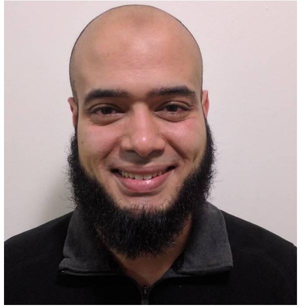
Data Structures

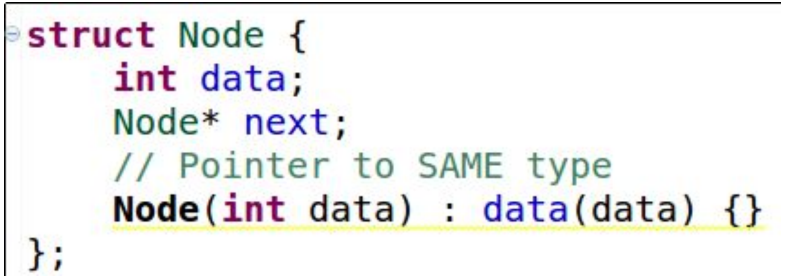
Doubly Linked List

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)

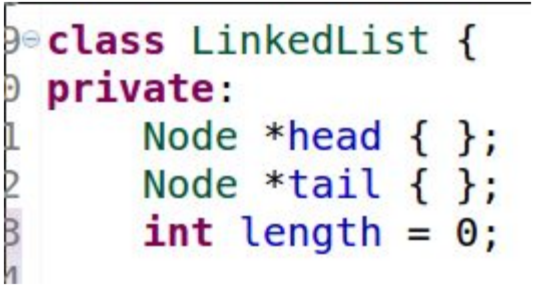


The Node & Linked List data structures

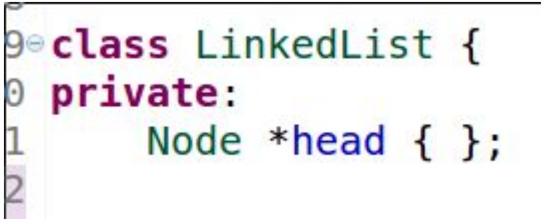


```
struct Node {
    int data;
    Node* next;
    // Pointer to SAME type
    Node(int data) : data(data) {}
};
```

- Different data structuring **design choices** will impact your code
  - Time & Memory Speed
  - Data assumption (e.g. data reversed)
  - Code simplicity!



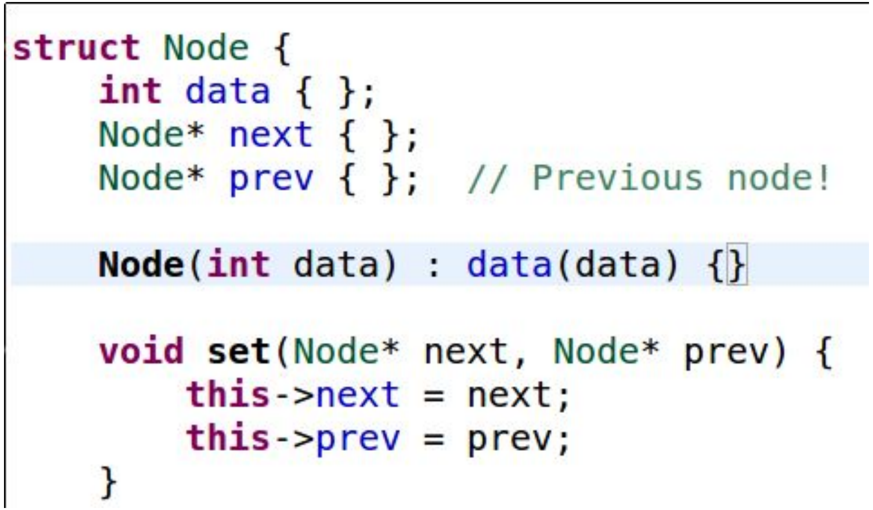
```
class LinkedList {
private:
    Node *head { };
    Node *tail { };
    int length = 0;
```



```
class LinkedList {
private:
    Node *head { };
```

Doubly Linked List

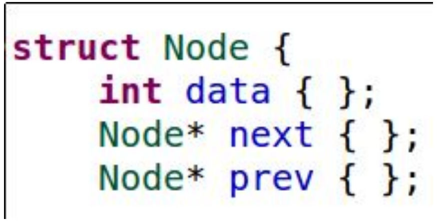
- Our node now has also a pointer to the **previous node**!
- This means we can move forward & backward easily!



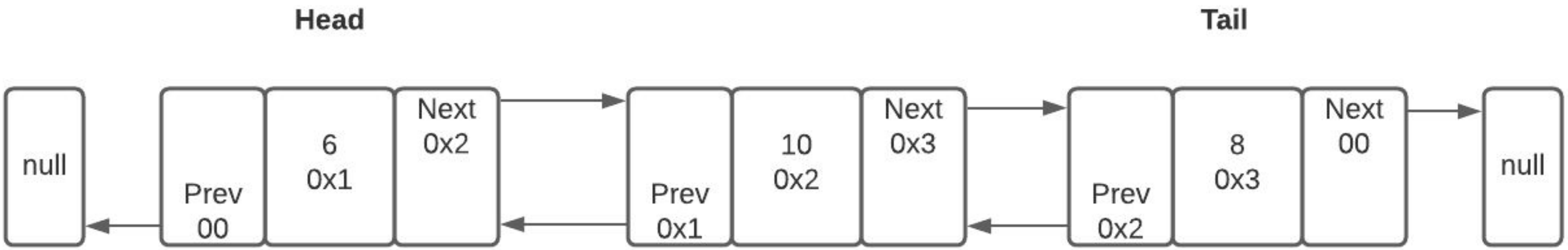
```
struct Node {
    int data { };
    Node* next { };
    Node* prev { };  // Previous node!

    Node(int data) : data(data) {}

    void set(Node* next, Node* prev) {
        this->next = next;
        this->prev = prev;
    }
```

