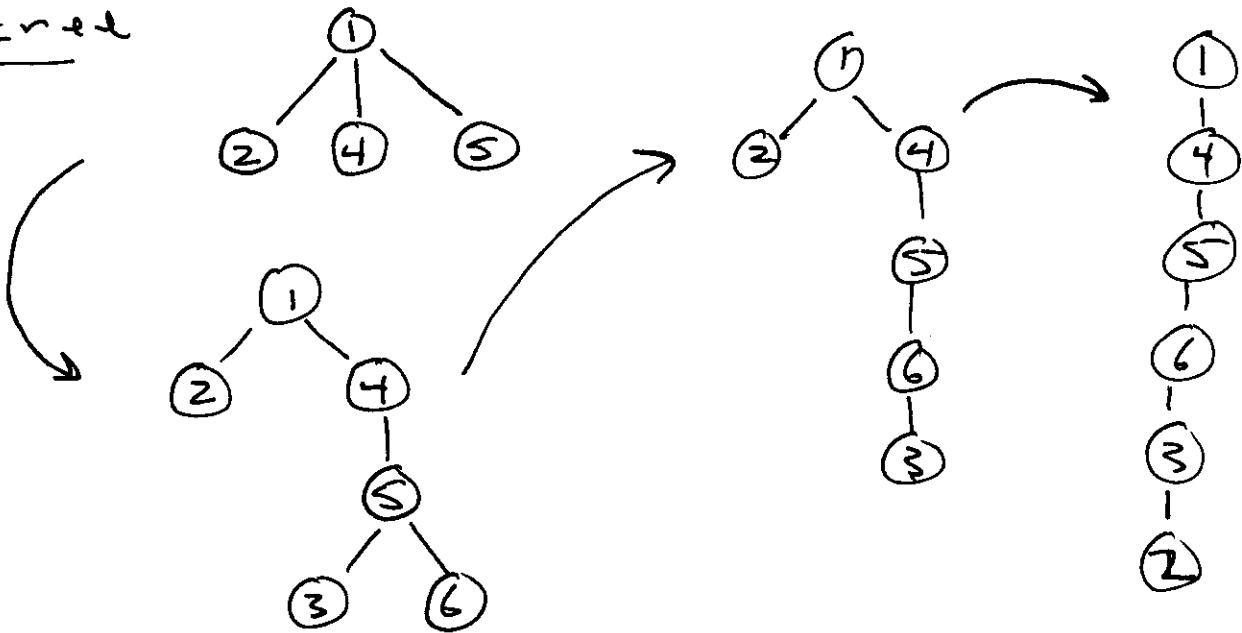


source = 1

P.Q. : ✓ ✗ ✗ ✗ ✗ ✗

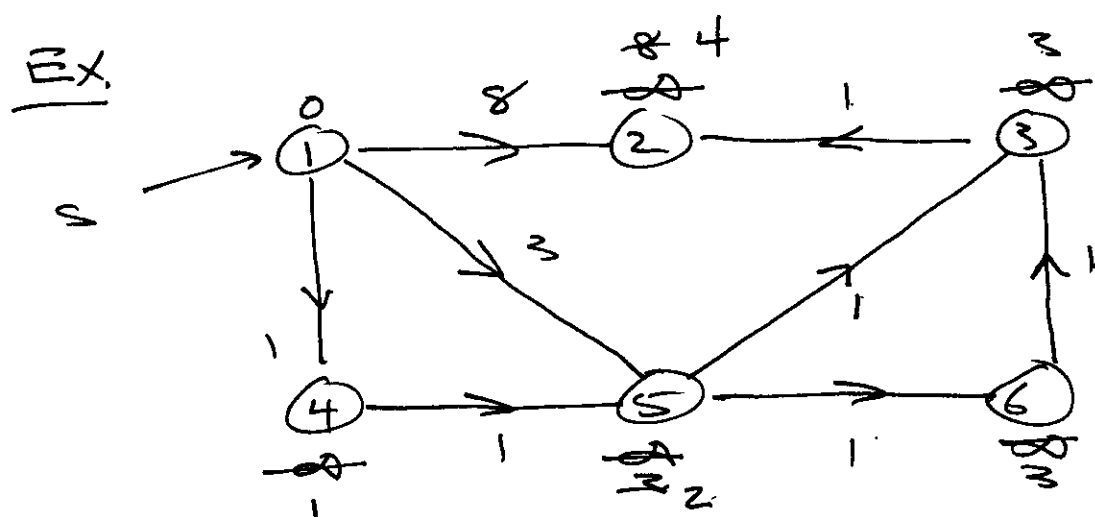
remove order : 1, 4, 5, 6, 3, 2

tree



Convention:

If there is a tie, i.e. 2 (or more) vertices with same min d-value, choose the one with smaller label.



