



SPLAY Tree

Pittsford Sutherland Programming Club

By Yiyou Chen, Yizuo Chen




What is a Splay tree?

- Splay tree is another type of balanced tree which is similar to AVL tree and Treap.
- Splay tree is a binary search tree. It support search, insert, delete, and more advanced operations.
- All splay tree operations run in $O(\log n)$ time on average.
- Splay tree also uses rotations to adjust.
- Splay tree is the most widely used balanced tree.



How does Splay work?

- Instead of basing on the height of the tree like an AVL tree does, Splay tree puts the most recently visited node as its root. In another word, every time after we ask or visit a node, we make the node as the whole tree's root by using rotations.
 - Why?
 - Because it's more likely that we visit the nodes we've recently visited. If we put the node to the top of the tree, it would be faster to retrieve it again.
- 



Declarations before start

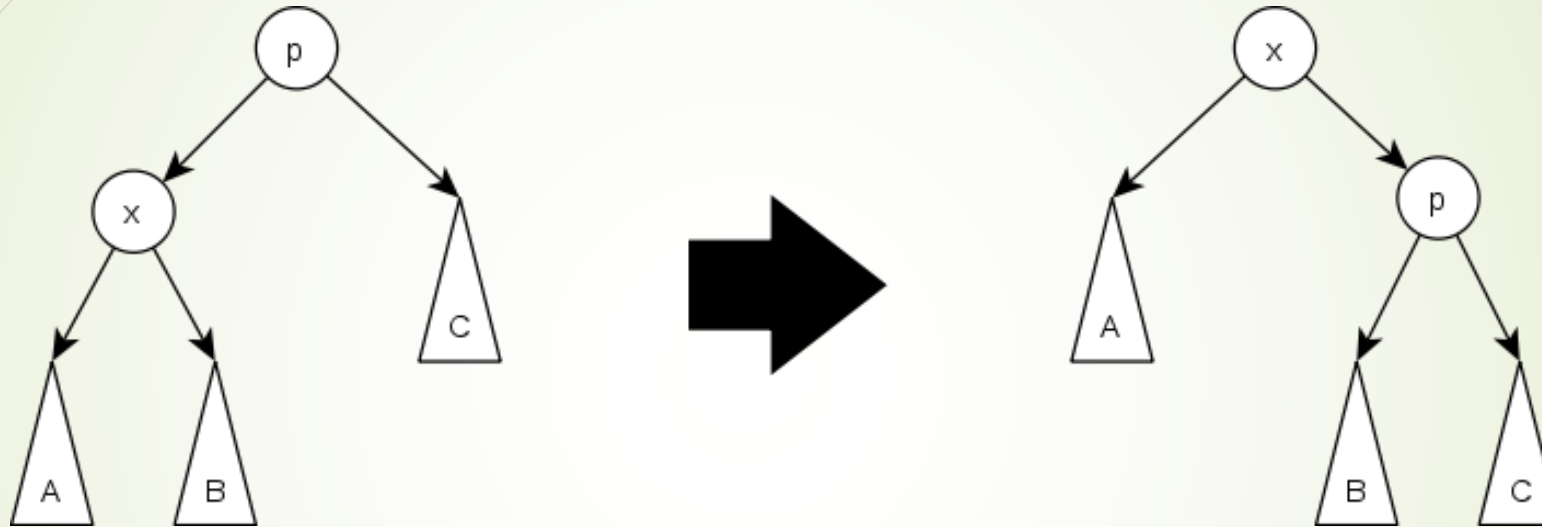
- In order to make the operations more clear. We are using x for the node that we are visiting, and it's also the node that we want to put as the root of the tree.



Rotations of a Splay tree

- The rotations of a splay tree is similar to which of an AVL tree.
- The rotations include:
 - Zig
 - Zag
 - Zig-zig
 - Zig-zag
 - Zag-zag
 - Zag-zig
- For Zig-zig and zag-zag all the things are same except the rotation direction.
- For zig-zag and zag-zig only the rotation order is different.

