Binary Search Tree

Problem: Find All Duplicates in an Array

Given an integer array nums of length n where all the integers of nums are in the range [1, n] and each integer appears once or twice, return an array of all the integers that appears twice.

Input	Output
[4,3,2,7,8,2,3,1]	[2,3] or [3, 2]
[1,1,2]	[1]
[1]	[]

https://leetcode.com/problems/find-all-duplicates-in-an-array/description/

Solution 1: Find All Duplicates in an Array

For the current index search all the elements after the current, if the element is found add it into the array and replace it with -1.

Input	Output
[4,3,2,7,8,2,3,1]	[2,3]
[1,1,2]	[1]
[1]	[]

https://leetcode.com/problems/find-all-duplicates-in-an-array/description/

Solution 1: Find All Duplicates in an Array

```
vector<int> findDuplicates(vector<int> &nums)
{
    vector<int> result;
    for (int x = 0; x < nums.size(); x++)
    {
        for (int y = x + 1; y < nums.size(); y++)
        {
            if (nums[x] == nums[y] && nums[y] != -1)
            {
                result.push_back(nums[y]);
                nums[y] = -1;
            }
        }
    }
    return result;
}
```

https://leetcode.com/problems/find-all-duplicates-in-an-array/description/

Solution 1: Time Complexity?

```
vector<int> findDuplicates(vector<int> &nums)
{
    vector<int> result;
    for (int x = 0; x < nums.size(); x++)
    {
        for (int y = x + 1; y < nums.size(); y++)
        {
            if (nums[x] == nums[y] && nums[y] != -1)
            {
                result.push_back(nums[y]);
                nums[y] = -1;
            }
        }
    }
    return result;
}
```

https://leetcode.com/problems/find-all-duplicates-in-an-array/description/

Solution 1: Time Complexity?

Time Complexity $O(n^2)$

```
vector<int> findDuplicates(vector<int> &nums)
{
    vector<int> result;
    for (int x = 0; x < nums.size(); x++)
    {
        for (int y = x + 1; y < nums.size(); y++)
        {
            if (nums[x] == nums[y] && nums[y] != -1)
            {
                result.push_back(nums[y]);
                nums[y] = -1;
            }
        }
    }
    return result;
}
```

https://leetcode.com/problems/find-all-duplicates-in-an-array/description/

Solution 2: Find All Duplicates in an Array

Let's first sort the array and then search if the next element is equal to the current element then add it into the result array.

Input	Output
[4,3,2,7,8,2,3,1]	[2,3]
[1,1,2]	[1]
[1]	[]

https://leetcode.com/problems/find-all-duplicates-in-an-array/description/

Solution 2: Find All Duplicates in an Array

```
vector<int> findDuplicates(vector<int> &nums)
{
    vector<int> result;
    sort(nums.begin(),nums.end());
    for(int x = 0; x < nums.size() - 1; x++)
    {
        if(nums[x] == nums[x+1])
        {
            result.push_back(nums[x]);
            x++;
        }
    }
    return result;
}
```

https://leetcode.com/problems/find-all-duplicates-in-an-array/description/

Solution 2: Time Complexity?

```
vector<int> findDuplicates(vector<int> &nums)
{
    vector<int> result;
    sort(nums.begin(),nums.end());
    for(int x = 0; x < nums.size() - 1; x++)
    {
        if(nums[x] == nums[x+1])
        {
            result.push_back(nums[x]);
            x++;
        }
    }
    return result;
}
```

https://leetcode.com/problems/find-all-duplicates-in-an-array/description/

Solution 2: Time Complexity?

Time Complexity of just Sort Function is O(nlog(n))

```
vector<int> findDuplicates(vector<int> &nums)
{
    vector<int> result;
    sort(nums.begin(),nums.end());
    for(int x = 0; x < nums.size() - 1; x++)
    {
        if(nums[x] == nums[x+1])
        {
            result.push_back(nums[x]);
            x++;
        }
    }
    return result;
}
```

https://leetcode.com/problems/find-all-duplicates-in-an-array/description/

Solution 2: Time Complexity?

