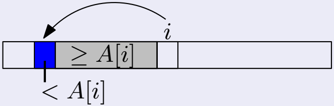
New Solution

Definition (Previous Smaller Values)

For array index i in A, let

$$PSV(i) = \text{argmax}\{k < i : A[k] < A[i]\}$$



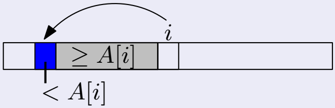
be the previous smaller value left of i.

New Solution

Definition (Previous Smaller Values)

For array index i in A, let

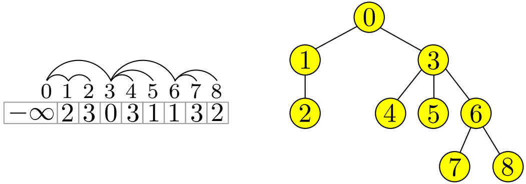
$$PSV(i) = \text{argmax}\{k < i : A[k] < A[i]\}$$



be the previous smaller value left of i.

Definition (2d-Min-Heap of array A)

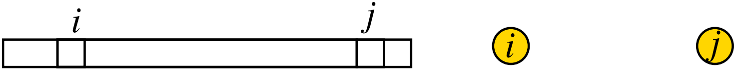
Ordered Tree on nodes $[1, n]$ defined by $parent(i) = PSV(i)$.



Relating 2d-Min-Heaps with RMQs

Lemma

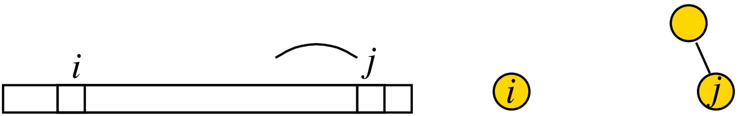
$\mathrm{RMQ}(i,j)$ *is given by the child of* $\mathrm{LCA}(i,j)$ *that is on the path to* j*.*



Relating 2d-Min-Heaps with RMQs

Lemma

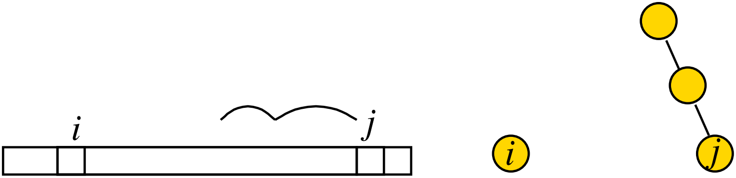
$\mathrm{RMQ}(i,j)$ *is given by the child of* $\mathrm{LCA}(i,j)$ *that is on the path to* j.



Relating 2d-Min-Heaps with RMQs

Lemma

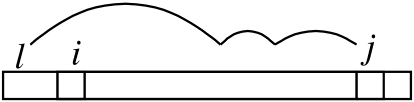
RMQ(i, j) *is given by the child of* LCA(i, j) *that is on the path to* j.

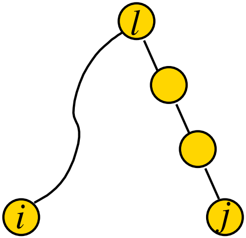


Relating 2d-Min-Heaps with RMQs

Lemma

RMQ(i, j) *is given by the child of* LCA(i, j) *that is on the path to* j*.*

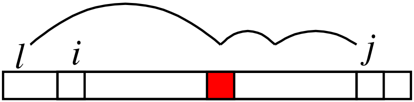


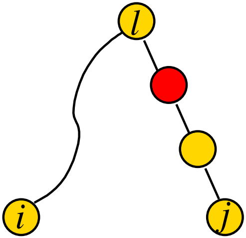


Relating 2d-Min-Heaps with RMQs

Lemma

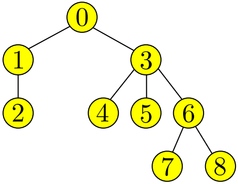
RMQ(i, j) *is given by the child of* LCA(i, j) *that is on the path to* j.





