**Interval Trees**

Set $S$ of intervals:

4, 7, 8, 10, 11, 15, 18, 19, 21, 23

$x$ is an interval in the set $S$ such that $lo(x) = a$ and $hi(x) = b$.

Query: given an interval $x$, return any interval in the set $S$ that partially overlaps $x$ (if one exists).
types of overlap:
1) "smaller"
2) "bigger"
3) "left" & "right"

First comparison: $lo[s_i]$ vs $lo[x]$
types of overlap:
1) "smaller"
2) "bigger"
3) "left" & "right"

First comparison: $lo[s_i]$ vs $lo[x]$

is there some large enough $hi[s_i]$?

If $lo[s_i] < lo[x]$ AND $hi[s_i] > lo[x]$
then overlap
types of overlap:
1) "smaller"
2) "bigger"
3) "left" & "right"

First comparison: $lo[s_i]$ vs $lo[x]$
- is there some large enough $hi[s_i]$?
- is there some small enough $lo[s_i]$?

If $lo[s_i] < lo[x]$
  AND
  $hi[s_i] > hi[x]$
  then overlap

If $lo[s_i] > lo[x]$
  AND
  $hi[s_i] \leq hi[x]$
  then overlap
SEARCHING FOR OVERLAPPING INTERVALS

BST w/ LEFT ENDS as KEYS

MAX RIGHT END OF SUBTREE
SEARCHING FOR OVERLAPPING INTERVALS

CASE 1

IF NO OVERLAP

right subtree can't overlap

keep searching LEFT

R < x < w
SEARCHING FOR OVERLAPPING INTERVALS

1D:

IF $Z \geq L$

search left

IF NO OVERLAP

$y < l < z'$

$z'$

$z$

case 2

L

R

guaranteed overlap

$y$
SEARCHING FOR OVERLAPPING INTERVALS

1D:

IF \( z \geq L \):
- search left

\[ \exists y_2' \text{ s.t. } y < l < z' \]
- \( \{ \text{guaranteed overlap} \) \]

IF NO OVERLAP
- case 2

\( L \rightarrow z \rightarrow R \)

ELSE (\( z < L \))
- no overlap to left
- search right
augmented BST

max(t) = max \left\{ h_i(t), \max(t_{L}), \max(t_{R}) \right\}
\( \text{max}_1, \text{max}_2, \text{max}_3 \) : unchanged by rotation

\( \text{max}(A) \ & \ \text{max}(B) \) : trivial to update

we can maintain a balanced BST augmented w/ max value of subtrees