Time Complexity of just Sort Function is O(nlog(n))

```
vector<int> findDuplicates(vector<int> &nums)
{
    vector<int> result;
    sort(nums.begin(),nums.end());
    for(int x = 0; x < nums.size() - 1; x++)
    {
        if(nums[x] == nums[x+1])
        {
            result.push_back(nums[x]);
            x++;
        }
    }
    return result;
}
```

Another for loop is used which runs for n times.

https://leetcode.com/problems/find-all-duplicates-in-an-array/description/

Solution 2: Time Complexity?

Time Complexity $O(n^2)$

```
vector<int> findDuplicates(vector<int> &nums)
{
    vector<int> result;
    sort(nums.begin(),nums.end());
    for(int x = 0; x < nums.size() - 1; x++)
    {
        if(nums[x] == nums[x+1])
        {
            result.push_back(nums[x]);
            x++;
        }
    }
    return result;
}
```

https://leetcode.com/problems/find-all-duplicates-in-an-array/description/

Problem: Find All Duplicates in an Array

Given an integer array **nums** of length n where all the integers of **nums** are in the range **[1, n]** and each integer appears **once** or **twice**, return an array of all the integers that **appears twice**.

Input	Output
[4,3,2,7,8,2,3,1]	[2,3]
[1,1,2]	[1]
[1]	[]

Time Complexity should be less than $O(n^2)$.

https://leetcode.com/problems/find-all-duplicates-in-an-array/description/

Problem: Find All Duplicates in an Array

Given an integer array **nums** of length n where all the integers of **nums** are in the range **[1, n]** and each integer appears **once** or **twice**, return an array of all the integers that **appears twice**.

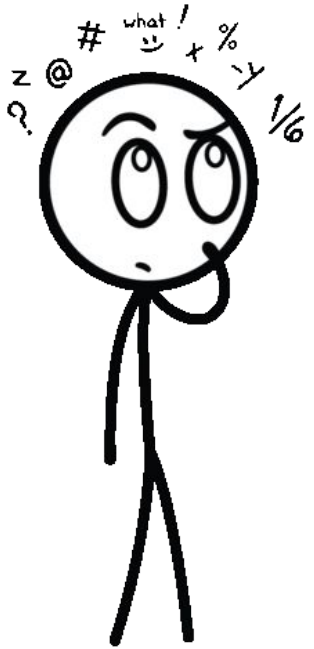
Input	Output
[4,3,2,7,8,2,3,1]	[2,3]
[1,1,2]	[1]
[1]	[]

Time Complexity should be less than $O(n^2)$.

https://leetcode.com/problems/find-all-duplicates-in-an-array/description/

Solution: Find All Duplicates in an Array

What was the last data structure that we studied?



https://leetcode.com/problems/find-all-duplicates-in-an-array/description/

Solution: Find All Duplicates in an Array

Yes..!!! Trees..!!

https://leetcode.com/problems/find-all-duplicates-in-an-array/description/

Solution: Find All Duplicates in an Array

Yes..!!! Trees..!!

Then, of course today's topic would be related to binary trees.

https://leetcode.com/problems/find-all-duplicates-in-an-array/description/

Solution: Find All Duplicates in an Array

Let's consider this test case.

[5,3,2,7,8,2,3,4]

Solution: Find All Duplicates in an Array

Let's consider this test case.

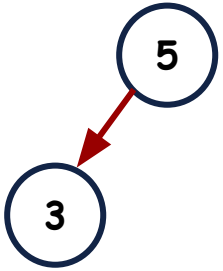
[5,3,2,7,8,2,3,4]

Make the first element the root node

Solution: Find All Duplicates in an Array

Let's consider this test case.

[5,3,2,7,8,2,3,4]

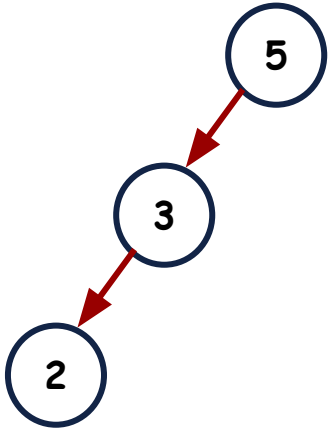


If it is less than the node, place it in the left child of the node.

