Implementing Data Structures

EECS 1030

moodle.yorku.ca

Binary trees

Binary trees are used to

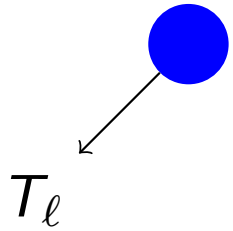
- implement `TreeSet` and `TreeMap`,
- determine which objects need to be rendered in a video game,
- store router tables in high-bandwidth routers,
- compress files (mp3 format),
- store data (databases),
- etc.

Binary trees

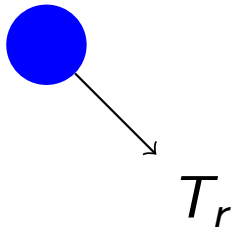
A binary tree has one of the following four shapes:

- a single node

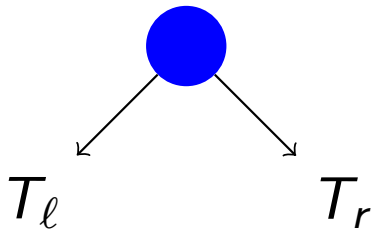
- a node with a left subtree



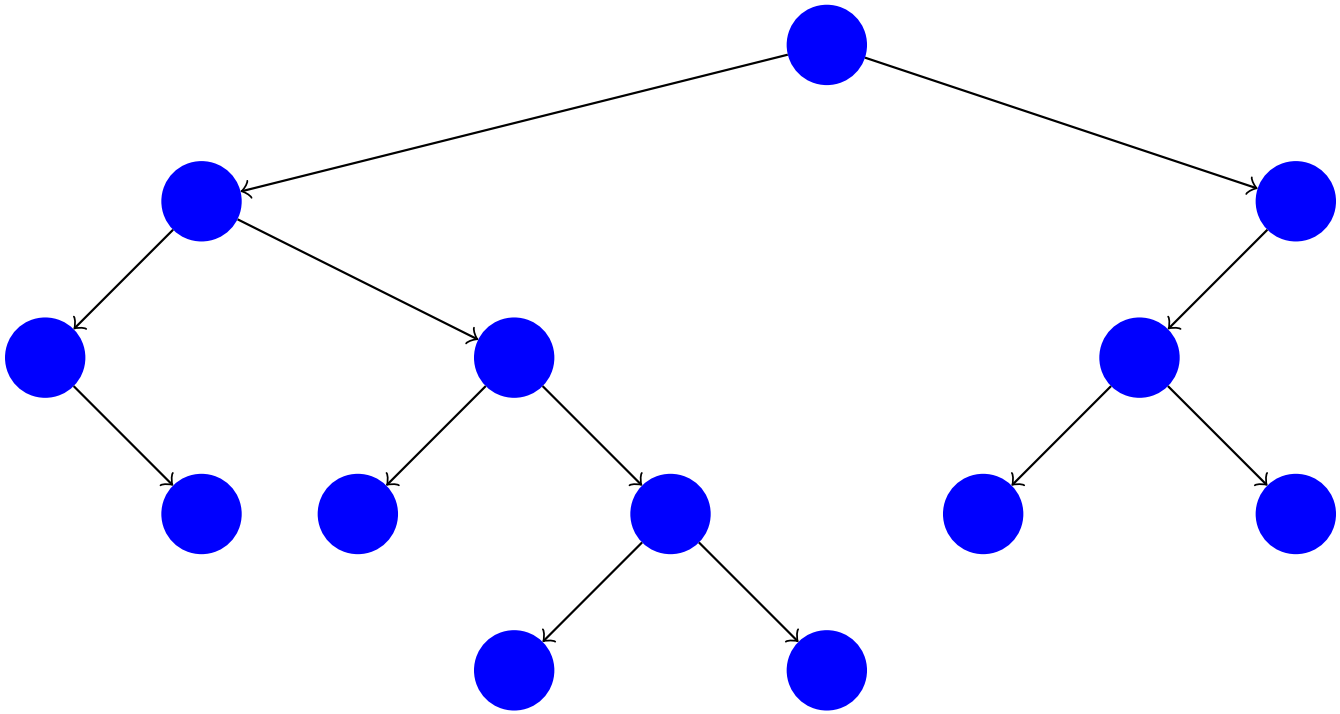
- a node with a right subtree



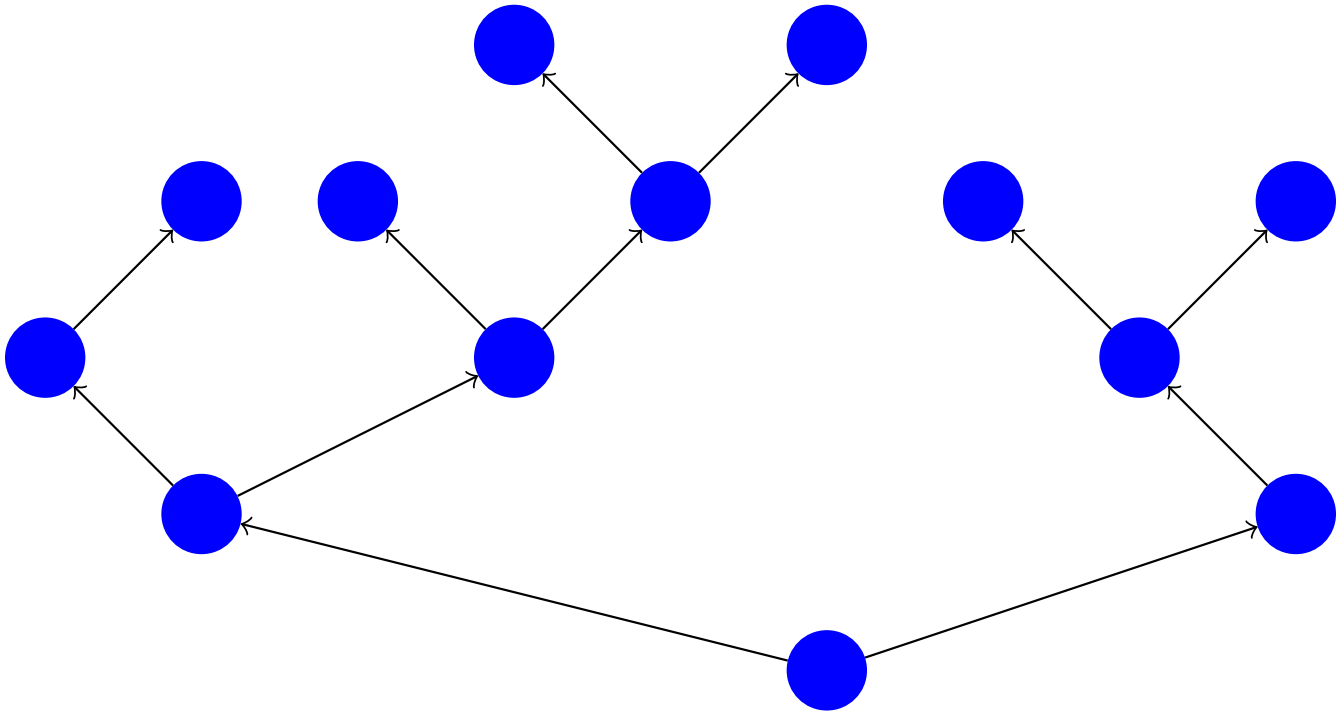
- a node with a left subtree and a right subtree



