Mobile Team Training: Session 3

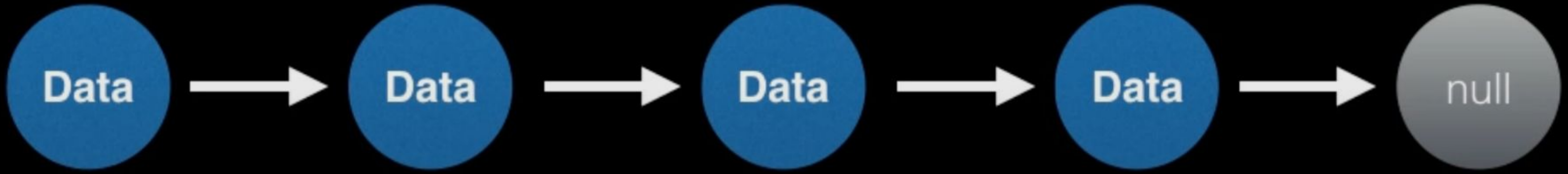
Linked List: Singly Linked List

By Sardorbek Abdulabbozov

What is Linked List?

What is a linked list?

A linked list is a sequential list of nodes that hold data which point to other nodes also containing data.



Where linked lists are used?

- Dynamic Memory Allocation: instead of dynamic arrays, for example;
- Implementing Other Data Structures: Stacks, Queues, Graphs;
- Handling Overflow in Hash tables;
- Undo Functionality in Applications: i.e Browsers;
- Much more...

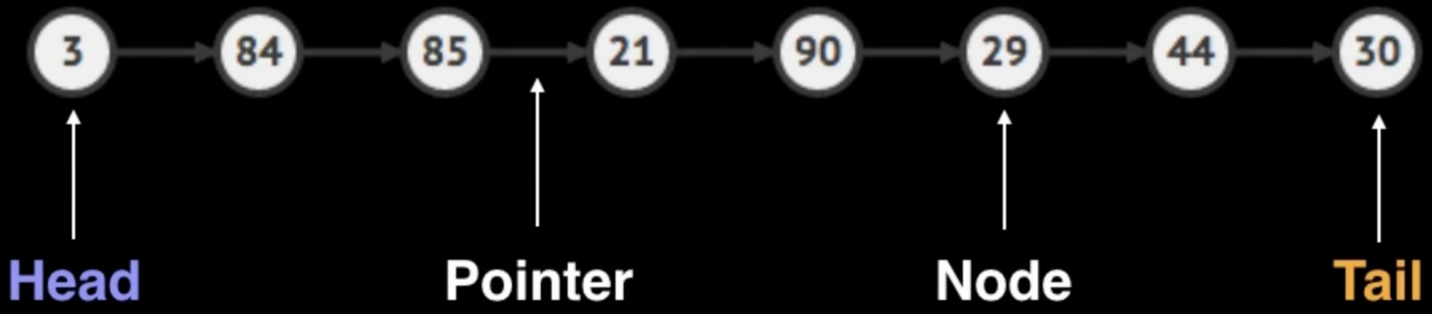
Terminology

Head: The first node in a linked list

Tail: The last node in a linked list

Pointer: Reference to another node

Node: An object containing data and pointer(s)