Solution: Find All Duplicates in an Array

Let's consider this test case.

[5,3,2,7,8,2,3,4]



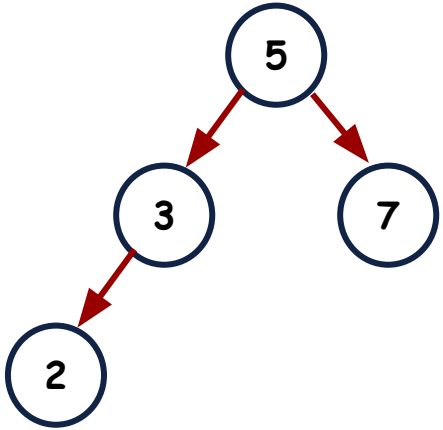
If it is less than the node, go to the left child, if the left of left child is null, and the element is less than the node value then place it in the left child of the node.

https://leetcode.com/problems/find-all-duplicates-in-an-array/description/

Solution: Find All Duplicates in an Array

Let's consider this test case.

[5,3,2,7,8,2,3,4]

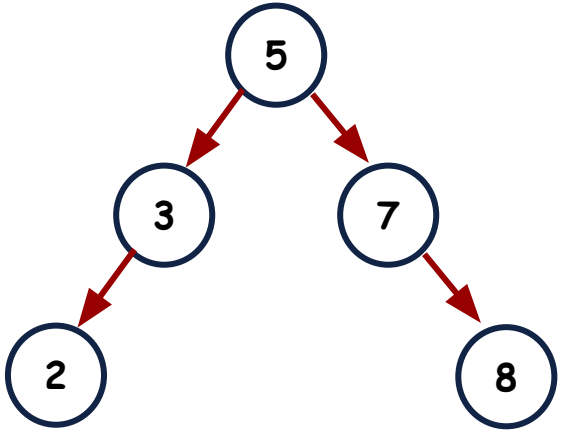


If it is greater than the node, go to the right child, if the right child is null, then place it in the right child of the node.

Solution: Find All Duplicates in an Array

Let's consider this test case.

[5,3,2,7,8,2,3,4]



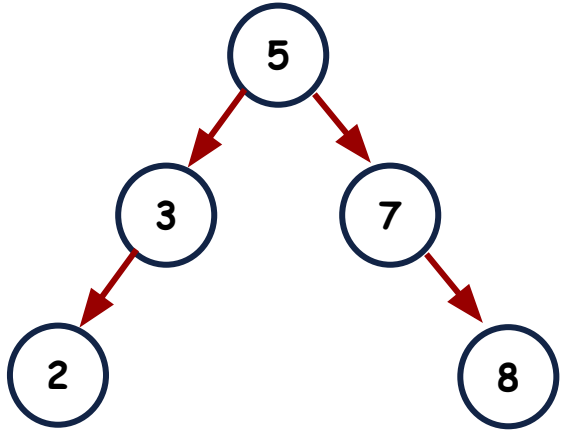
If it is greater than the node, go to the right child, if it is also greater than the right child, then go to the right child of right child, if it is null then place it in the right child of the node.

https://leetcode.com/problems/find-all-duplicates-in-an-array/description/

Solution: Find All Duplicates in an Array

Let's consider this test case.

[5,3,2,7,8,2,3,4]



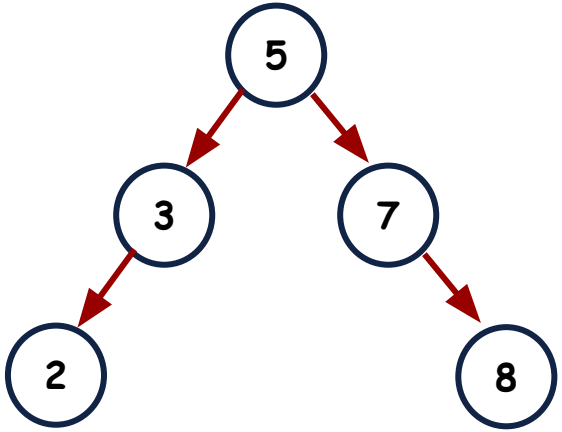
[2]

Let's keep on going to the left if the element is less than the node values, If the value is equal to the node value then store it into the array.