Representation and Construction of 2d-Min-Heaps

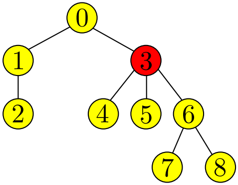
- Represent heap succinctly by DFUDS:
  - list degrees of nodes in pre-order:
  - node of out-degree $k \Rightarrow (^k)$
  - $\Rightarrow$ space $2n$ bits
  - $\Rightarrow$ array-index i corresponds to i'th ')'



$((()())((()))(()))$

Representation and Construction of 2d-Min-Heaps

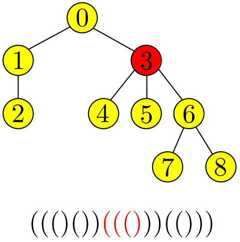
- Represent heap succinctly by DFUDS:
  - list degrees of nodes in pre-order:
  - node of out-degree $k \Rightarrow (^k)$
  - $\Rightarrow$ space $2n$ bits
  - $\Rightarrow$ array-index i corresponds to i'th ')'



((()())((()))(()))

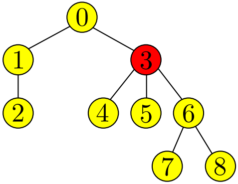
Representation and Construction of 2d-Min-Heaps

- Represent heap succinctly by DFUDS:
  - list degrees of nodes in pre-order:
  - node of out-degree $k \Rightarrow (^k)$
  - $\Rightarrow$ space $2n$ bits
  - $\Rightarrow$ array-index i corresponds to i'th ')'
- $\mathcal{O}(\frac{n \log\log n}{\log n})$-bit index for simulating $\mathcal{O}(1)$-LCAs (technical!)
- DFUDS can be constructed "in-place"



Representation and Construction of 2d-Min-Heaps

- Represent heap succinctly by DFUDS:
  - list degrees of nodes in pre-order:
  - node of out-degree $k \Rightarrow (^k)$
  - $\Rightarrow$ space $2n$ bits
  - $\Rightarrow$ array-index i corresponds to i'th ')'
- $\mathcal{O}(\frac{n \log\log n}{\log n})$-bit index for simulating $\mathcal{O}(1)$-LCAs (technical!)
- DFUDS can be constructed "in-place"



((()())((()))(()))

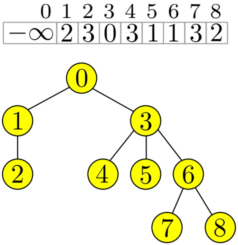
Theorem

There is a preprocessing scheme of optimal size $2n + o(n)$ bits for $\mathcal{O}(1)$-range minimum queries. Workspace is also $O(n)$ bits.

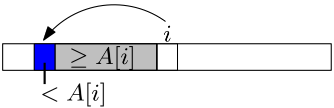
More Functionality: PSV

Definition (2d-Min-Heap)

Ordered Tree defined by $parent(i) = PSV(i)$.



- $PSV(i) = \max\{k < i : H[k] < H[i]\}$

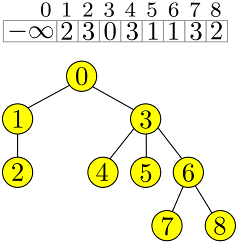


$\Rightarrow$ PSV simple (move to parent in $\mathcal{O}(1)$ time!)

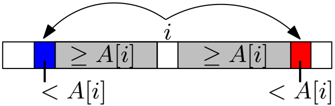
More Functionality: PSV

Definition (2d-Min-Heap)

Ordered Tree defined by $parent(i) = PSV(i)$.



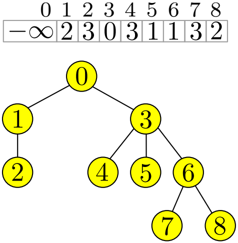
- $PSV(i) = \max\{k < i : H[k] < H[i]\}$
  $NSV(i) = \min\{k > i : H[k] < H[i]\}$



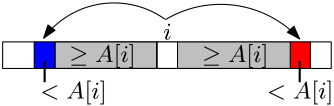
More Functionality: PSV

Definition (2d-Min-Heap)

Ordered Tree defined by $parent(i) = PSV(i)$.

