Lec 24 Meldable Priority Queues

Meld operation

meld : $Q \times Q \to Q$ that unions two priority queues

Possible impl: ($m \le n$

	insert	delMin	fromSeq	meld
- balanced tree	$\lg n$	$\lg n$	$n \lg n$	$m \lg(\frac{n}{m}+1)$
- binary heap	$\lg n$	$\lg n$	n	$n+m$ or $m \lg n$
- leflist heap	$\lg n$	$\lg n$	n	$\lg(m+n)$

impl dependent

operations based on meld

```
datatype PQ = Empty | Node (k × PQ × PQ)
singleton x = Node (x, Empty, Empty)
insert (Q, x) = meld (Q, singleton x)
delMin Q = case Q of
             Empty => (Q, None)
           | Node (k, L, R) => (meld (L, R), Some k)
fromSeq S = reduce meld Empty ⟨singleton x : x ∈ S⟩
```

cost analysis assuming meld is $O(\lg(m+n))$

insert $\quad \lg(n+1) = \lg n$

delMin $\quad \lg n$

fromSeq $\quad W(n) = 2W(\frac{n}{2}) + O(\lg n)$

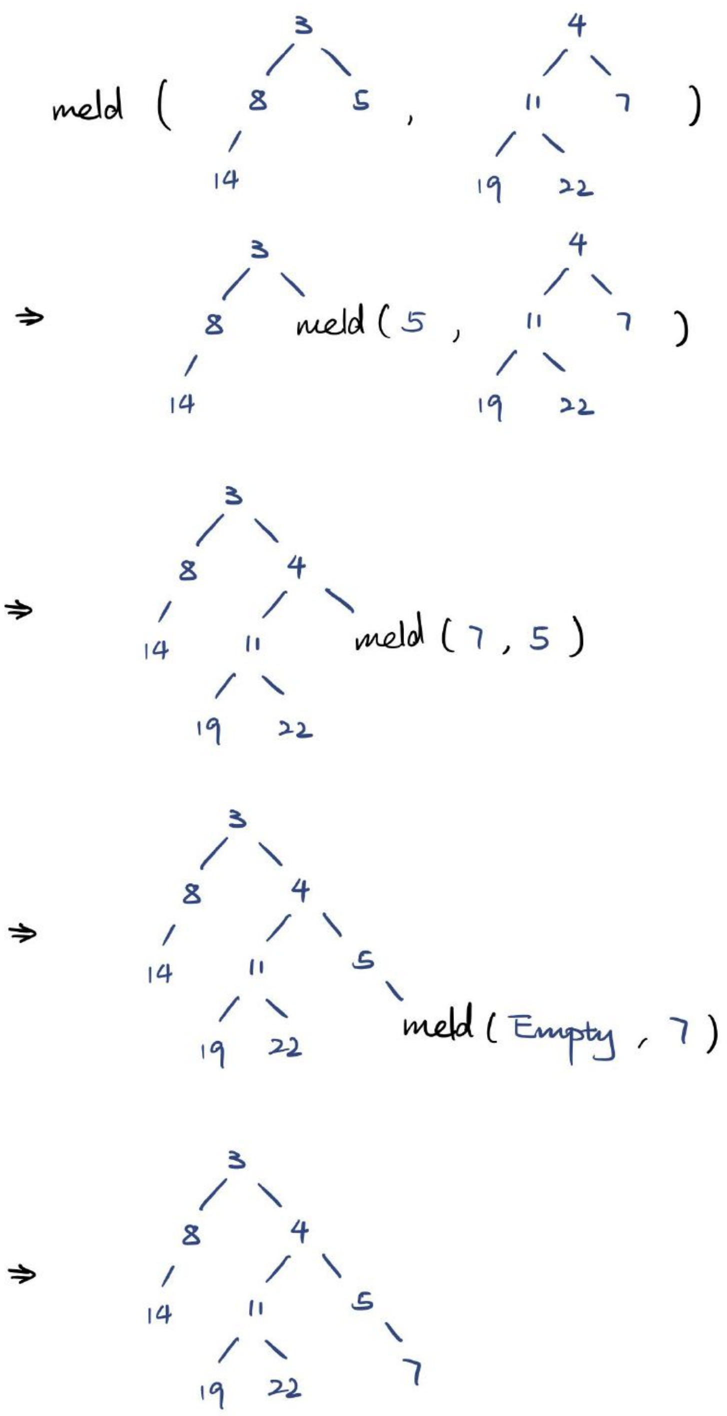
$\qquad \in O(2^{\lg n}) = O(n)$

$\qquad S(n) = S(\frac{n}{2}) + O(\lg n)$

$\qquad \in O(\lg^2 n)$

Bad meld (correct but out of bound)

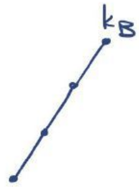
```
meld (A, B) = case (A, B) of
  (_, Empty) => A
 | (Empty, _) => B
 | (Node (kA, LA, RA), Node (kB, LB, RB)) =>
   if kA < kB then
     Node (kA, LA, meld (RA, B))
   else
     Node (kB, LB, meld (RB, A))
```

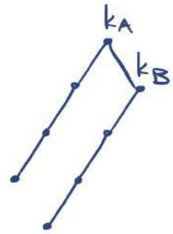


Cost analysis

Observe we only recurse down right subtrees (right spine)

So if right spine is short, we're efficient meld (

, 

) $\overset{O(1)}{\Rightarrow}$ WLOG $k_A < k_B$ 

<u>Fact</u> Cost meld $\in O(|\text{right spine of } A|)$

Leftist queue

<u>Def</u> rank Q := # nodes in right spine

<u>Def</u> leftist property:

$\forall$ Node(_, L, R) $\in$ PQ , rank R $\leq$ rank L

<u>Impl</u>

```
datatype PQ = Empty | Node ( int x k x PQ x PQ )
rank Q  = case Q of
            Empty => 0
          | Node (r, _, _, _) => r
node' (k, A, B) = if rank B < rank A then
                     Node ( rank B + 1 , k , A , B )
                  else
                     Node ( rank A + 1, k , B , A )
```

↑ maintains leftist property

<u>Proof</u>

Let $m(r)$ be min size of any leftist heap of rank r.

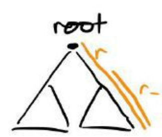
Claim : $m(r) = 2^r - 1$.

<u>BC</u> $r = 0 \Rightarrow m(0) = 0 = 2^0 - 1$ ✓

<u>IC</u> $m(r) = \underbrace{1}_{\text{root}} + \underbrace{m(r-1)}_{\text{left, smallest case}} + \underbrace{m(r-1)}_{\text{min of right}}$

$= 1 + 2(2^{r-1} - 1)$

$= 2^r - 1$



So size is exponential to rank

<u>Coro</u> rank Q $\leq \lg(|Q| + 1)$

proof is that $|Q| \geq 2^{\text{rank } Q} - 1$

$|Q| + 1 \geq 2^{\text{rank } Q}$

$\lg(|Q| + 1) \geq \text{rank } Q$