Solution: Find All Duplicates in an Array

Let's consider this test case.

[5,3,2,7,8,2,3,4]



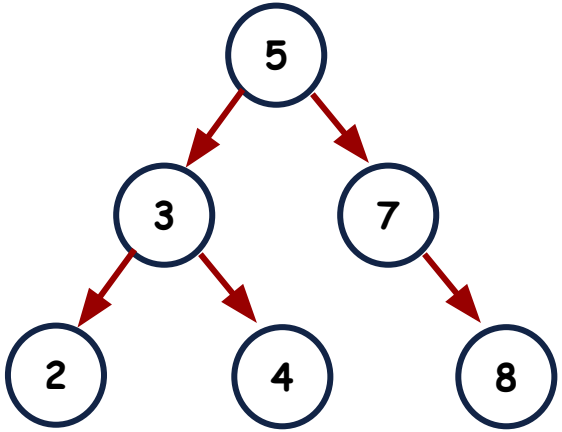
[2, 3]

Let's keep on going to the left if the element is less than the node values, If the value is equal to the node value then store it into the array.

Solution: Find All Duplicates in an Array

Let's consider this test case.

[5,3,2,7,8,2,3,4]



[2, 3]

If the value is less than the node value go to the left, if the value is greater than the node value go to the right, if the value is equal to the node value then store it into the array.

https://leetcode.com/problems/find-all-duplicates-in-an-array/description/

Solution: Find All Duplicates in an Array

```
struct TreeNode
{
    int val;
    TreeNode *left;
    TreeNode *right;
};
```

```
class binaryTree
{
    TreeNode *root;

public:
    TreeNode *createNode(int value)
    {
        TreeNode *record = new TreeNode();
        record->val = value;
        record->left = NULL;
        record->right = NULL;
        return record;
    }
```

https://leetcode.com/problems/find-all-duplicates-in-an-array/description/

Solution:

```
bool insert(TreeNode *node)
    {
        TreeNode *prev = root;
        TreeNode *next = root;
        if (root == NULL)
        {
            root = node;
            return false;
        }
        while (node->val != prev->val && next != NULL)
        {
            prev = next;
            if (node->val < prev->val)
                next = prev->left;
            else
                next = prev->right;
        }
        if (node->val == prev->val)
        {
            delete node;
            return true;
        }
        else if (node->val > prev->val)
            prev->right = node;
        else
            prev->left = node;
        return false;
    }
};
```

https://leetcode.com/problems/find-all-duplicates-in-an-array/description/

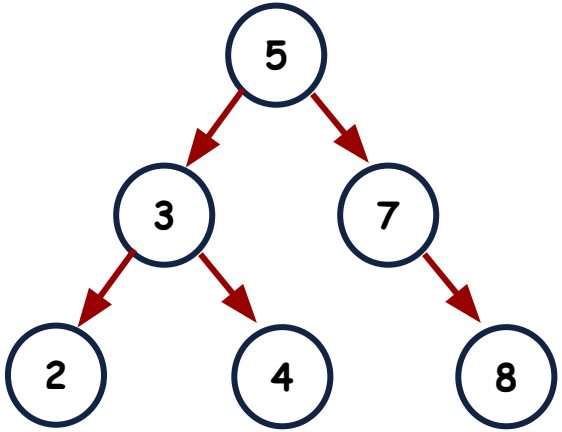
Solution: Find All Duplicates in an Array