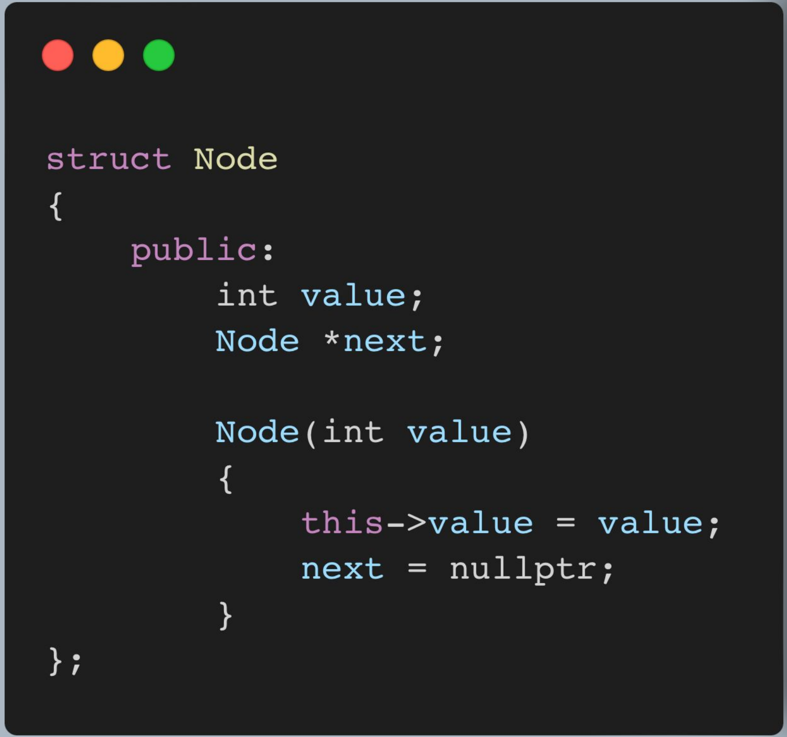
Head Pointer Node Tail

Time Complexity

	Array	Linked List (with tail pointer)
Access	O(1)	O(n)
Search	O(n)	O(n)
Insert/Delete (from beginning)	O(n)	O(1)
Delete (from end)	O(1)	O(n)
Insert (from end)	O(1)	O(1)
Insert/Delete (from middle)	O(n)	O(n)

Implementation of Linked List in C++ and Dart

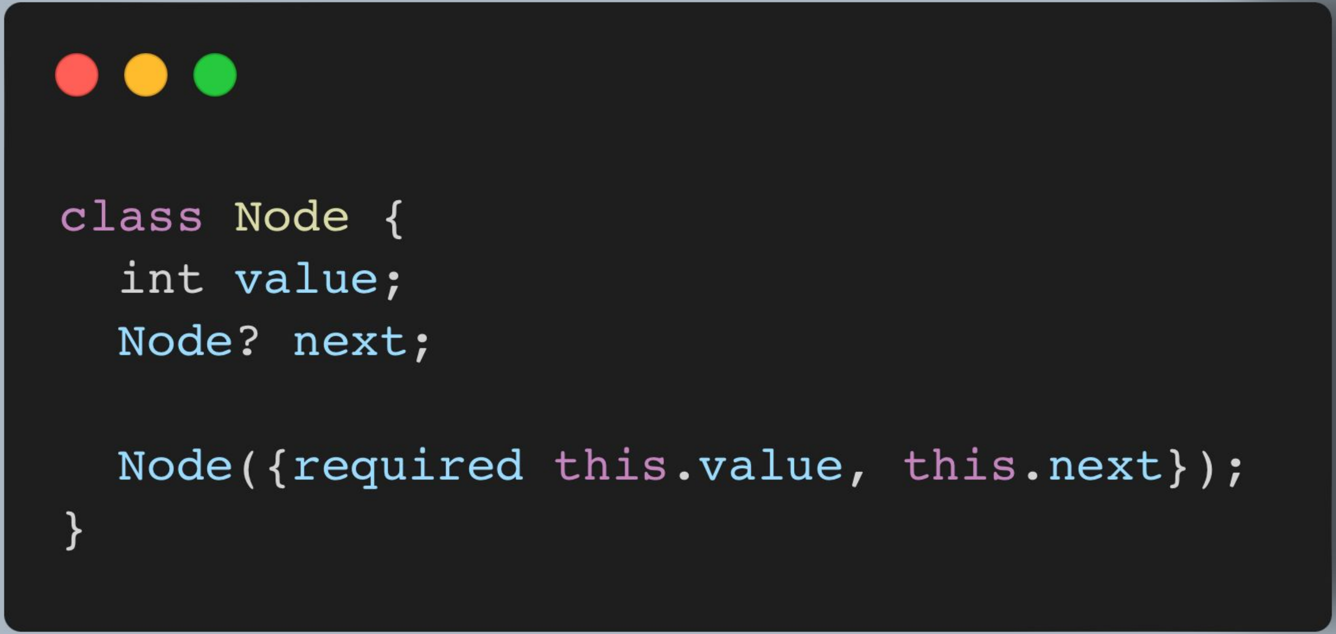
C++



```cpp
struct Node
{
    public:
        int value;
        Node *next;

        Node(int value)
        {
            this->value = value;
            next = nullptr;
        }
};
```