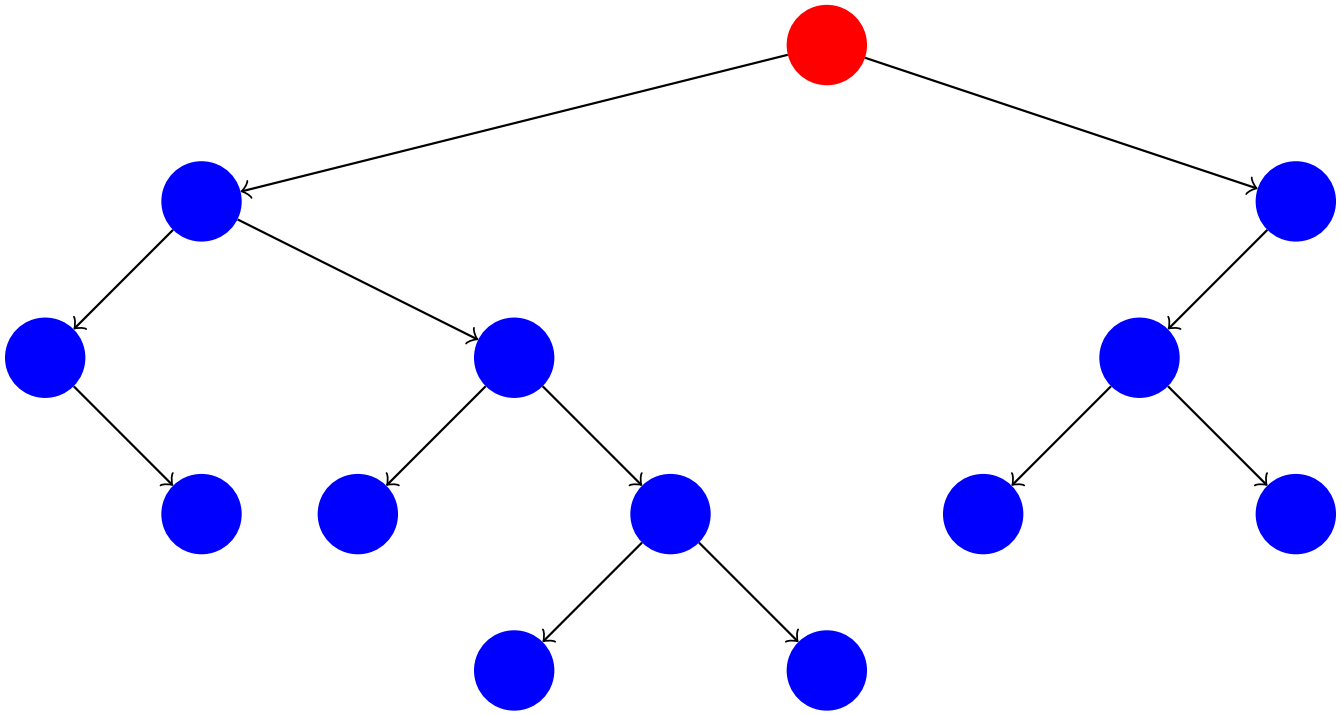
Binary trees



Binary trees

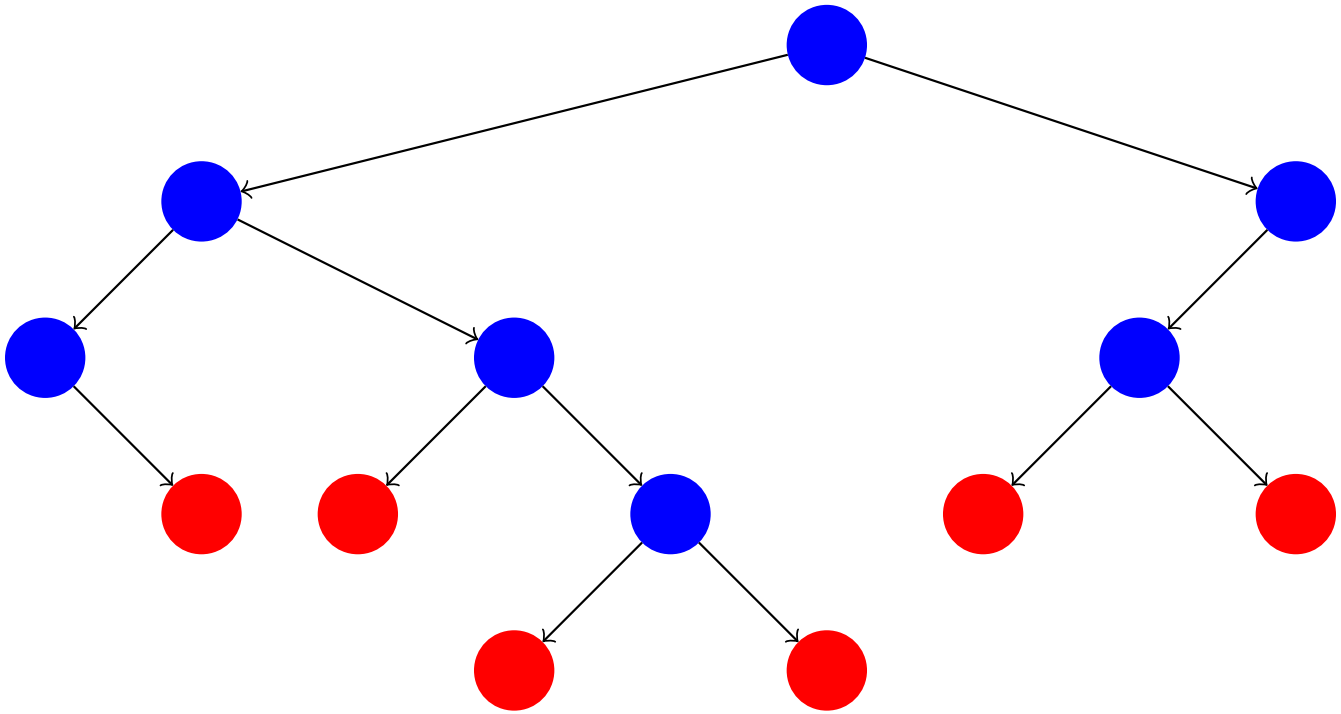


