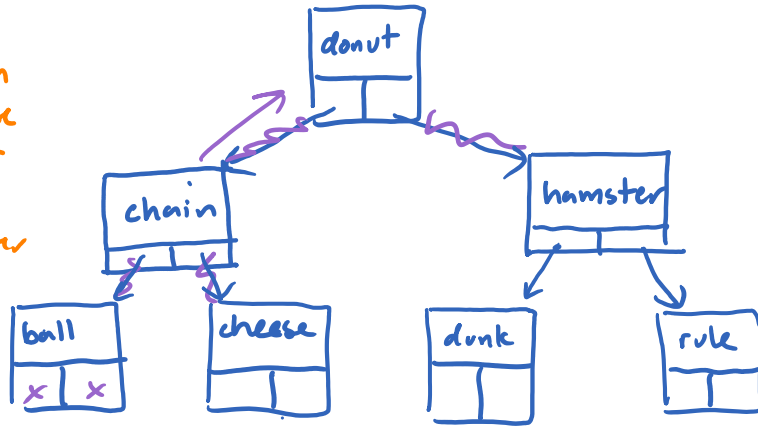


# Traversing a BST

	1	2	3
donut	.	.	.
→ chain	.	.	.
→ ball	.	.	.
→ cheese	.	.	.
hamster	.	.	.
dunk	.	.	.

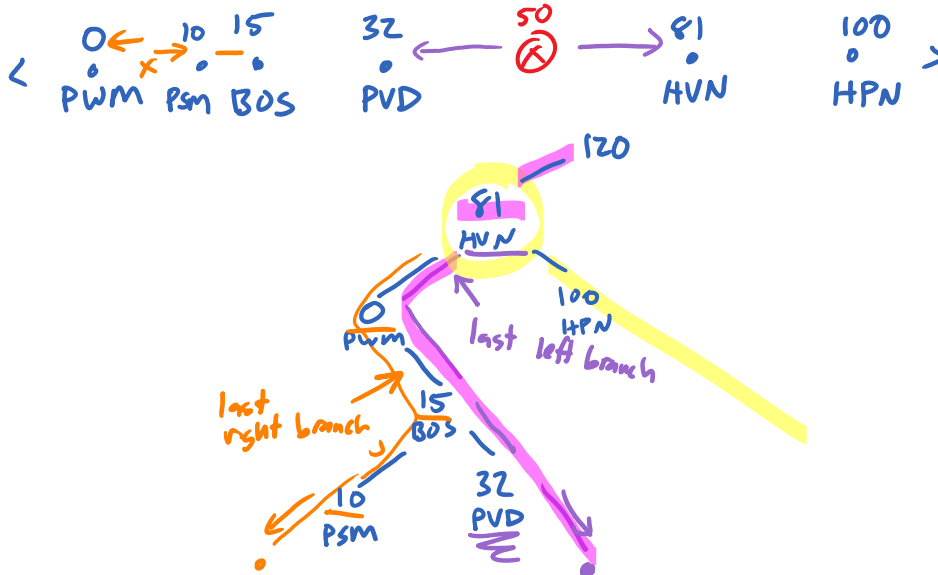
ball  
chain  
cheese  
donut  
dunk  
hamster  
rule