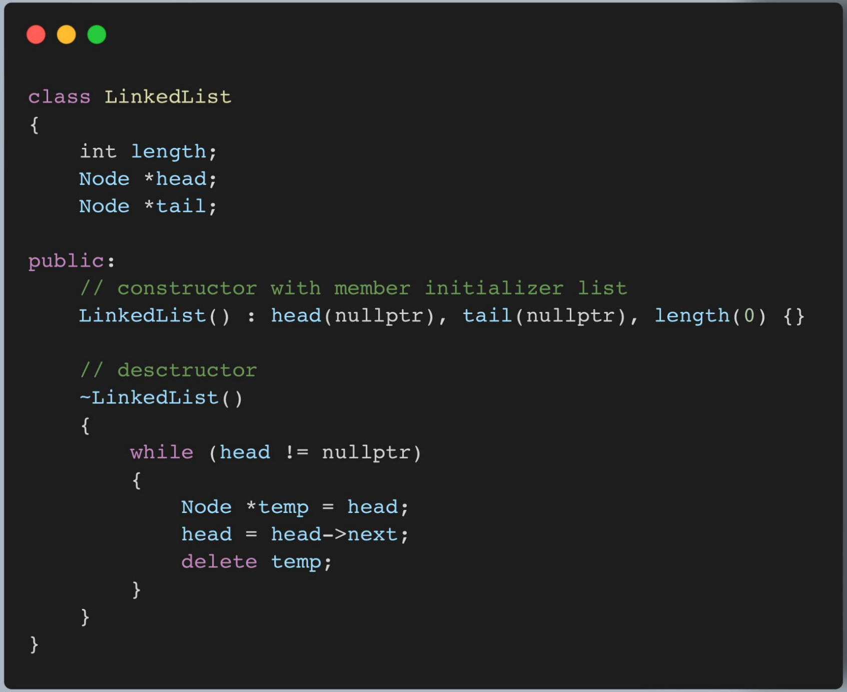
Dart



```dart
class Node {
  int value;
  Node? next;

  Node({required this.value, this.next});
}
```

C++

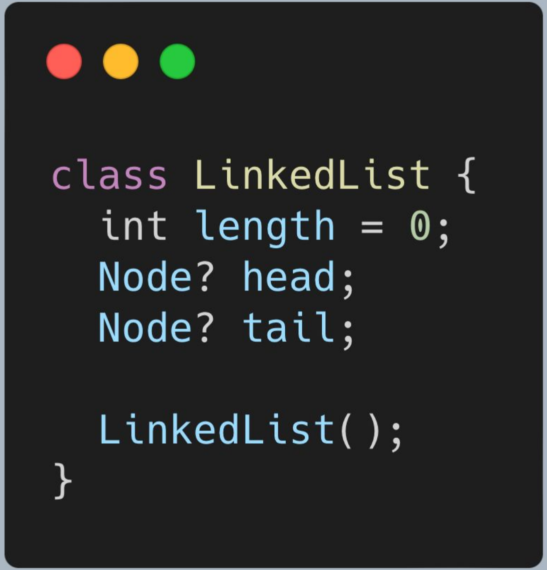


```
class LinkedList
{
    int length;
    Node *head;
    Node *tail;

public:
    // constructor with member initializer list
    LinkedList() : head(nullptr), tail(nullptr), length(0) {}

    // desctructor
    ~LinkedList()
    {
        while (head != nullptr)
        {
            Node *temp = head;
            head = head->next;
            delete temp;
        }
    }
}
```

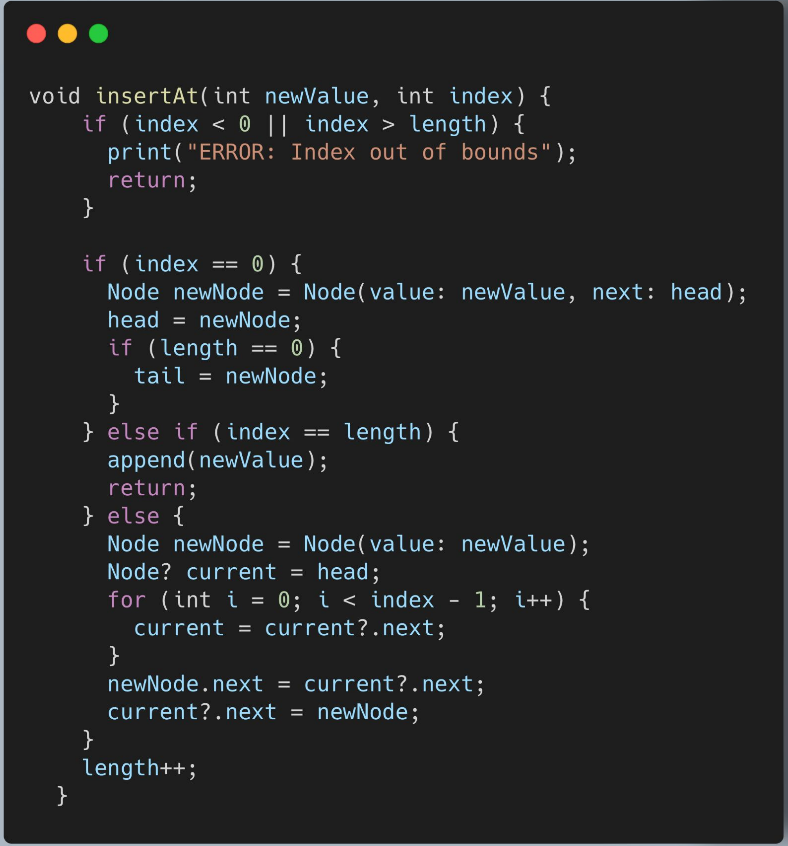
Dart



```
class LinkedList {
  int length = 0;
  Node? head;
  Node? tail;

  LinkedList();
}
```

