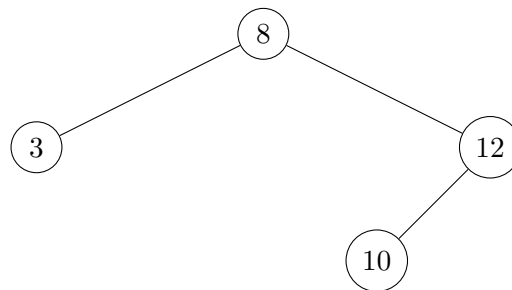


Algorithms and Data Structures Summer Term 2019 Sample Solution Exercise Sheet 7

Exercise 1: Binary Search Trees I

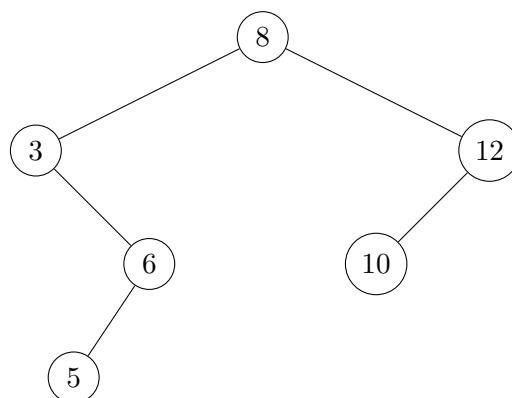
Consider the following binary search tree



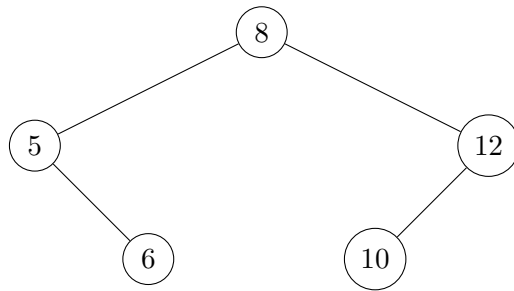
1. Give all sequences of `insert(key)` operations that generate the tree.
2. Draw the tree after the following sequence of operations: `insert(6)`, `insert(5)`, `remove(3)`.

Sample Solution

1. (i) `insert(8),insert(3), insert(12), insert(10)`
(ii) `insert(8),insert(12), insert(3), insert(10)`
(iii) `insert(8),insert(12), insert(10), insert(3)`
2. After `insert(6)` and `insert(5)`:



After `remove(3)`:



Exercise 2: Binary Search Trees II

Write an algorithm that takes as input a node of a binary search tree and outputs its successor in the tree (i.e., the node with the next larger key). What is the runtime of the algorithm?

Sample Solution

```
if  $u.right \neq \text{None}$  then
    current =  $u.right$ 
    while  $current.left \neq \text{None}$  do
        current =  $current.left$ 
    return current
else
    current =  $u$ 
    parent =  $current.parent$ 
    while  $parent \neq \text{None}$  and  $parent.left \neq current$  do
        current =  $parent$ 
        parent =  $current.parent$ 
    return parent
```

Runtime: $\mathcal{O}(\text{depth})$