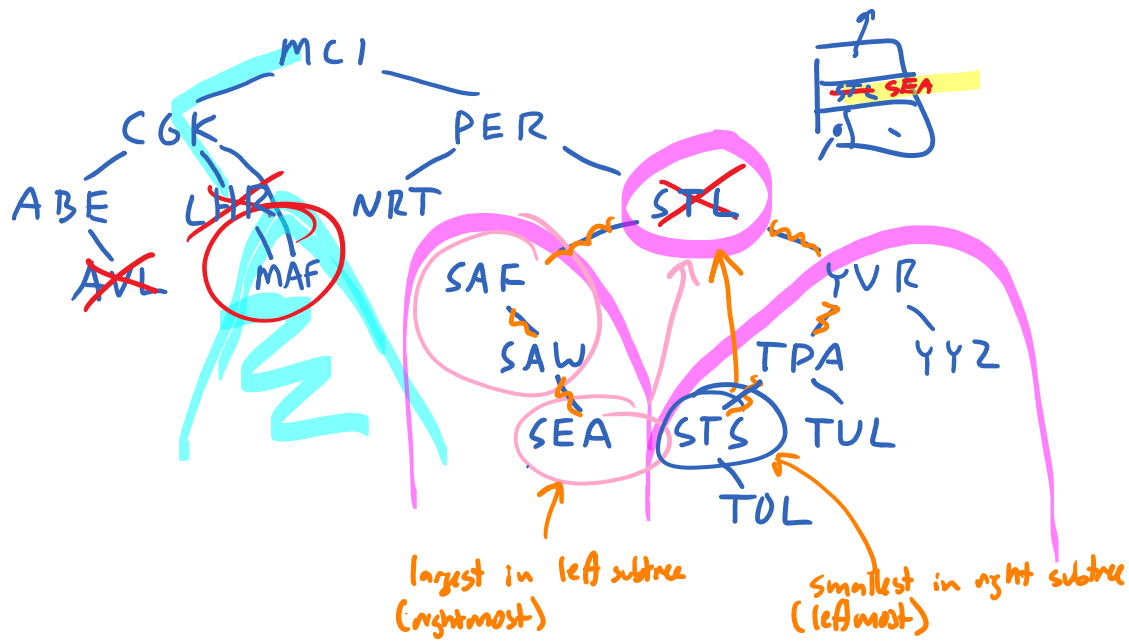
```
void smap_for_each_helper(smap_node *curr, void (*f)(const char *, void *, void *), void *arg)
{
    if (curr != NULL)
    {
        < → smap_for_each_helper(curr->left, f, arg); // 1
        > → f(curr->key, curr->value, arg); // 2
        > → smap_for_each_helper(curr->right, f, arg); // 3
    }
}
```

$O(n)$  time

tree sort  
add items to balanced BST  
add to output array during in-order traversal