Binary trees



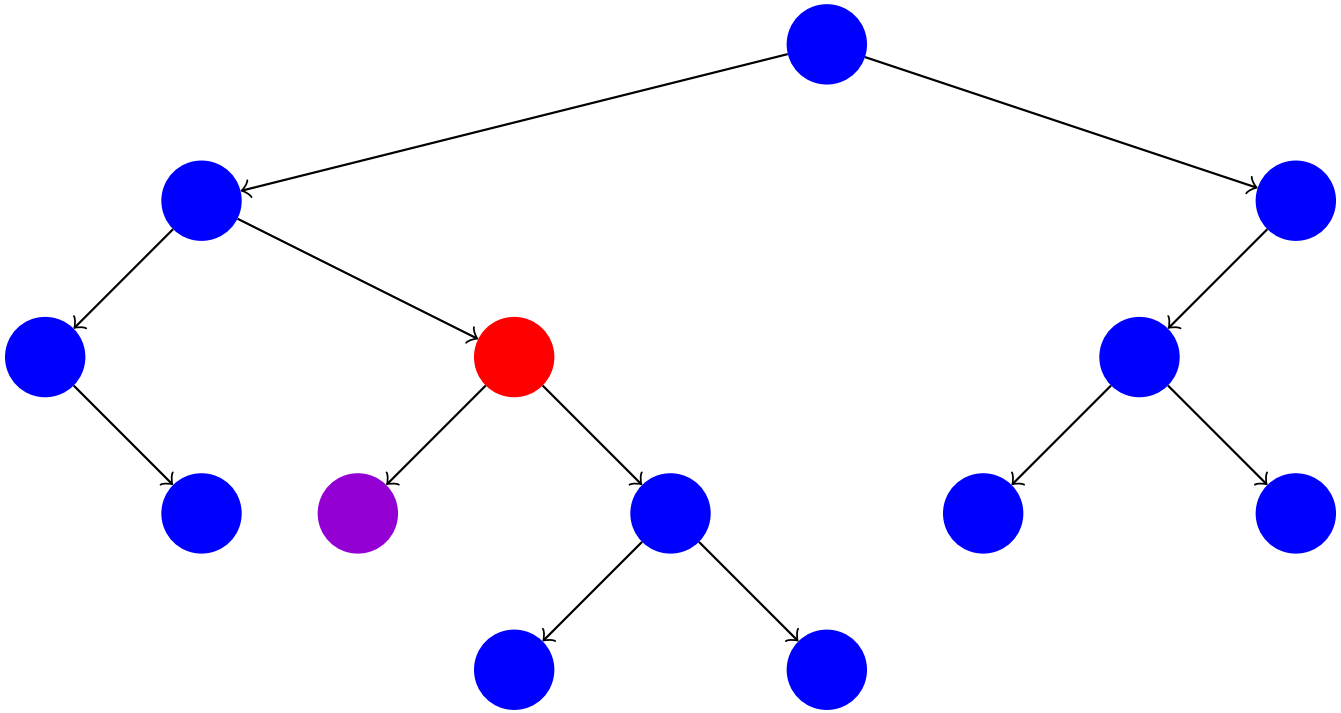
The red node is called the root.