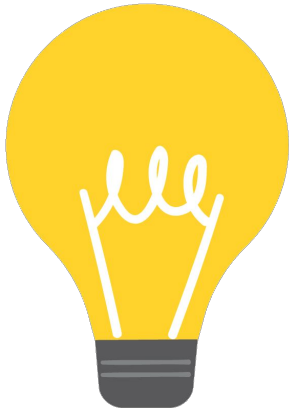
```
vector<int> findDuplicates(vector<int> &nums)
    {
        vector<int> result;
        binaryTree b;
        for (int x = 0; x < nums.size(); x++)
        {
            if (b.insert(b.createNode(nums[x])))
            {
                result.push_back(nums[x]);
            }
        }
        return result;
    }
```

https://leetcode.com/problems/find-all-duplicates-in-an-array/description/

Solution: Find All Duplicates in an Array

What will be the time complexity of this Algorithm?



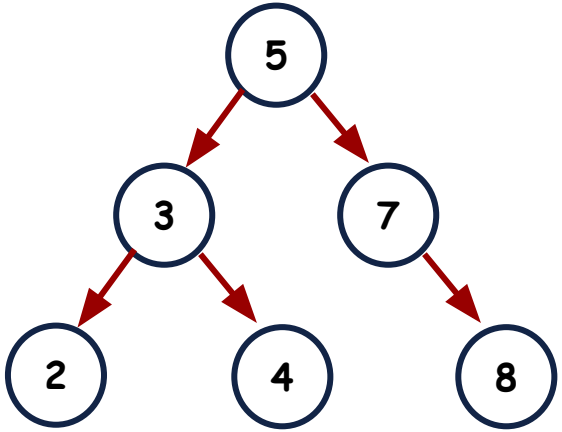


https://leetcode.com/problems/find-all-duplicates-in-an-array/description/

Solution: Find All Duplicates in an Array

What will be the time complexity of this Algorithm?

In order to insert the element In the binary we need h comparisons i.e., the loop in the insert function runs for h times



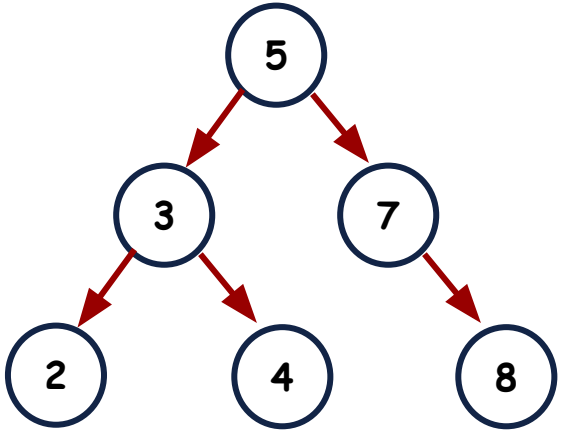
Time Complexity of insert function O(h)

https://leetcode.com/problems/find-all-duplicates-in-an-array/description/

Solution: Find All Duplicates in an Array

What will be the time complexity of this Algorithm?

And this insert is done for all the elements in the input.



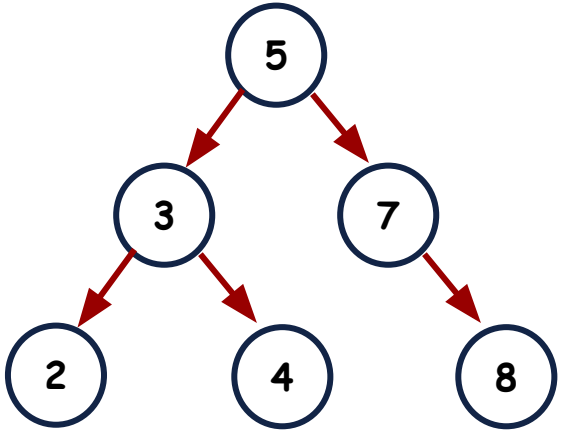
Time Complexity
O(n * h)

https://leetcode.com/problems/find-all-duplicates-in-an-array/description/

Solution: Find All Duplicates in an Array

What will be the time complexity of this Algorithm?