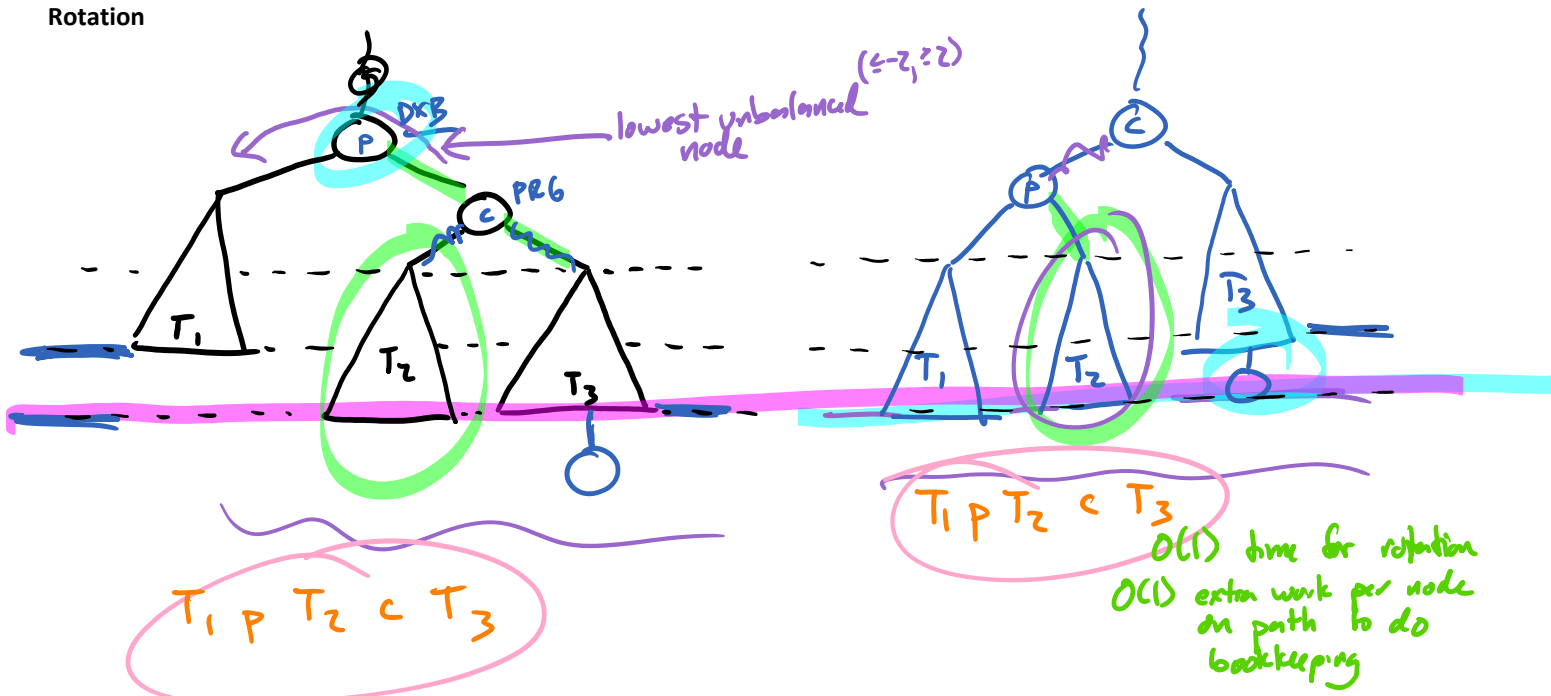


# Removing from a BST

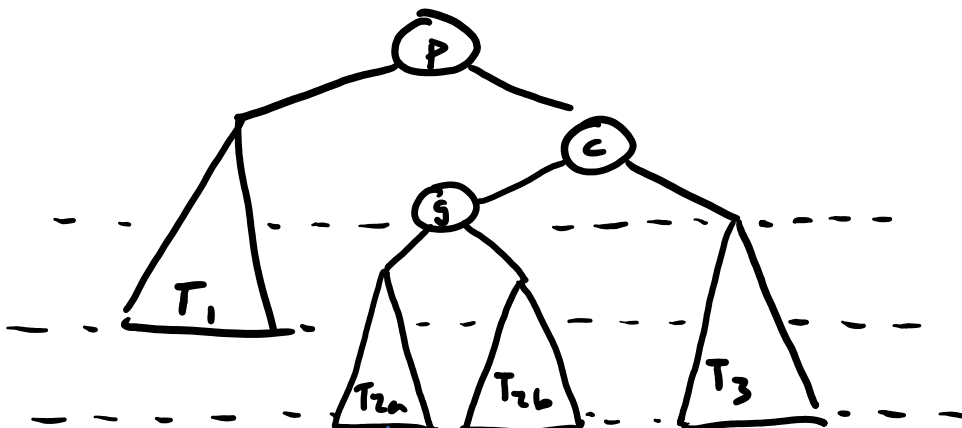