Binary trees



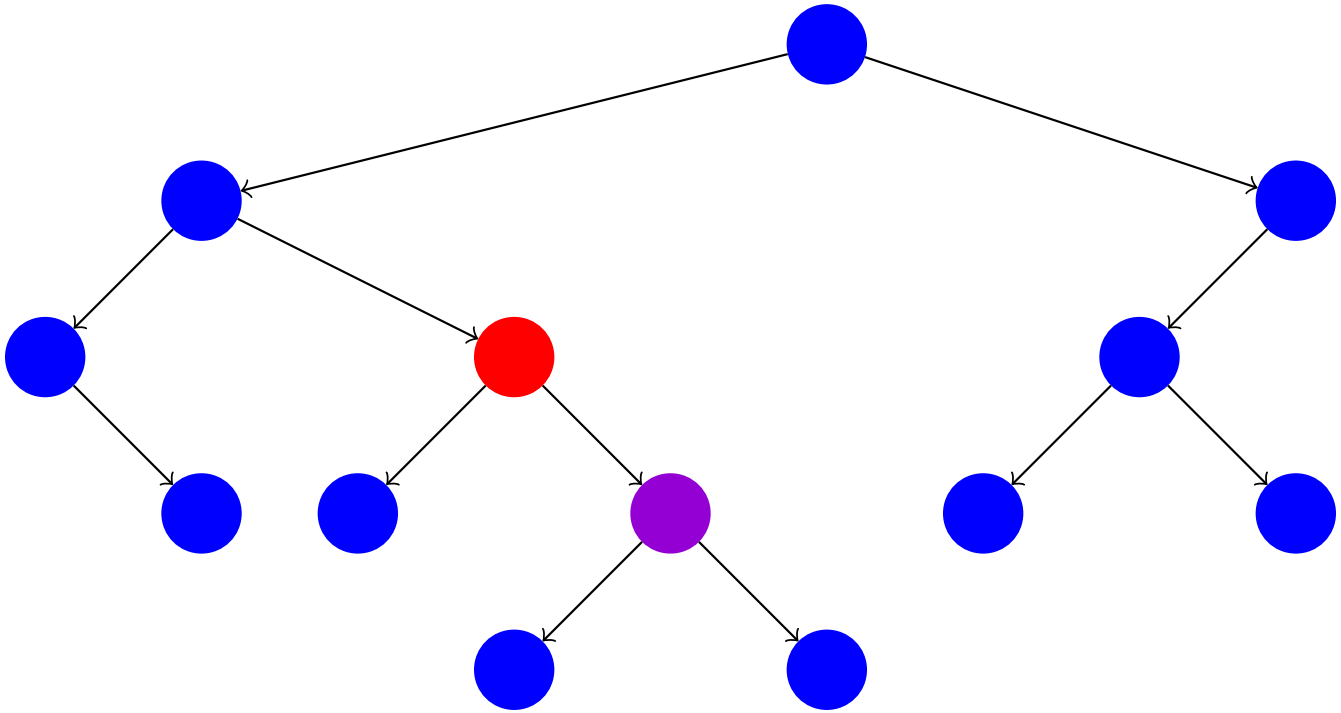
The red nodes are called leaves.

