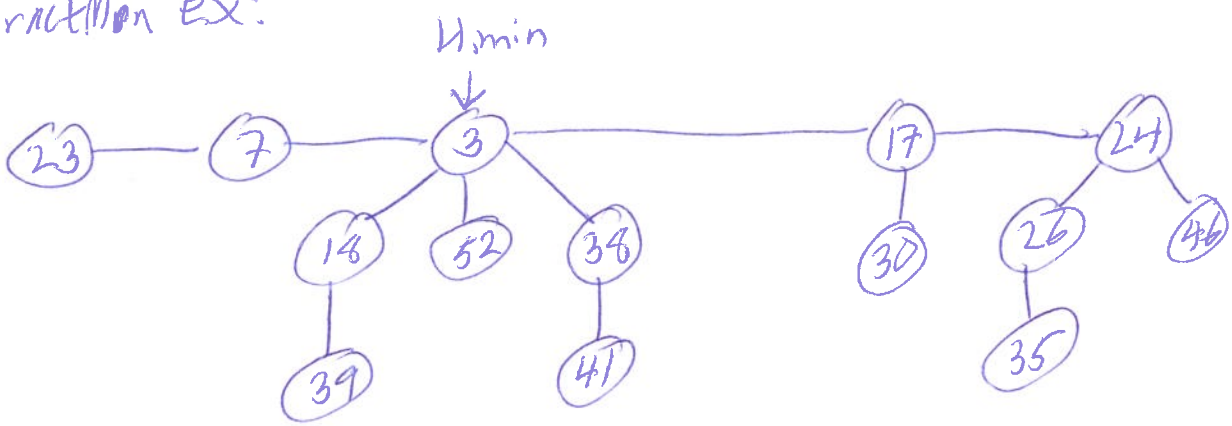
CSCE 750
11/2/2023

Fib heap analysis & operations

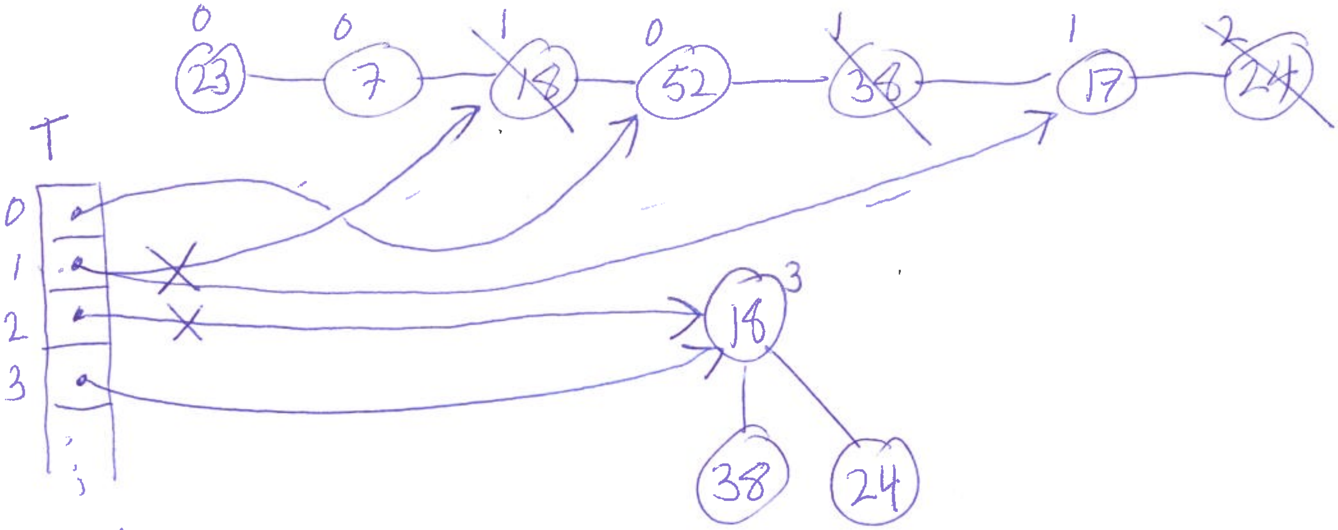
ExtractMin(H)

- Remove H.min from the root list $\boxed{O(1)}$
- Promote each child of H.min to be a root
- "Consolidate":
  - Use a direct address table (array) keyed to the degrees of the root nodes
  - Scan through the root list
  - If 2 trees of same degree d are found, combine (as with a binomial heap) into a tree of degree d+1