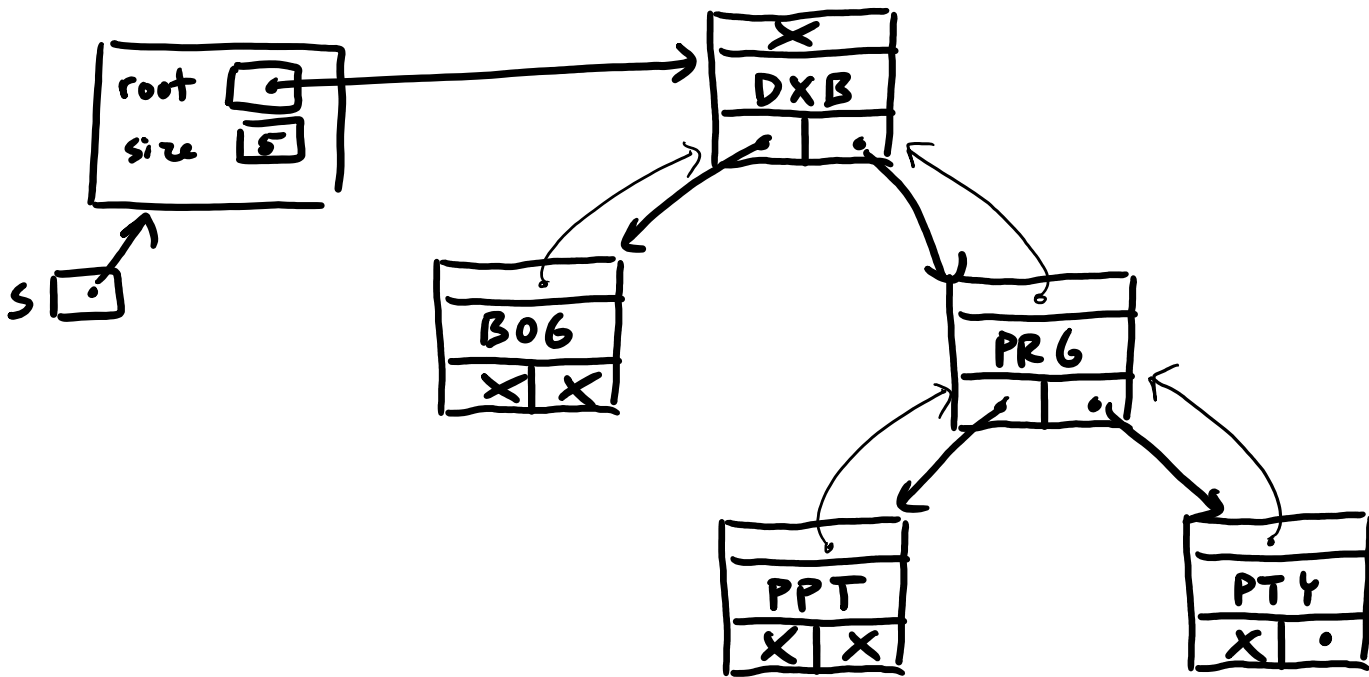
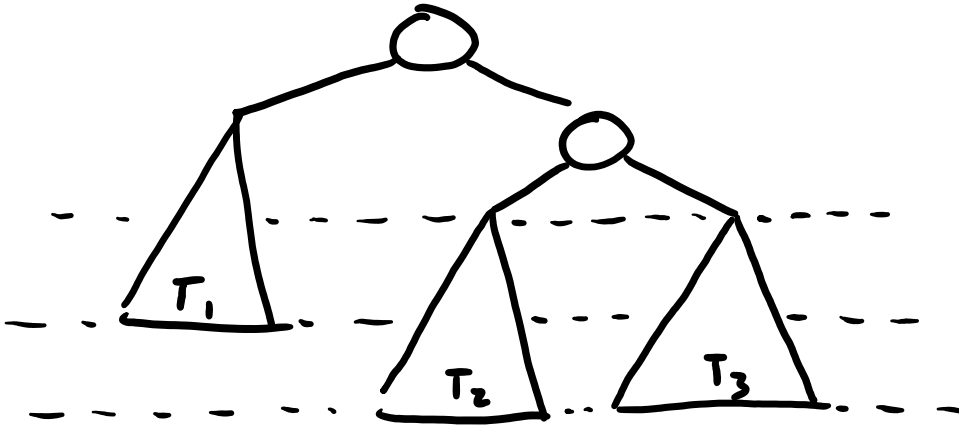