Binary trees



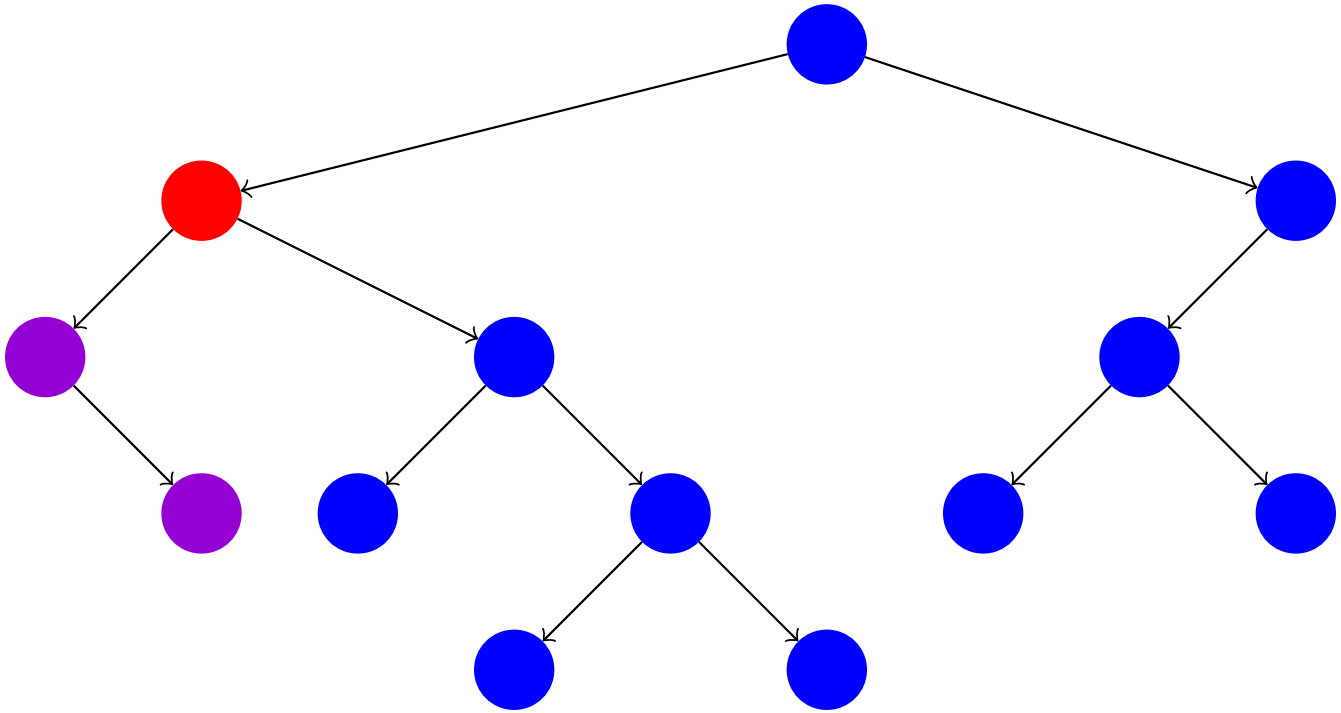
The purple node is the left child of the red node.

Binary trees



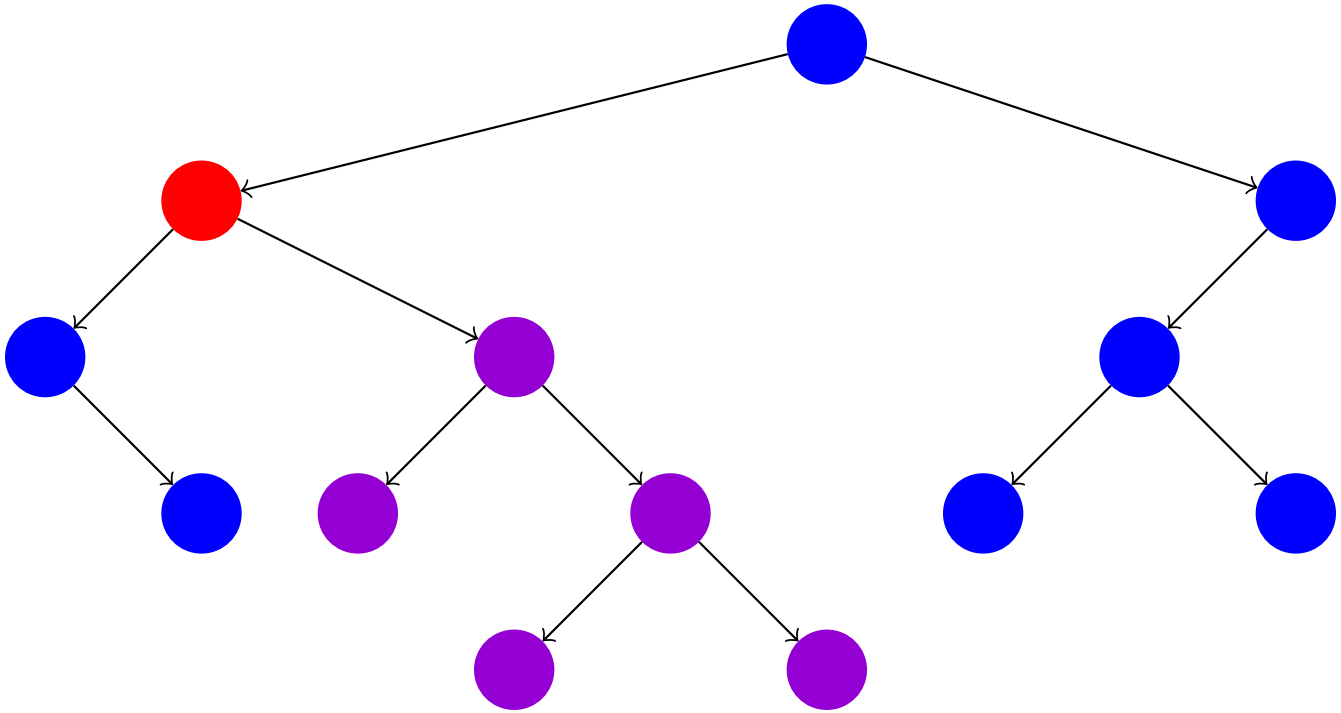
The purple node is the right child of the red node.