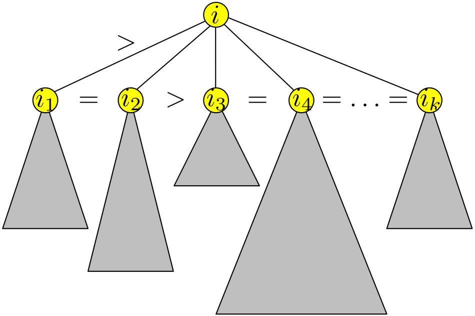


- $PSV(i) = \max\{k < i : H[k] < H[i]\}$
  $NSV(i) = \min\{k > i : H[k] < H[i]\}$

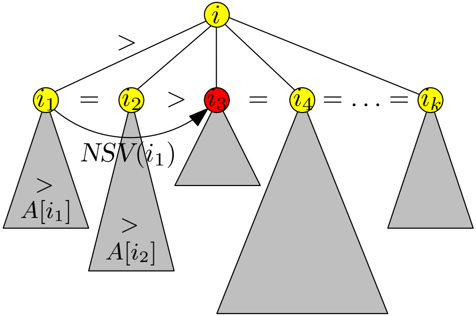


Can we also do NSV???

More Functionality: NSV

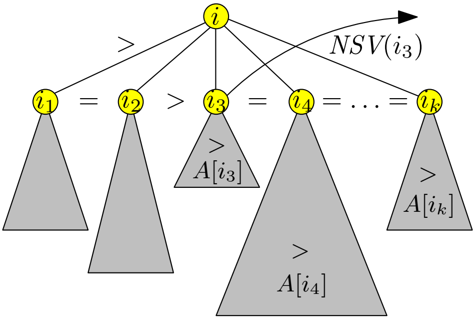


More Functionality: NSV



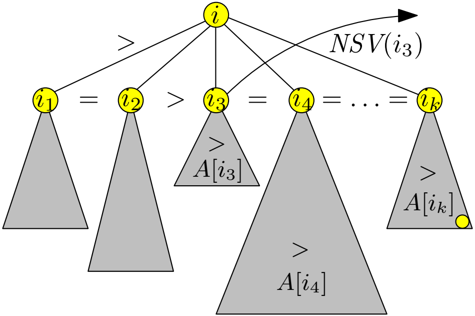
1. Find leftmost $<$-sibling to the right

More Functionality: NSV



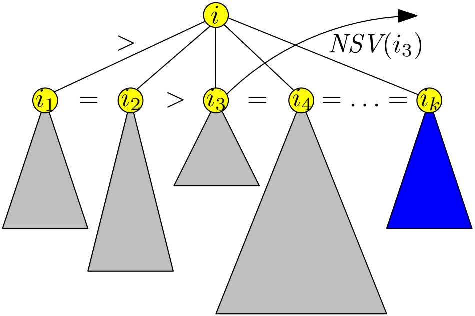
1. Find leftmost $<$-sibling to the right
2. If it does not exist…

More Functionality: NSV



1. Find leftmost $<$-sibling to the right
2. If it does not exist...

More Functionality: NSV



1. Find leftmost $<$-sibling to the right
2. If it does not exist…
3. … $NSV(i_3) = i_k + |T_{i_k}|$

More Functionality: NSV

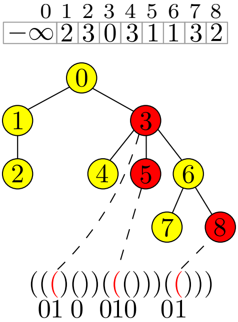
- Distinguish $=$- and $<$-siblings?

$\rightsquigarrow$ Mark $<$-children in additional bit-vector

- Bit-tricks for $\mathcal{O}(1)$-computations

Theorem (Extended 2d-Min-Heap)

$3n + o(n)$ bits suffice to support RMQ, PSV and NSVs in $\mathcal{O}(1)$ time.

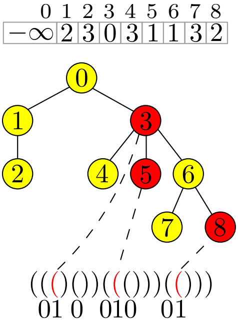


More Functionality: NSV

- Distinguish $=$- and $<$-siblings?

$\rightsquigarrow$ Mark $<$-children in additional bit-vector

- Bit-tricks for $\mathcal{O}(1)$-computations



Theorem (Extended 2d-Min-Heap)

$3n + o(n)$ bits suffice to support RMQ, PSV and NSVs in $\mathcal{O}(1)$ time.

- Not necessarily optimal…
- … $\leq 2.54\ldots n$ possible (Schröder Tree!)