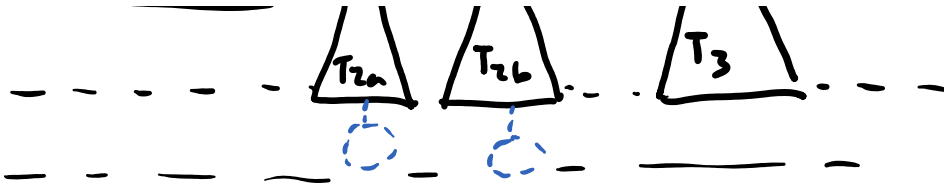


Rotation

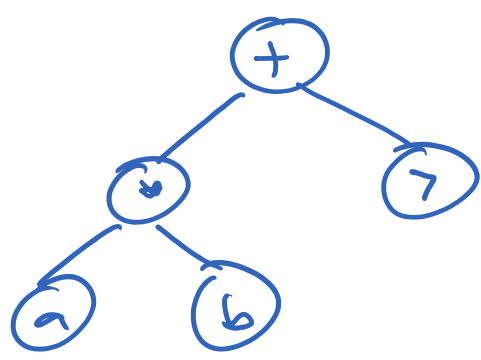
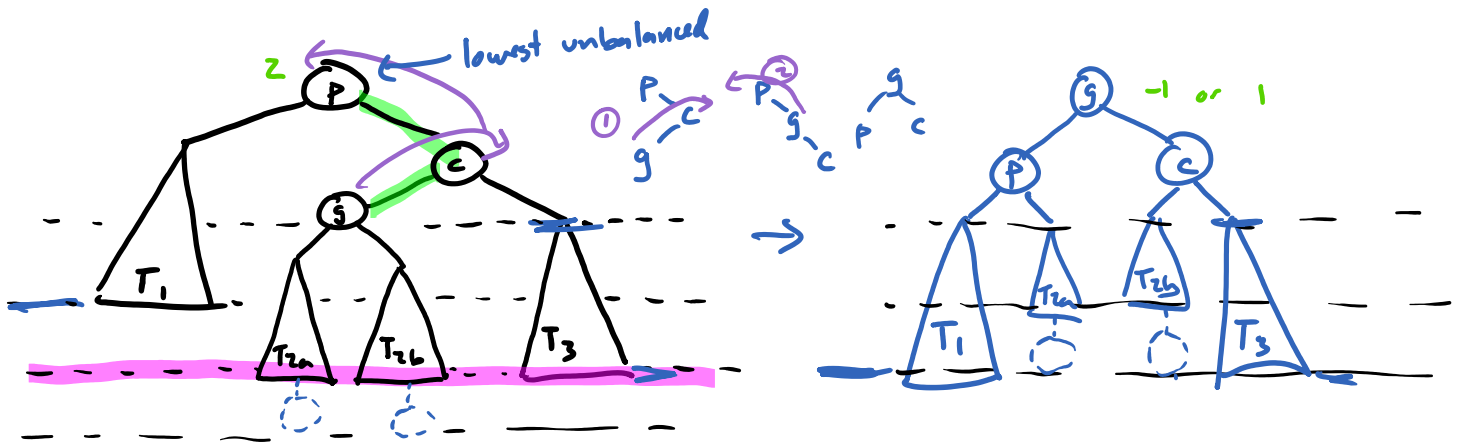
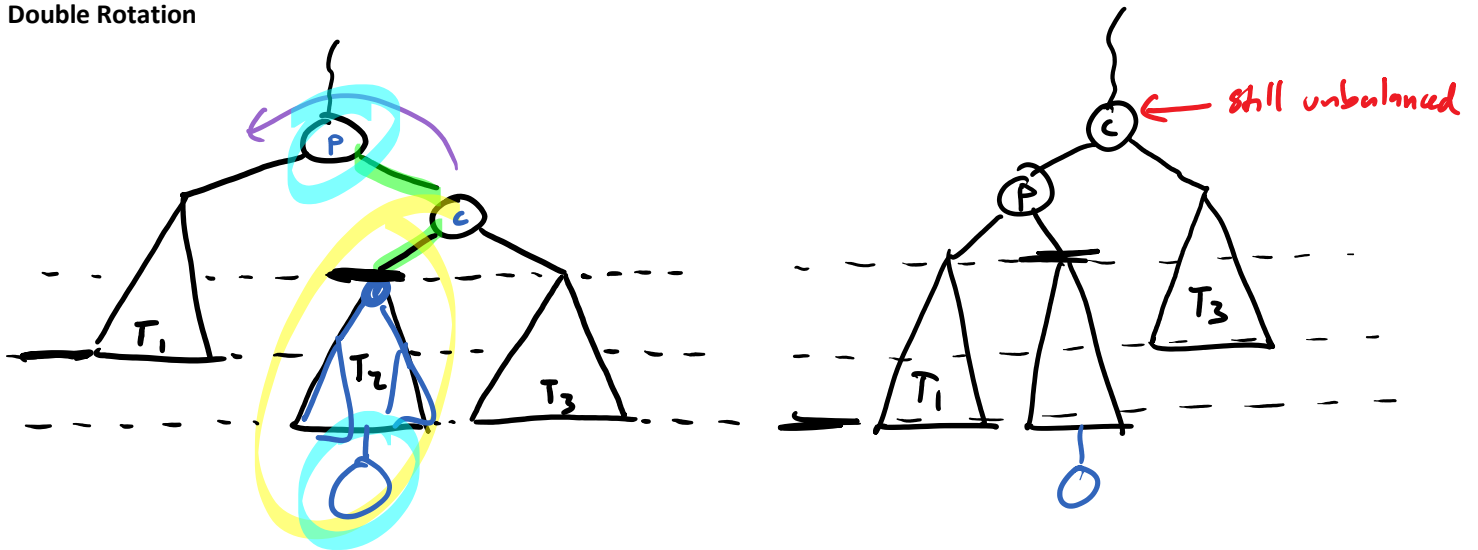