Let’s visualize

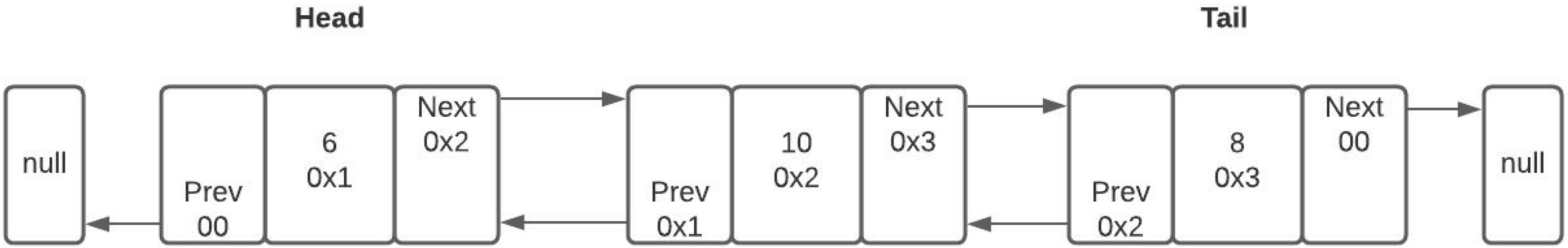
- We have 3 nodes, first is head and last is tail
- In **SLL**, node (6) is connected (with next) to node (10)
  - node1->next is node2
- In **DLL**, node (10) is connected (with previous) to node (6)
  - node2->prev is node 1
- Head has no previous (null) and tail has no next (null)

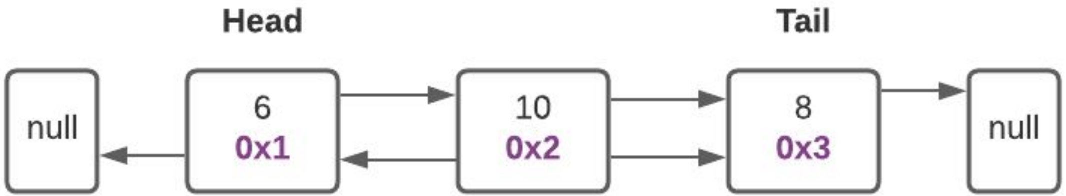


```
struct Node {
    int data { };
    Node* next { };
    Node* prev { };
```



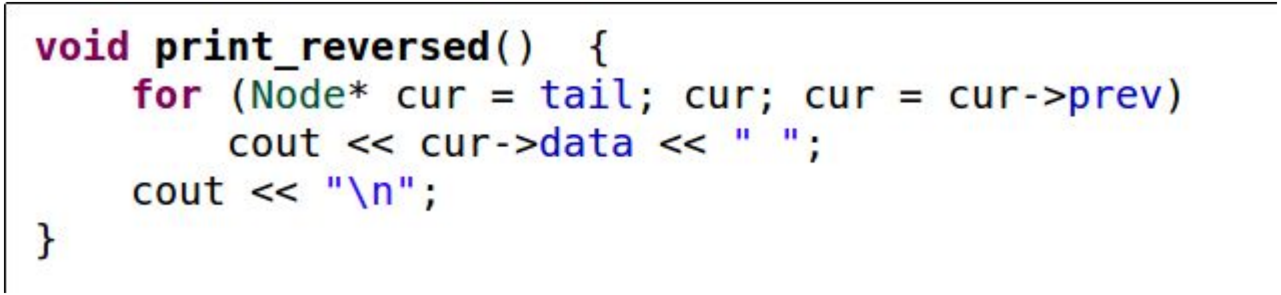
Let’s simplify drawing



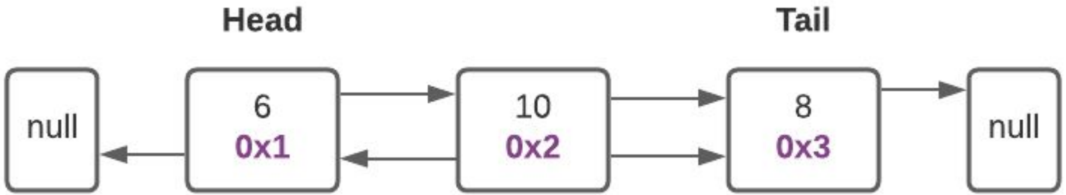


Print reversed

- Now we can go both forward and backward



```
void print_reversed()  {
    for (Node* cur = tail; cur; cur = cur->prev)
        cout << cur->data << " ";
    cout << "\n";
}
```



Why?!

- In many scenarios, we need to get the previous node!
- We can easily get this node in O(n)!
- With previous connection, we can have it in O(1)
- **In return**, you have to maintain data integrity for this added pointer!
    - Minor concern: It takes more space (another pointer)

"Acquire knowledge and impart it to the people."

"Seek knowledge from the Cradle to the Grave."