Binomial (min) Heaps

([W] Section 6.8. Binomial Queues are similar to Binomial Heaps)

Binomial Heaps allow for efficient merging.

Binomial (min) Heap: Sequence of Binomial Trees (that satisfy the min heap property) of different order.

Binomial Tree of order $k$ ($k\geq 0$)

- $k=0$: The tree is a single node
- $k>0$: The tree has a root whose children are roots of binomial trees of orders $k-1$, $k-2$, ..., 1, 0.
Binomial Trees

$k = 0$

$k = 1$

$k = 2$

$k = 3$

$k = 4$
Binomial Heap

```
0 1 2 3 4 5 ...

0 1 2 3 3

min heap property
```
A binomial tree of order \( k \) has \( 2^k \) nodes and height \( k \).

**Implementation**

**Merge**

Merge binomial trees of the same order:

\[
\begin{array}{c}
3 \\
5
\end{array} \implies
\begin{array}{c}
3 \\
5
\end{array}
\]
MERGE: \( BH_1 + BH_2 \rightarrow BH \)

\[ \text{CARRY} = \text{Null} \]

for \( i = 0, \ldots, \text{Size of array} \)

let \( j \) be the number of not null pointers in \( BH_1[i], BH_2[i] \) & \( \text{CARRY} \)

**CASE**

: \( j = 3 \): merge \( \text{CARRY} + BH_1[i] \) into \( \text{CARRY} \)

\[ BH[i] \leftarrow BH_2[i] \]

: \( j = 2 \): merge both trees into \( \text{CARRY} \)

\[ BH[i] \leftarrow \text{Null} \]

: \( j = 1 \): let \( BH[i] \) get the non-null pointer & set \( \text{CARRY} \leftarrow \text{Null} \)

: \( i = 0 \): \( BH[i] \leftarrow \text{Null} \)

end for

if \( \text{CARRY} \neq \text{Null} \) then error
INSERT x in BH,

Create Binomial Heap BH2 with only x

MERGE BH1 & BH2

Find Min BH

min at
Delete Min \[ \text{BH}\]

First locate Min

\[\text{BH}_1\]

\[\text{BH}_2\]

\[\text{BH}_3\]

\[\downarrow\]

Merge \[\text{BH}_1, \text{BH}_2\]
Delete $x$ from BH,

Locate $x$

Change value to $-\infty$

BH,

Delete $\uparrow$

OP like Delete min