Binary trees



The purple nodes form the left subtree of the red node.

Binary trees



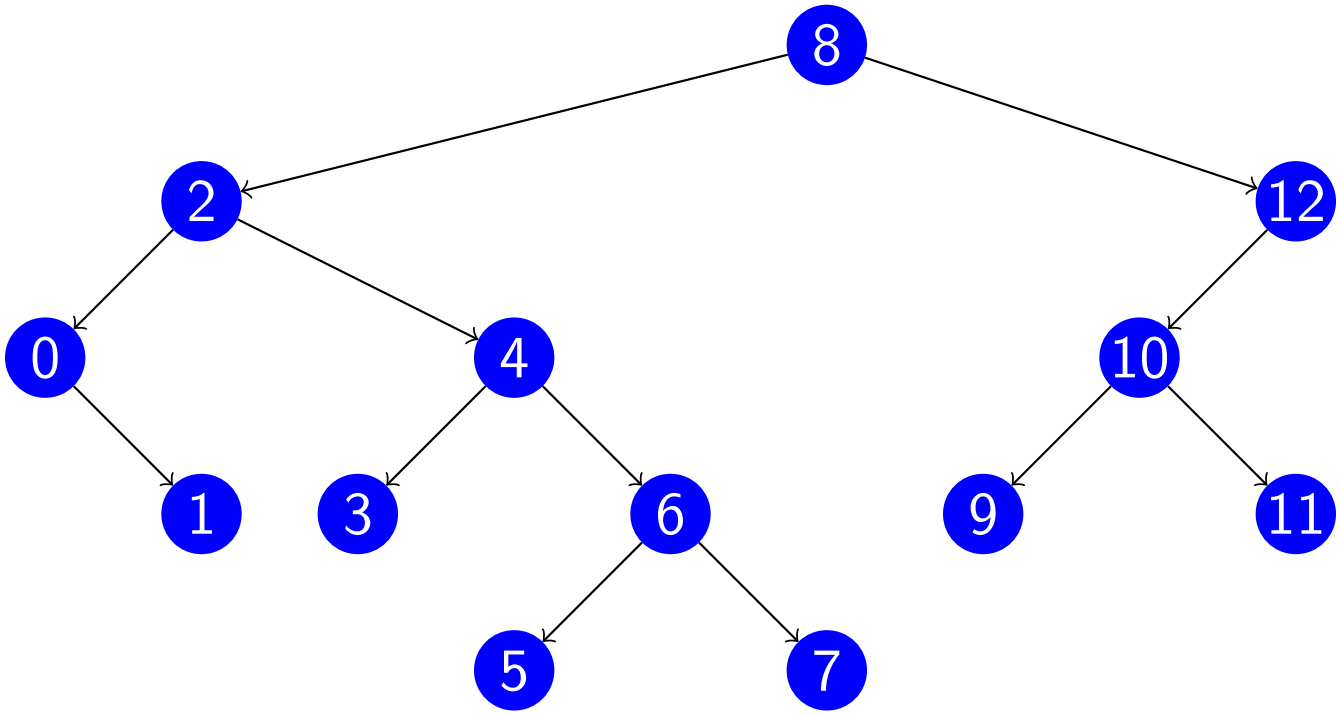
The purple nodes form the right subtree of the red node.

Binary search trees

A binary search tree is a binary tree where each node contains a value such that for each node of the tree

- if the node has a left child then the value of the node is greater than all the values stored in its left subtree, and
- if the node has a right child then the value of the node is greater than all the values stored in its right subtree.

Binary search trees



Binary search tree

Problem

Implement a BinarySearchTree.

Node of LinkedList

```
public class Node
{
    private String element;
    private Node next;

    ...
}
```

Node of BinarySearchTree

Draw the UML diagram of the `Node` class.

Node

Problem

Create a class named `Node` with attributes `element` of type `int` and `left` and `right` of type `Node`.