§3 Amortized Analysis

KAIST
CS500 M. Ziegler

Motivating example: Repeated binary increment,
#bit flips when counting to n? (+decrement?)

Definition (amortized cost):
Let $\mathcal{A}$.**Method**$_1$, …, $\mathcal{A}$.**Method**$_k$ denote an implementation of an abstract data type.

Let $T(n)$ denote the worst-case cost of <u>any</u> sequence of n calls of $\mathcal{A}$'s methods.

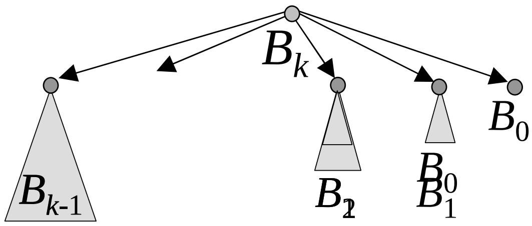
Then amortized cost of $\mathcal{A}$ is $T(n)/n$.

Don't confuse:
- amortized cost
- average-case cost
- expected cost

Potential method of analysis:
Let c_j denote cost of j-th operation,
Φ_j:=#**1**s in counter after j-th op. ⇔ before $(j+1)$-st op.
$\Rightarrow c_j + \Phi_j - \Phi_{j-1} \leq 2$, $\Phi_0 = 0$, $\Phi_j \geq 0$ $\sum_{1\leq j\leq n} c_j/n \leq 2 - \Phi_n/n$

§3 *Relaxed* Binomial Trees

KAIST
CS500 M. Ziegler



"Mark" indicates j-th child having order $=j$-2

A <u>*relaxed* Binomial Tree</u> of order $k\geq 1$ consists of a root with k children, the j^{th} being a relaxed binom.tree of order $\geq j$-2

Lemma: A relaxed binomial tree of order k has $\geq F_{k+2} \geq \Omega(1.6^k)$ nodes

Merge

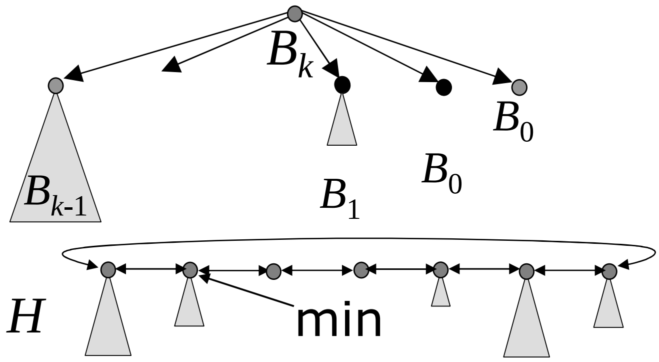
Prune

$\sum_{j>0} j\cdot 2^{-j} = 2$: $\sum_{j>0} j\cdot q^j = q\cdot\partial_q \sum_{j\geq 0} q^j = q\cdot\partial_q\, 1/(1-q) = q/(1-q)^2$

$\varphi := (1+\sqrt{5})/2 \approx 1.618$

$1 + F_1 + F_2 + \ldots + F_k = F_{k+2} \geq \varphi^k$ Fibonacci no.s F_k

§3 Fibonacci Heaps



A *relaxed* Binomial Tree of order $k \geq 1$ consists of a root with k children, the j^{th} being a relaxed binom.tree of order $\geq j$-2

Lemma: A relaxed binomial tree of n nodes has order $\leq O(\log n)$

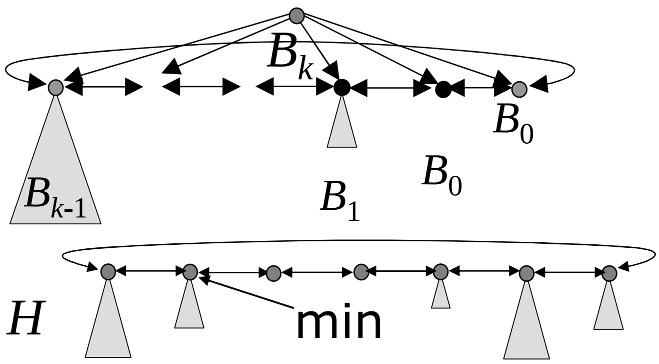
A Fibonacci Heap H is a list of t heap-ordered relaxed binomial trees with pointer to the min.

amor-tized cost

Extract min.key: $O(\log n)$
Decrease key: $O(1)$
Merge two Fib.heaps: $O(1)$
Insert element: $O(1)$
Create 1-elem.Fib.heap: $O(1)$

KAIST
CS500 M. Ziegler

§3 Extract Minimum



A *relaxed* Binomial Tree of order $k \geq 1$ consists of a root with k children, the j^{th} being a relaxed binom.tree of order $\geq j$-2

Lemma: A relaxed binomial tree of n nodes has order $\leq O(\log n)$

A Fibonacci Heap H is a list of t heap-ordered relaxed binomial trees with pointer to the min.