P.Q.	1	2	3	4	5	6
d	0	4	3	1	2	3
P	n	1 3	5	1	1 4	5

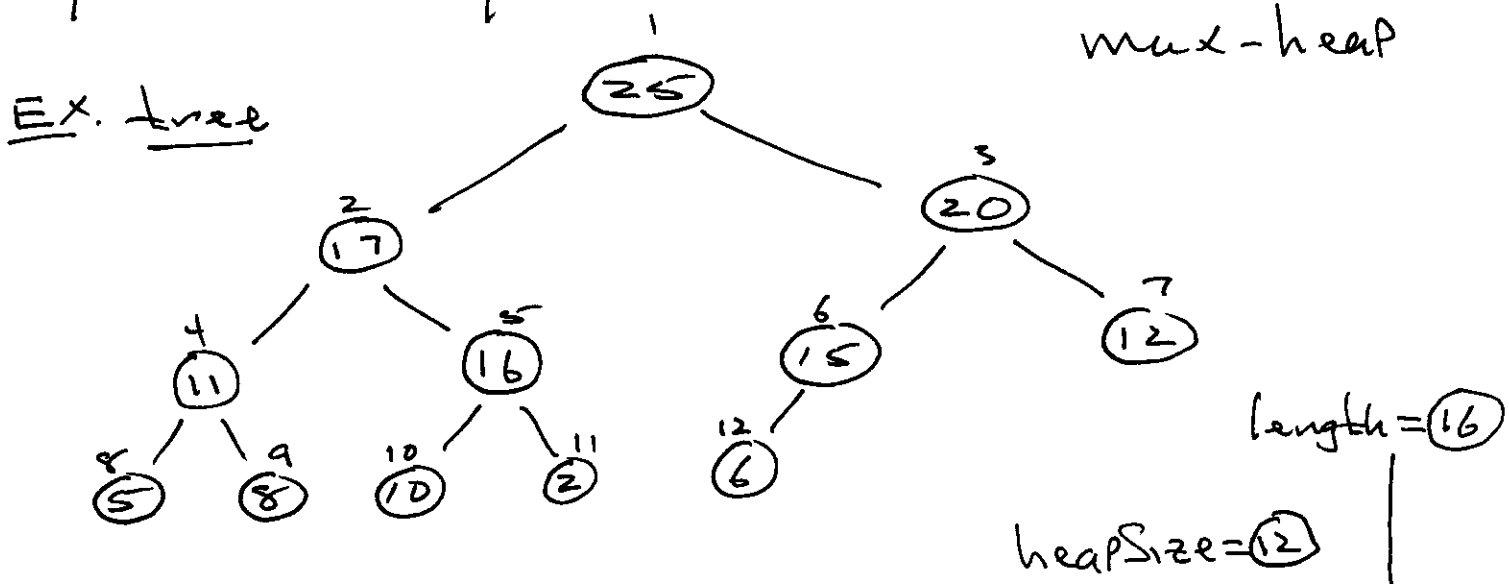
extract order: 1, 4, 5, 3, 6, 2

How do we implement a (min or max) Priority Queue.

New data structure : (Binary) Heap

2 kinds : { min heap
max heap ← illustration

A heap is an A.C.R.T. represented by an array.



array

A

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
25	17	20	11	16	15	12	5	8	10	2	6	.	.	.	.

Two array attributes:

- $\text{length}[A]$
- $\text{heapSize}[A]$

three helper functions:

$$\text{Parent}(i) = \lfloor \frac{i}{2} \rfloor \quad (\text{for } i \geq 2)$$

$$\text{left}(i) = 2i$$

$$\text{right}(i) = 2i + 1$$

2 heap properties:

- max heap property: $A[\text{Parent}(i)] \geq A[i]$
- min " " : $A[\text{Parent}(i)] \leq A[i]$

How to build one?

- Heapify
- BuildHeap

why build one?

- Heapsort
- implement a Priority queue.

Insert

Max

DeleteMax

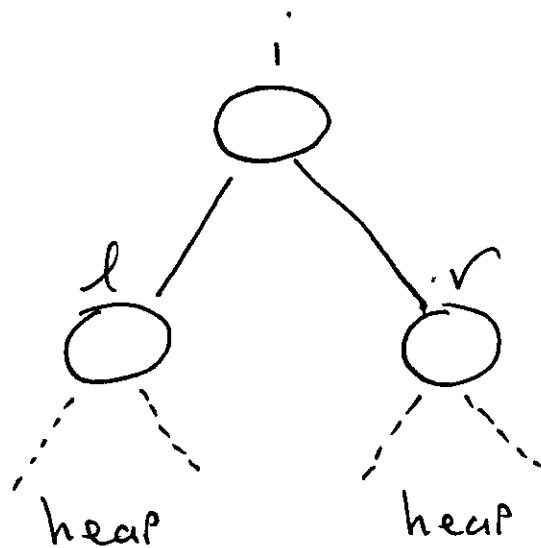
ExtractMax

IncreaseKey

$\text{Heapify}(A, i)$

17

Assumes subtrees rooted at $\text{left}(i)$ and $\text{right}(i)$ are valid heaps



Goal of $\text{Heapify}(A, i)$: establish max heap property at i .

→ see pseudo-code.