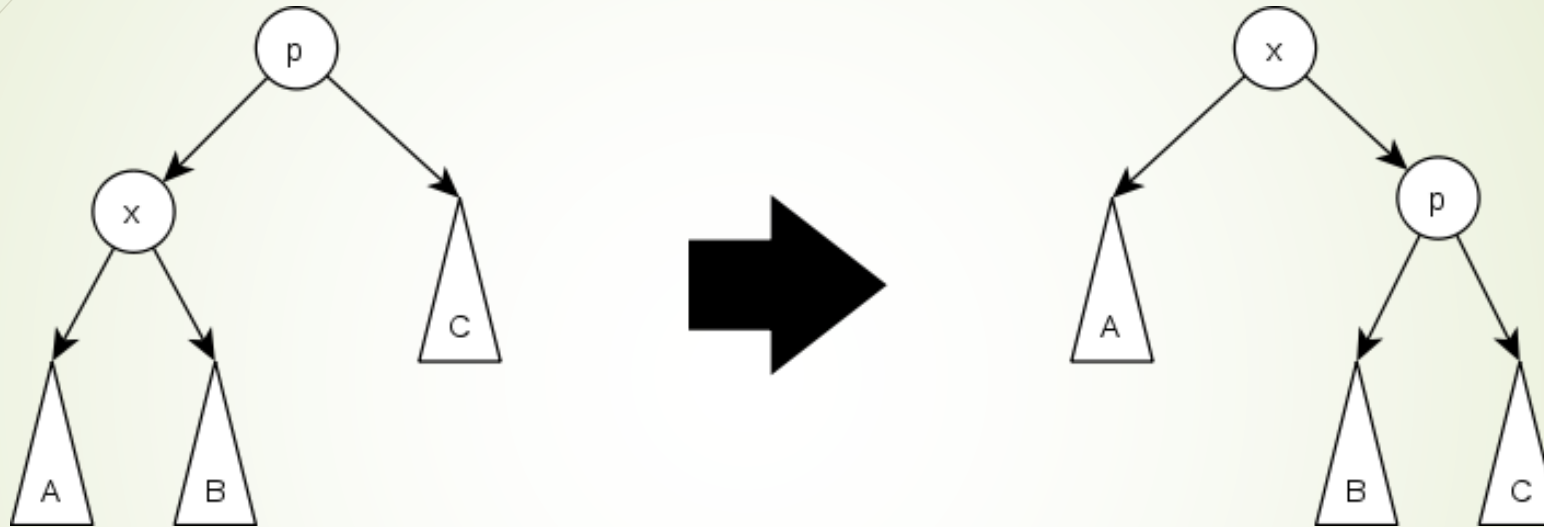
Zig rotation(single-right rotation)



Situation: x is the left child of the root of the whole tree.

Zig rotation is a single step rotation that rotate a single node right to the position of the root.

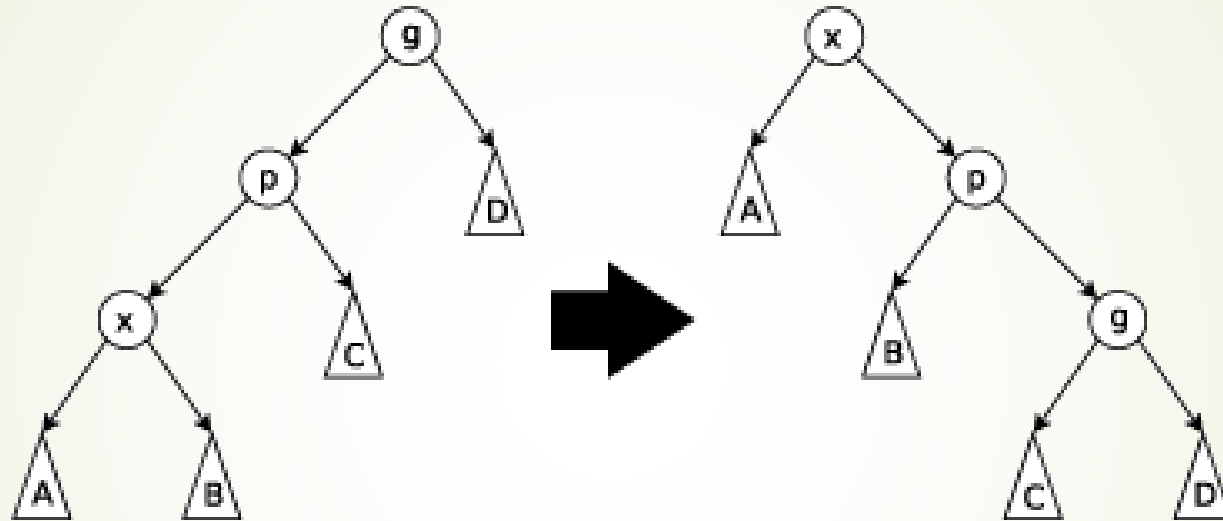
Zag rotation(single-left rotation)



Situation: x is the right child of the root of the whole tree.