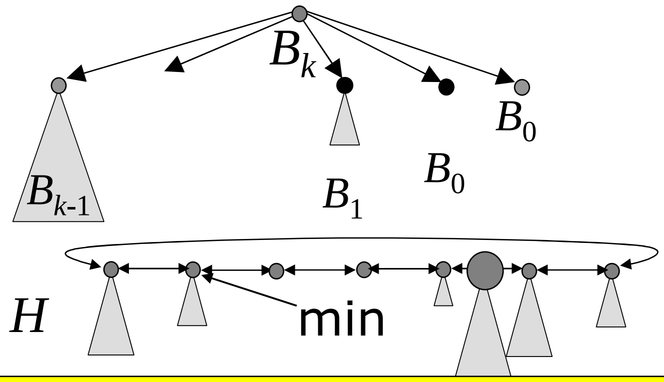
Extract min.key: $O(\log n)$ amor-tized cost

- Delete target of min.ptr
- Merge two Fibonacci heaps.
- Consolidate s.t. each tree order occurs only once!

$c_j + \Phi_j - \Phi_{j-1} \leq O(\log n)$, $\Phi_0 = 0$, $\Phi_j \geq 0$

Potential $\Phi = O(t)$

§3 Decrease Key

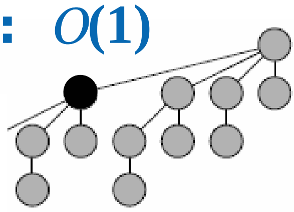


A *relaxed* Binomial Tree of order $k \geq 1$ consists of a root with k children, the j^{th} being a relaxed binom.tree of order $\geq j$-2

Lemma: A relaxed binomial tree of n nodes has order $\leq O(\log n)$

A Fibonacci Heap H is a list of t heap-ordered relaxed binomial trees with pointer to the min.

Decrease key: $O(1)$ amortized cost

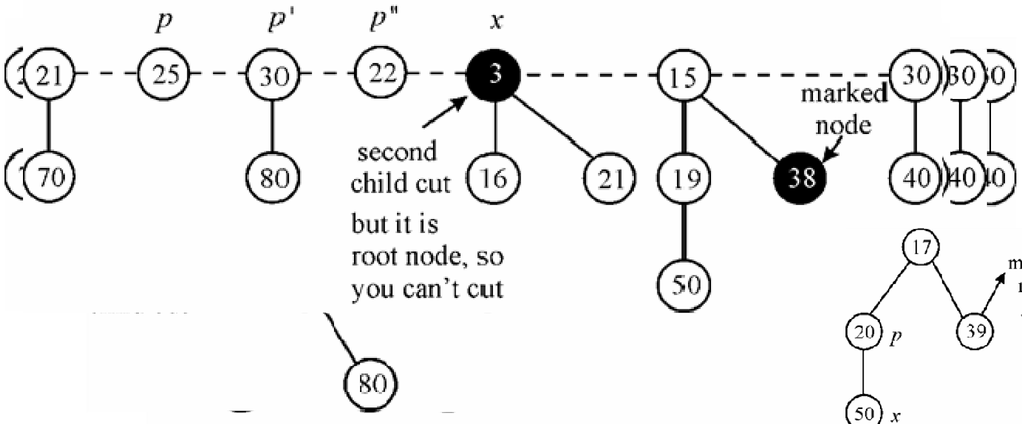


- cut subtree
- **mark** parent
- if already **marked**: reset, cut & cascade

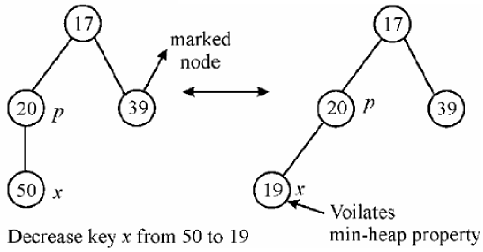
$c_j + \Phi_j - \Phi_{j-1} \leq O(1)$, $\Phi_0 = 0, \Phi_j \geq 0$

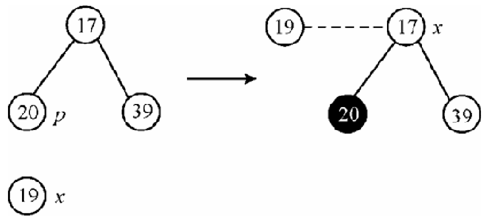
Potential $\Phi = O(t + 2m)$

§3 Cuts, Marks, and Cascading

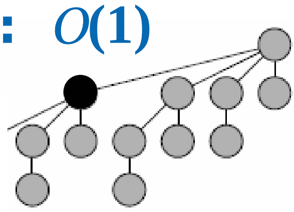


(decrease key of x from 42 to 21)





Decrease key: $O(1)$



- cut subtree
- **mark** parent
- if already **marked**: reset, cut & cascade

$c_j + \Phi_j - \Phi_{j-1} \leq O(1)$, $\Phi_0 = 0, \Phi_j \geq 0$

Potential $\Phi = O(t + 2m)$