Red-Black Trees

A red-black tree is another balanced BST, faster than AVL's in practice.

- **Color property**: Every node is either red or black.
- **Root property**: The root node is black.
- **Internal property**: The children of a red node are black.
- **Depth property**: For each node v, all v → null pointer paths contain the same number of black nodes.
- **Nil pointer path property**: The children of a red node are black.
- **Binary search tree property**: A red-black tree is a binary search tree that is almost balanced in practice.

Structural Induction

Structural induction is a proof technique used to prove boolean properties. That is, it's used to prove that a given boolean method always returns true.

To use structural induction, you need:

- A recursively defined structure S with both
  - A base case, and
  - A recursive case

To prove that P(S) returns true, you need:

- To prove that P(S) is true when S is one of the base cases, and
- To prove that P(S) is true when S is a recursively defined structure S with both
  - A recursively defined structure S with both
    - A base case, and
    - A recursive case

To use structural induction, you need:

- A recursively defined structure S with both
  - A base case, and
  - A recursive case

To prove that P(S) returns true, you need:

- To prove that P(S) is true when S is one of the base cases, and
- To prove that P(S) is true when S is a recursively defined structure S with both
  - A recursively defined structure S with both
    - A base case, and
    - A recursive case
A red-black tree \( T \) is either

- an empty tree, or
- a black root node \( r \) with left and right RB trees that may violate the root property (so root must be black if either child is red, otherwise no restrict'n).

For a RB tree \( T \), let \( b(T) \) denote the number of black nodes in \( T \), and let \( m(T) \) denote the number of black nodes along any root-leaf path.

For a RB tree \( T \) that possibly violates the root property, let \( P(T) \) be the boolean property that

\[
b(T) \geq 2m(T) - 1
\]

This fact helps show that RB trees have height \( O(\log n) \). (It isn't immediately obvious, but it helps nonetheless.)

We will show that \( P(T) \) is true for all RB trees \( T \) by structural induction on \( T \).

**Step 0:** Play around and convince yourself \( P(T) \) is true for some specific \( T \)’s.

We will show that \( P(T) \) is true for all RB trees \( T \) “by structural induction” on \( T \).

**Base case:** \( T \) is an empty tree

We want to show (WTS) that \( P(T) \) is true when \( T \) is an empty tree, i.e. that

\[
b(T) \geq 2m(T) - 1
\]

Step 2: Show that \( P(T) \) is true when \( T \) is your recursive case.

We will show that \( P(T) \) is true for all RB trees \( T \) “by structural induction” on \( T \).

**Inductive step:** \( T \) is a root \( r \) with subtrees \( L \) and \( R \) that may violate root prop.

Suppose, hypothetically, that \( P(L) \) and \( P(R) \) are true, i.e. that

\[
b(L) \geq 2m(L) - 1 \quad \text{and} \quad b(R) \geq 2m(R) - 1
\]

We want to use these two hypothetical assumptions to show \( P(T) \) is true, i.e. that

\[
b(T) \geq 2m(T) - 1
\]

We will show that \( P(T) \) is true for all RB trees \( T \) “by structural induction” on \( T \).

We will show that \( P(T) \) is true for all RB trees \( T \) “by structural induction” on \( T \).

**Base case:** \( T \) is an empty tree

We want to show (WTS) that \( P(T) \) is true when \( T \) is an empty tree, i.e. that

\[
b(T) \geq 2m(T) - 1
\]

Step 1: Show that \( P(T) \) is true when \( T \) is your base case.

We will show that \( P(T) \) is true for all RB trees \( T \) “by structural induction” on \( T \).

**Inductive step:** \( T \) is a root \( r \) with subtrees \( L \) and \( R \) that may violate root prop.

Suppose, hypothetically, that \( P(L) \) and \( P(R) \) are true, i.e. that

\[
b(L) \geq 2m(L) - 1 \quad \text{and} \quad b(R) \geq 2m(R) - 1
\]

We want to use these two hypothetical assumptions to show \( P(T) \) is true, i.e. that

\[
b(T) \geq 2m(T) - 1
\]

We will show that \( P(T) \) is true for all RB trees \( T \) “by structural induction” on \( T \).