Zag rotation is a single step rotation that rotate a single node left to the position of the root.

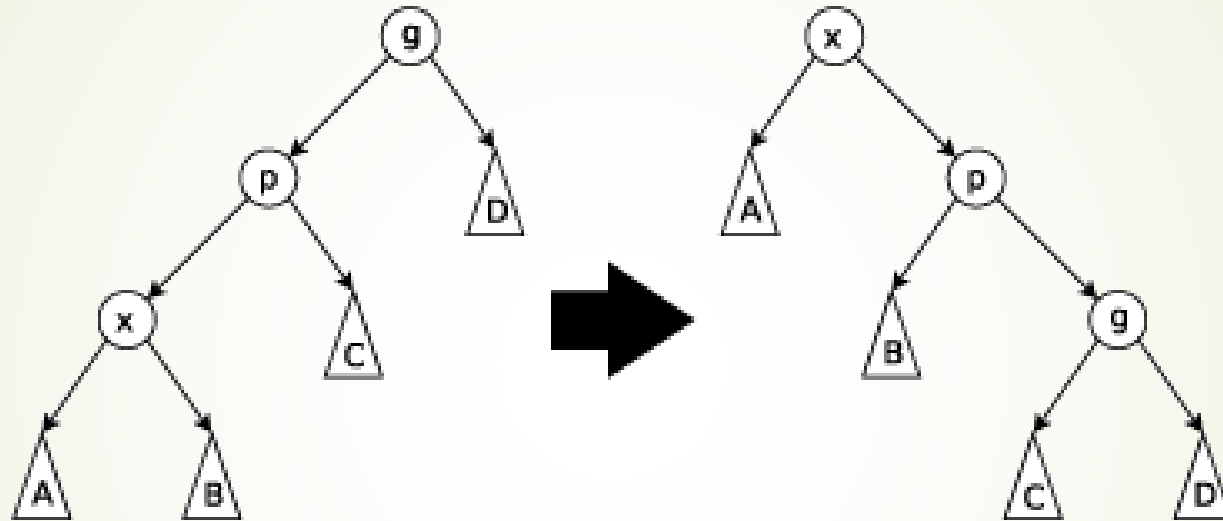
Zig-Zig rotation(double-right rotation)



Situation: x is not the child of the root of the whole tree; It as parent and grandparent, and x is the left child of its parent, which is also the left child of x's grandparent.

Zig-Zig rotation is the combination of two separate zig(single-right) rotations.

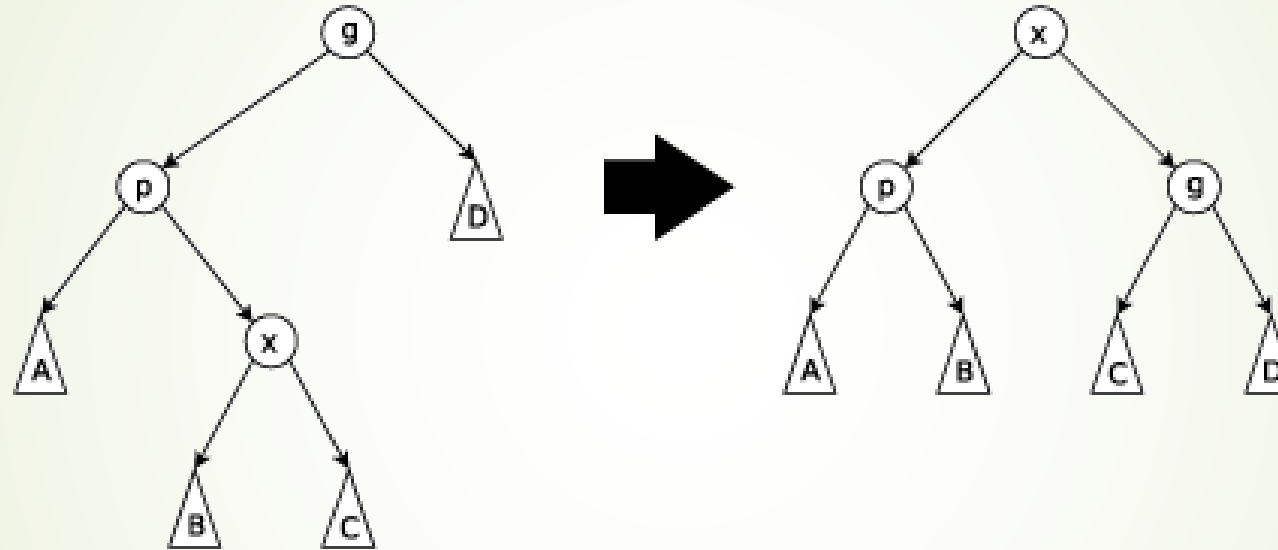
Zag-Zag rotation(double-left rotation)