You can see that the height of the tree is way less than the input size.
Therefore, the worst time complexity will not exceed n^2

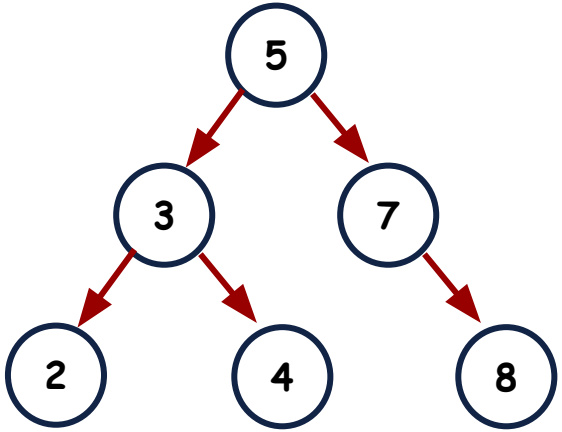


Time Complexity
O(n * h)

https://leetcode.com/problems/find-all-duplicates-in-an-array/description/

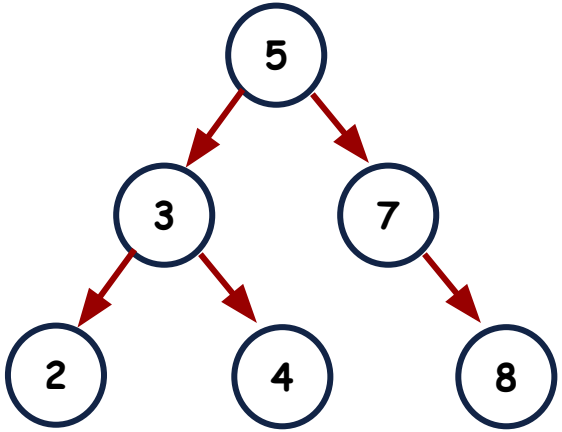
Binary Tree: Any specific order?

Do you see some specific order in this Binary Tree?



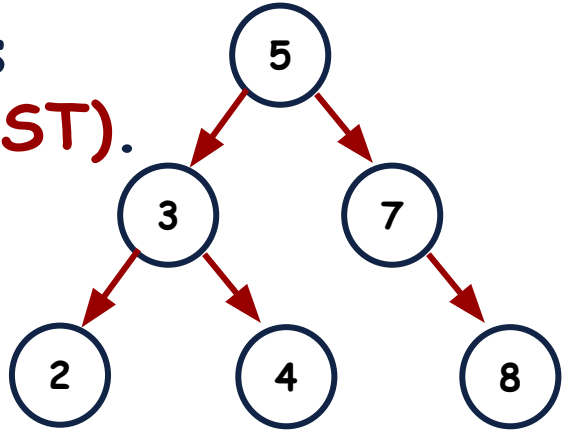
Binary Tree: Any specific order?

Yes, all the elements in the left subTree are less than the root node value and all the elements in the right subTree are greater than the root node value.



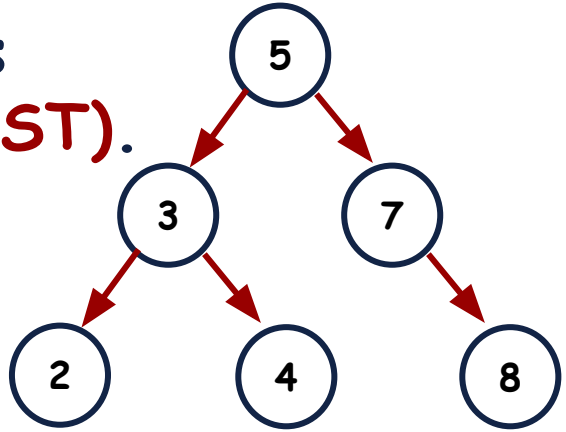
Binary Search Tree

This specific order binary tree is called the **Binary Search Tree (BST)**.



