Data Structures

Tree Node

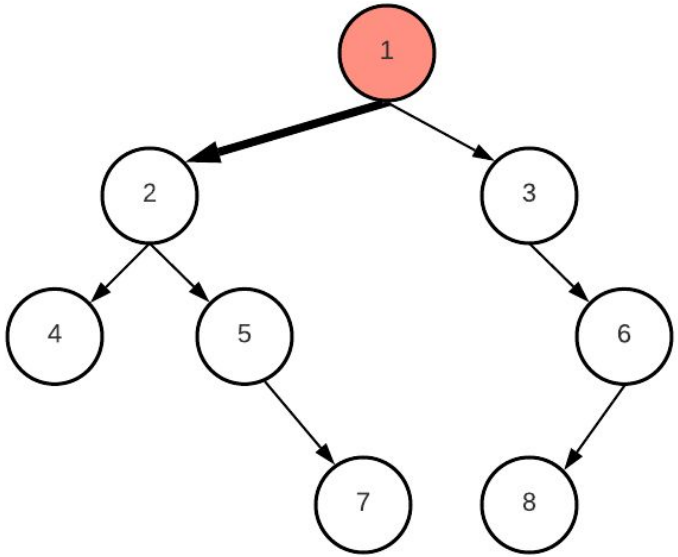
Mostafa S. Ibrahim
Teaching, Training and Coaching since more than a decade!

Artificial Intelligence & Computer Vision Researcher
PhD from Simon Fraser University - Canada
Bachelor / Msc from Cairo University - Egypt
Ex-(Software Engineer / ICPC World Finalist)

Tree Node

- Similar to linkedlist, we need to point to other nodes
  - Specifically 2 nodes: we called them left and right

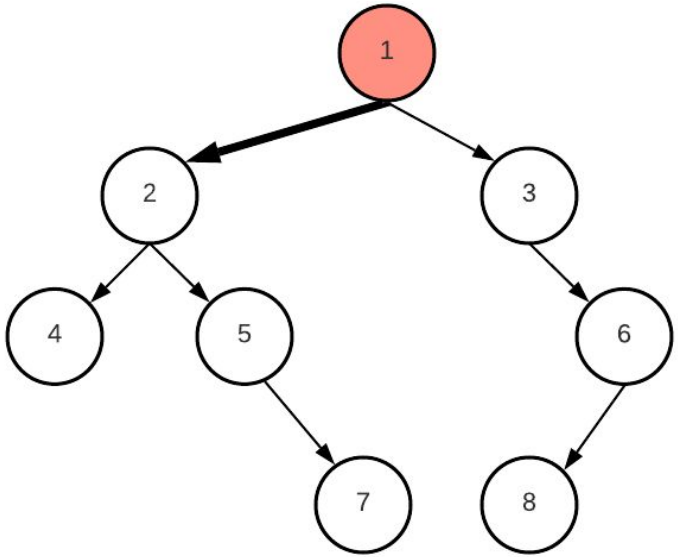
```
struct Node {
    int data { };
    Node* left { };
    Node* right { };
    Node(int data) :
            data(data) {
    }
};
```



Nodes Creation

- Let’s create the nodes, then link them

```
Node* root = new Node(1);
Node* node2 = new Node(2);
Node* node3 = new Node(3);
Node* node4 = new Node(4);
Node* node5 = new Node(5);
Node* node6 = new Node(6);
Node* node7 = new Node(7);
Node* node8 = new Node(8);
```



Edges Linking!

- Now we can link them. Congrats! We built a binary tree!
- Starting from the root: *how to print 7?*

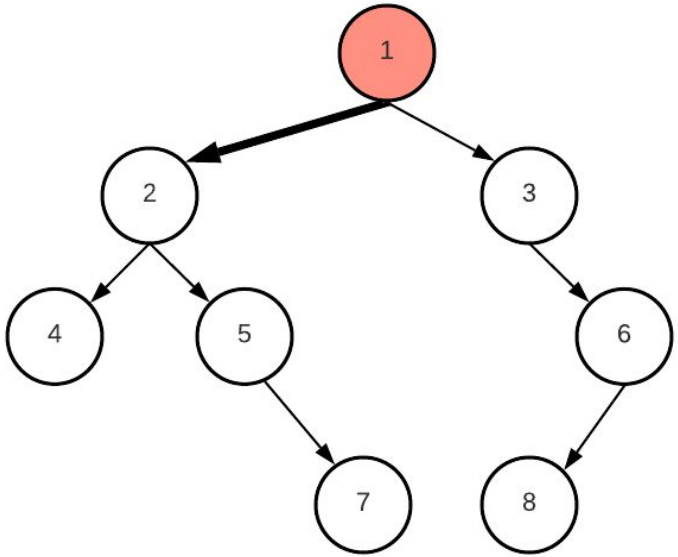
```
root->left = node2;
root->right = node3;

node2->left = node4;
node2->right = node5;

node5->right = node7;

node3->right = node6;

node6->left = node8;
```



Printing paths

- Path is chain of nodes, e.g. [1 ⇒ 2 ⇒ 5 ⇒ 7]
- Common mistake, like linkedlist, to use a child node that is null pointer!

```
root->left = node2;
root->right = node3;

node2->left = node4;
node2->right = node5;

node5->right = node7;

node3->right = node6;

node6->left = node8;
```

```
cout << root->left->right->right->data << "\n"; // 7
cout <<      node2->right->right->data << "\n"; // 7
cout <<             node5->right->data << "\n"; // 7
cout <<                    node7->data << "\n"; // 7

cout << root->right->right->data << "\n";       // 6
cout << root->right->right->left->data << "\n"; // 8
cout << root->right->right->right << "\n";      // 0x00
```

Systematic printing!

- Printing the linked list was trivial, we keep going till the tail!
- But a tree now has 2 subtrees!
- To print a tree, you need to print its subtrees! Recursion!
- Your turn:
  - Implement: void print(node* node)
  - Call it with print(root)
  - The function should print the tree content!
- Bonus: How can we represent a binary tree using an array?
  - We will learn later in heap section

"Acquire knowledge and impart it to the people."

"Seek knowledge from the Cradle to the Grave."