Situation: x is not the child of the root of the whole tree; It as parent and grandparent, and x is the right child of its parent, which is also the right child of x's grandparent.

Zag-Zag rotation is the combination of two separate zag(single-left) rotations.

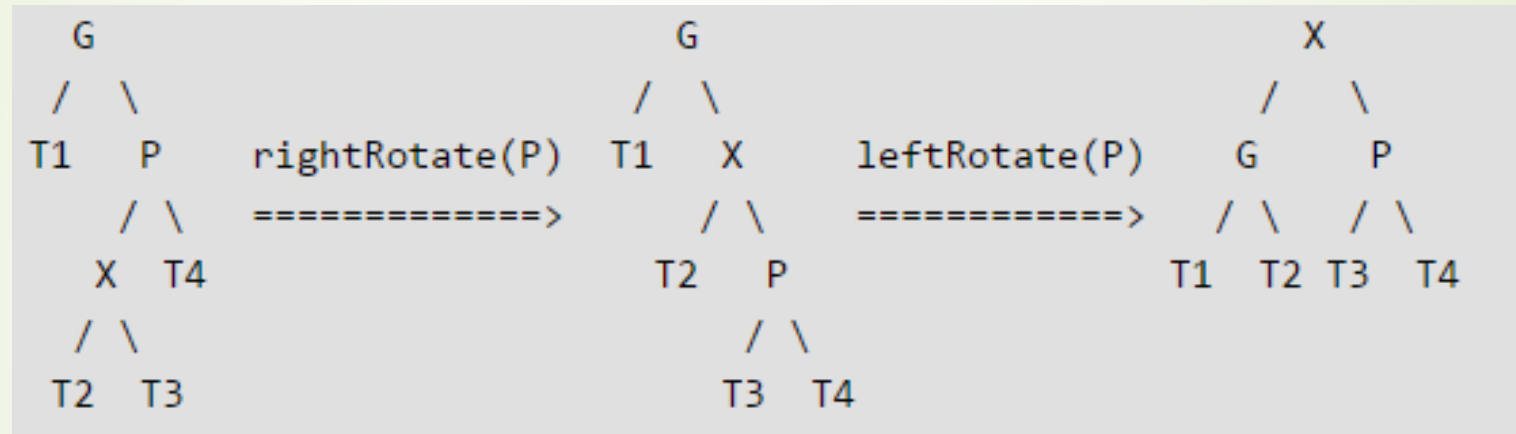
Zig-Zag rotation(left-right rotation)



Situation: x has a parent and a grandparent. X is the right child of its parent, which is the left child of x's grandparent.

Zig-Zag is the combination of two separate single rotations: left rotation and right rotation.

Zag-Zig rotation(right-left rotation)



Situation: x has a parent and a grandparent. X is the left child of its parent, which is the right child of x's grandparent.

Zig-Zag is the combination of two separate single rotations: right rotation and left rotation.



Coding



- Since Splay tree has many kinds of rotations and operations, it's harder to code compared to easier data structures like queues and stacks.
- However, it's such an important data structure that most people practice a lot in order to write the code correctly and quickly.
- If you are interested in writing a splay tree including inserting, searching, and deleting operations, the code below might be helpful to you.