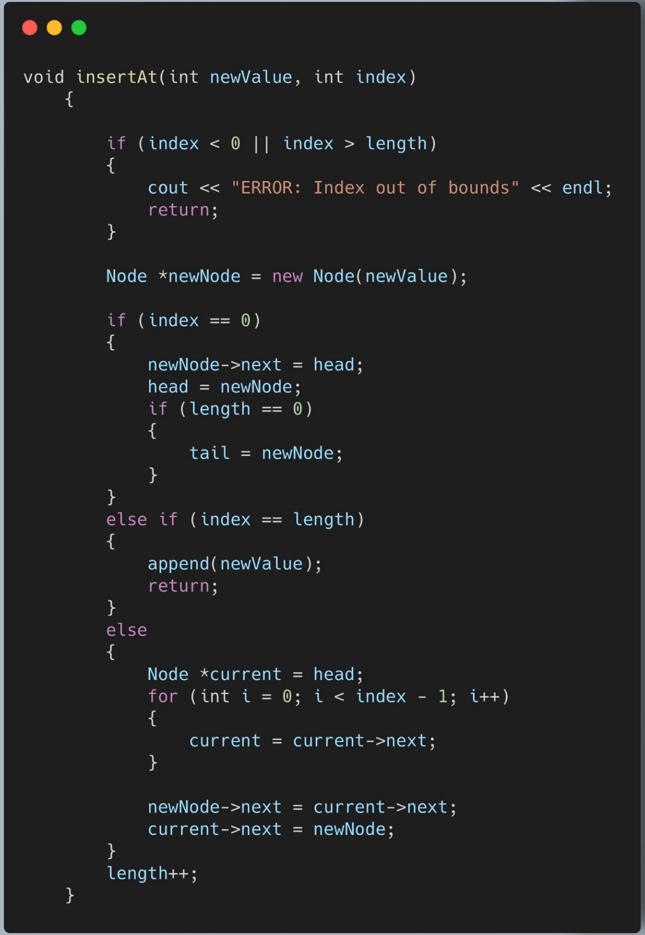
Dart



```
void insertAt(int newValue, int index) {
    if (index < 0 || index > length) {
      print("ERROR: Index out of bounds");
      return;
    }

    if (index == 0) {
      Node newNode = Node(value: newValue, next: head);
      head = newNode;
      if (length == 0) {
        tail = newNode;
      }
    } else if (index == length) {
      append(newValue);
      return;
    } else {
      Node newNode = Node(value: newValue);
      Node? current = head;
      for (int i = 0; i < index - 1; i++) {
        current = current?.next;
      }
      newNode.next = current?.next;
      current?.next = newNode;
    }
    length++;
  }
```

C++



```
void insertAt(int newValue, int index)
    {

        if (index < 0 || index > length)
        {
            cout << "ERROR: Index out of bounds" << endl;
            return;
        }

        Node *newNode = new Node(newValue);

        if (index == 0)
        {
            newNode->next = head;
            head = newNode;
            if (length == 0)
            {
                tail = newNode;
            }
        }
        else if (index == length)
        {
            append(newValue);
            return;
        }
        else
        {
            Node *current = head;
            for (int i = 0; i < index - 1; i++)
            {
                current = current->next;
            }

            newNode->next = current->next;
            current->next = newNode;
        }
        length++;
    }
```

Link for full implementation [C++]

Link for full implementation [Dart]

Thanks

References:

1) https://www.geeksforgeeks.org/program-to-implement-singly-linked-list-in-c-using-class/
2) https://www.youtube.com/watch?v=-Yn5DU0_-lw&list=PLDV1Zeh2NRsB6SWUrDFW2RmDotAfPbeHu&index=6