Binary Search Tree

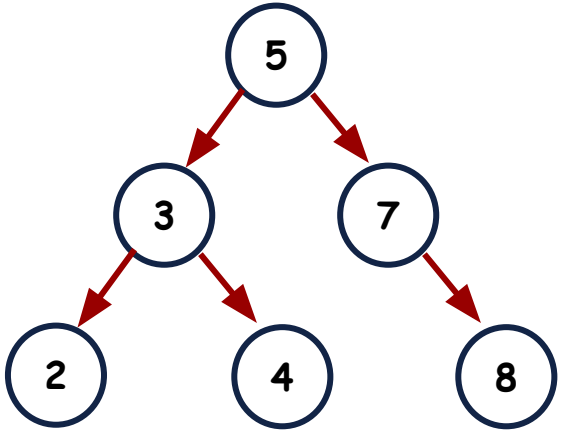
This specific order binary tree is called the **Binary Search Tree (BST)**.



It is obvious from the name that it is very helpful in searching the values.

Binary Search Tree: Food for thought

What will be the output if we traverse Binary Search Tree in In-Order Traversal?

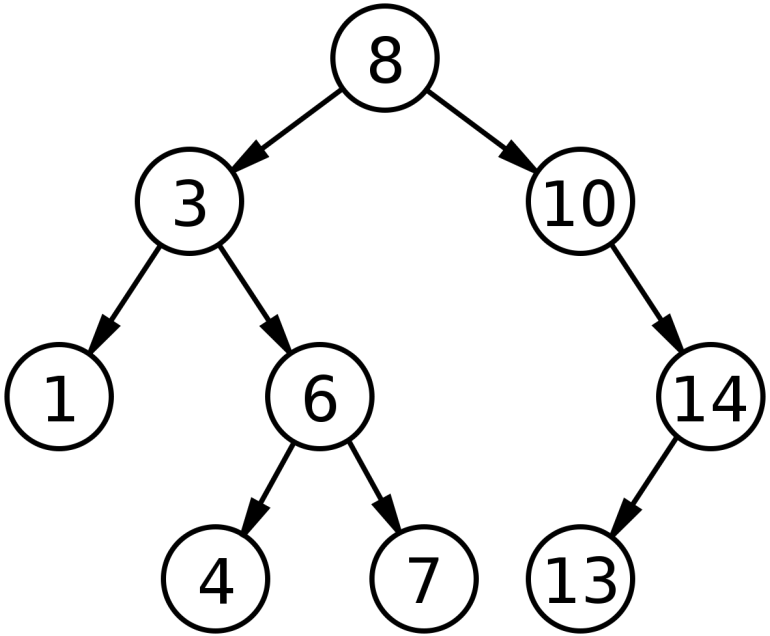


Learning Objective

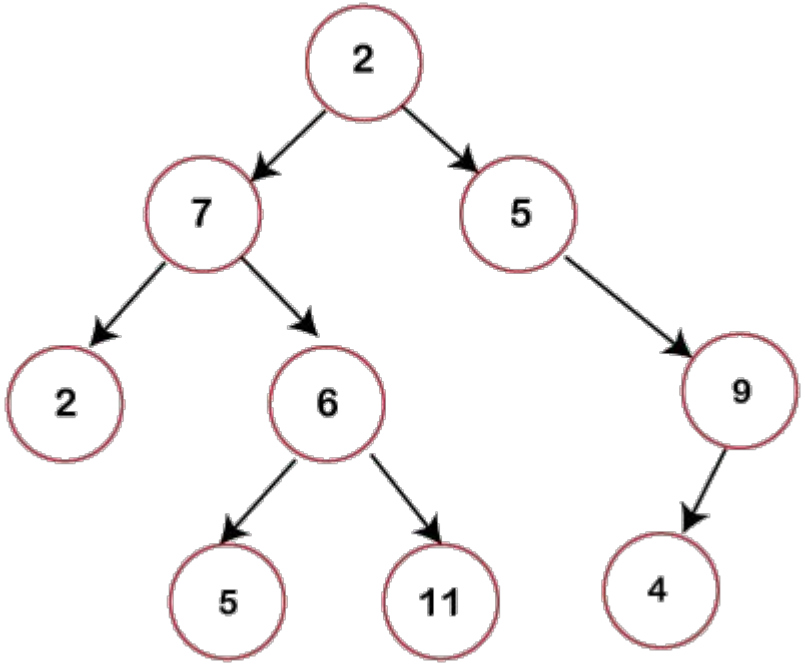
Students should be able to understand **Binary Search Trees** in order to solve the problems **efficiently**.