We will show that \( P(T) \) is true for all RB trees \( T \) “by structural induction” on \( T \).
For a RB tree $T$, let $b(T)$ denote the number of black nodes in $T$, and let $m(T)$ denote the number of black nodes along any root-leaf path.

$$b(L) \geq 2m(L) - 1$$

$$b(R) \geq 2m(R) - 1$$

Recall that we want to show that $b(T) \geq 2m(T) - 1$ given our IH that $b(T) \geq 2m(T) - 1$.

• Algebraically connect $b(T)$ to $b(L), b(R)$
• Algebraically connect $m(T)$ to $m(L), m(R)$

Let $x(T) = 1$ if the root of $T$ is black, and let $x(T) = 0$ if it is red. Then

$b(T) = b(L) + b(R) + x(T)$ (because $T$ contains black nodes of $R$ and $L$)

$m(T) = m(L) + x(T) = m(R) + x(T)$ (because of the RB tree depth property)

$b(T) = b(L) + b(R) + x(T) = b(L) + b(R) + x(T)$

because $T$ contains nodes of $R$, $L$, and itself

$b(T) = b(L) + b(R) + x(T)$

by our induction hypothesis (IH)

$\geq 2m(L) - 1 + 2m(R) - 1 + x(T)$

by algebra

Now there are 2 cases. If $x(T) = 0$ then

$b(T) \geq 2m(T) - 1 + 2m(T) - 1$ = $2m(T) - 1 + 2m(T)$

by algebra

$= 2m(T) - 1 + 2m(T) - 1 + x(T)$

$= 2m(T) - 1 + 2m(T) - 1 + x(T)$

$= 2m(T) - 1 + 2m(T) - 1 + x(T)$

$= 2m(T) - 1 + 2m(T) - 1 + x(T)$

$= 2m(T) - 1 + 2m(T) - 1 + x(T)$

$= 2m(T) - 1 + 2m(T) - 1 + x(T)$

$= 2m(T) - 1 + 2m(T) - 1 + x(T)$

$= 2m(T) - 1 + 2m(T) - 1 + x(T)$

$= 2m(T) - 1 + 2m(T) - 1 + x(T)$

$= 2m(T) - 1 + 2m(T) - 1 + x(T)$

$= 2m(T) - 1 + 2m(T) - 1 + x(T)$

$= 2m(T) - 1 + 2m(T) - 1 + x(T)$

$= 2m(T) - 1 + 2m(T) - 1 + x(T)$

$= 2m(T) - 1 + 2m(T) - 1 + x(T)$

$= 2m(T) - 1 + 2m(T) - 1 + x(T)$

$= 2m(T) - 1 + 2m(T) - 1 + x(T)$

$= 2m(T) - 1 + 2m(T) - 1 + x(T)$

$= 2m(T) - 1 + 2m(T) - 1 + x(T)$

$= 2m(T) - 1 + 2m(T) - 1 + x(T)$

$= 2m(T) - 1 + 2m(T) - 1 + x(T)$

$= 2m(T) - 1 + 2m(T) - 1 + x(T)$

$= 2m(T) - 1 + 2m(T) - 1 + x(T)$

$= 2m(T) - 1 + 2m(T) - 1 + x(T)$

$= 2m(T) - 1 + 2m(T) - 1 + x(T)$

$= 2m(T) - 1 + 2m(T) - 1 + x(T)$

$= 2m(T) - 1 + 2m(T) - 1 + x(T)$

$= 2m(T) - 1 + 2m(T) - 1 + x(T)$

$= 2m(T) - 1 + 2m(T) - 1 + x(T)$

$= 2m(T) - 1 + 2m(T) - 1 + x(T)$

$= 2m(T) - 1 + 2m(T) - 1 + x(T)$

$= 2m(T) - 1 + 2m(T) - 1 + x(T)$

$= 2m(T) - 1 + 2m(T) - 1 + x(T)$

$= 2m(T) - 1 + 2m(T) - 1 + x(T)$

$= 2m(T) - 1 + 2m(T) - 1 + x(T)$

$= 2m(T) - 1 + 2m(T) - 1 + x(T)$

$= 2m(T) - 1 + 2m(T) - 1 + x(T)$

$= 2m(T) - 1 + 2m(T) - 1 + x(T)$

$= 2m(T) - 1 + 2m(T) - 1 + x(T)$

$= 2m(T) - 1 + 2m(T) - 1 + x(T)$

$= 2m(T) - 1 + 2m(T) - 1 + x(T)$

In both cases, we've shown our inequality holds. The proof is done!