# Rotations





Double Rotation

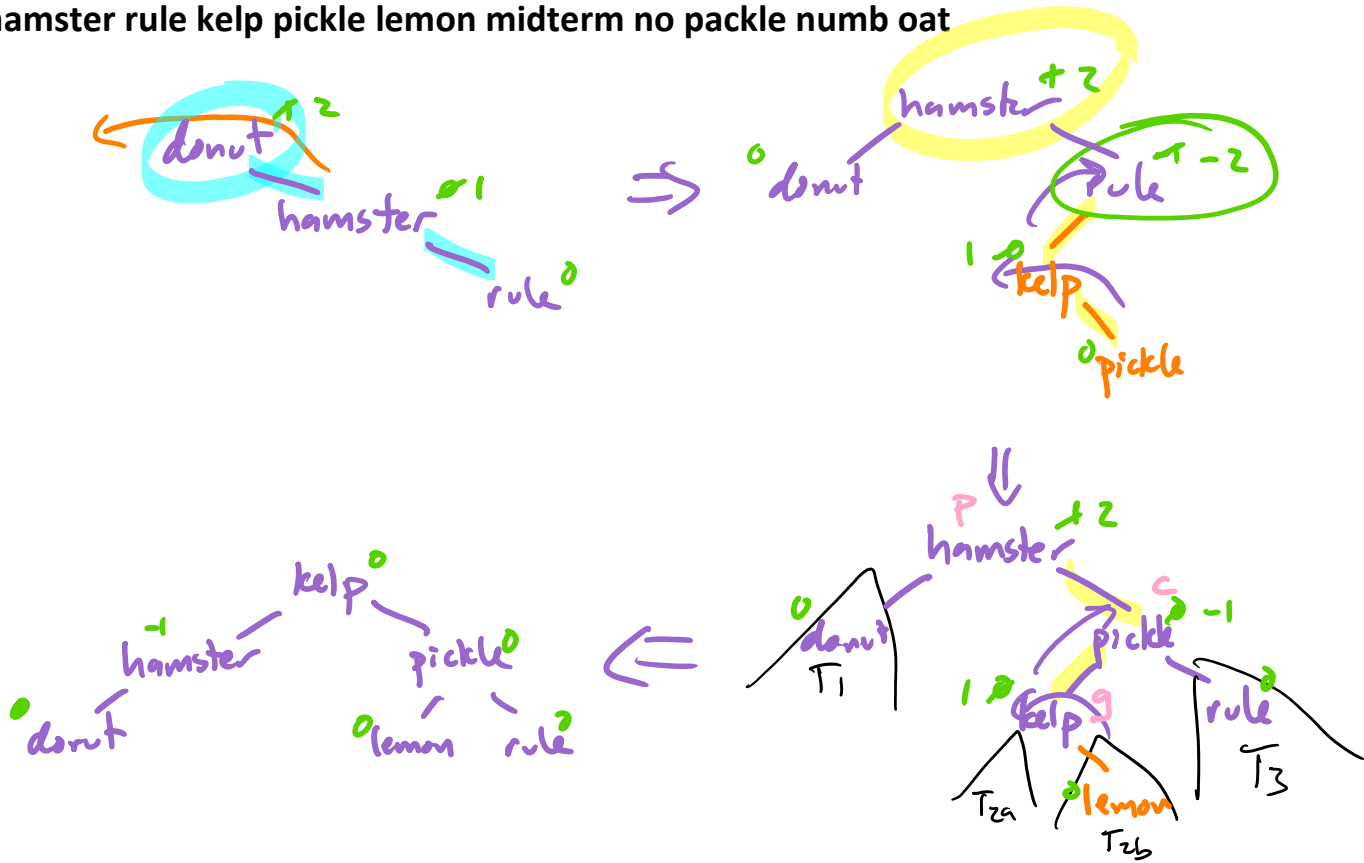


$a \cdot b + 7$

$(+ (* a b) 7)$

# Example

donut hamster rule kelp pickle lemon midterm no packle numb oat





Remove