Self Assessment

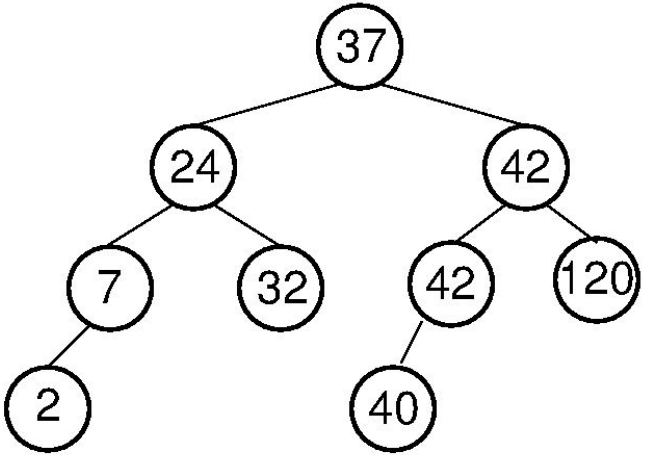
Which of these are binary search trees and which of these are just binary trees?



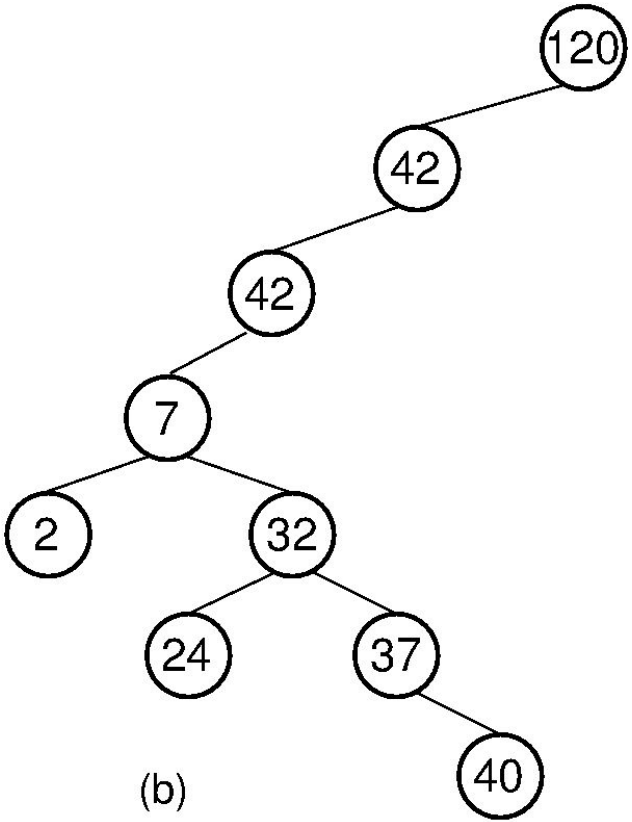
Self Assessment



Self Assessment

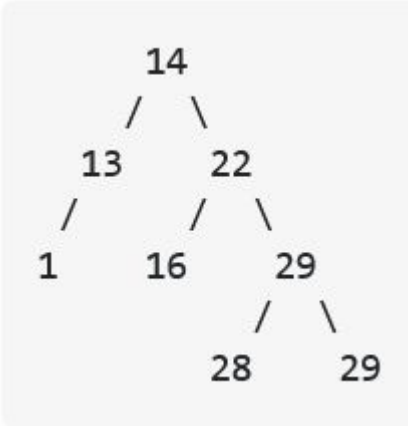


(a)



(b)

Self Assessment



Online Links

btv.melezinek.cz/binary-search-tree.html
https://visualgo.net/en/bst