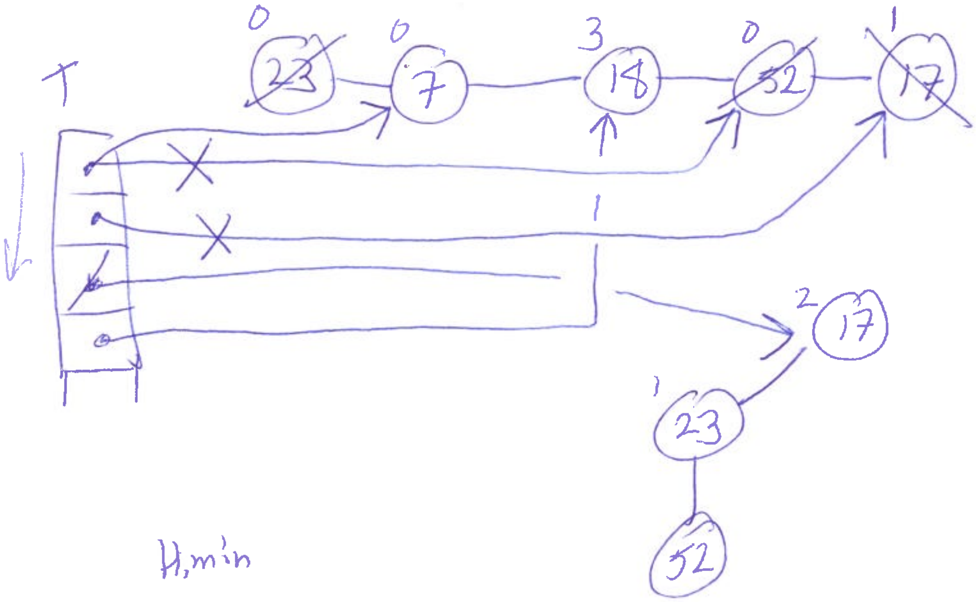
ExtractMin Ex:



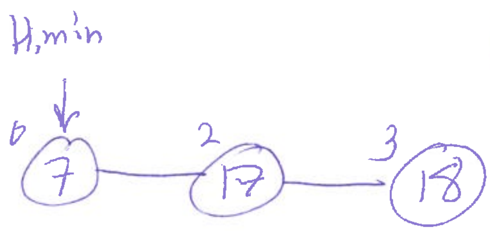
⇓



Redraw:



$$\Phi = t(H) + 2m(H)$$



Let $D(n)$ be the max degree in any Fib heap of size n.
Can show that $D(n) = O(\lg n)$

$T[0 \ldots D(n)]$

Analysis of ExtractMin

Amortized cost =

$$\underbrace{\underbrace{t(n)}_{\text{amortized analysis!}} + D(n) + 1}_{\text{actual cost}} + \underbrace{\overbrace{D(n)+1}^{\#\text{ trees}} + \cancel{2m(H)}}_{\Phi_{after}} - \underbrace{(t(H) + \cancel{2m(H)})}_{\Phi_{before}}$$

$= 2D(n) + 2 = O(D(n)) = O(\lg n)$

DecreaseKey(H, x, k) – decrease key of x down to k

- If x is a root or if x's parent has key $\leq k$, then decrease x's key to k (done).
- Otherwise, (1) cut x out of its tree, make it (with its children) the root of a new tree, and set its key to k. Update H.min if necessary
  - (2) If x's former parent is unmarked, then mark x's parent. (done)
  - (3) If marked, cut x's parent out of the tree & make it a new root. Mark x's former parent's former parent if unmarked, otherwise continue up the until root or unmarked node.

cascading cut

Analysis of DecreaseKey: Let c be the number of cascading cuts

Actual cost: $c + O(1)$

$\Delta\Phi$: $t(H)$ increases by c

$m(H)$ decreases by at least $c-2$

$\Delta\Phi = \Delta t(H) + 2\Delta m(H) = c - 2(c-2) = 4 - c$

$\therefore$ Amortized cost $= c + O(1) + 4 - c = O(1)$

Delete(x) : DecreaseKey(H, x, $-\infty$); ExtractMin(H)

Amortized cost $O(1) + O(\lg n) = O(\lg n)$

Need a $O(\lg n)$ bound on $D(n)$

Let size(x) = the size of the tree rooted at x, for any node x.

Lemma: Let x be a node of degree k. Then $\text{size}(x) \geq F_{k+2}$, where

$F_0 := 0$

$F_1 := 1$

$\vdots$

$F_n := F_{n-1} + F_{n-2}$

$\vdots$

Fibonacci sequence

Proof (omitted);

Let $\phi := \frac{1+\sqrt{5}}{2} \approx 1.618\ldots$ (Golden Ratio)

$\phi' := \frac{1-\sqrt{5}}{2}$ $(-1 < \phi' < 0)$

Lemma: For any $k \geq 0$,

$$F_k = \frac{\phi^k - (\phi')^k}{\sqrt{5}}$$

(Proof by induction on k)

Cor: $F_k = \Theta(\phi^k)$

Cor: $\text{size}(x) = \Omega(\phi^k)$

~~Cor:~~ So $\text{size}(x) \geq c \cdot \phi^k$ (some constant $c > 0$)

$$\log_\phi(c \cdot \phi^k) \leq \log_\phi \text{size}(x) \leq \log_\phi n$$

H has size n

$$\parallel$$

$$k + \log_\phi c$$

$$D(n) = k = O(\log_\phi n) = O(\lg n) \quad //$$