

- Recap Class / Object Intro
- Create a singly linked list class
 - Define constructor
 - Define grow-list method
 - Display list method
 - Define destructor

Recap Class / Object

- Object is an instance of a class.
- Similar to structs, but have public as well as private members
- Have member functions called methods.
 - ↳ control access to member data
 - Constructors used to initialize object of given class.
 - ↳ gets same name as the class

Create a Linked List Class

Still use a node struct as before

struct Node

```
{ string item;
  Node *next;
}
```

Introduce an approach to constructing a LL that lends itself well to class-based

- 1) 2 private members:
 - head and tail pointers

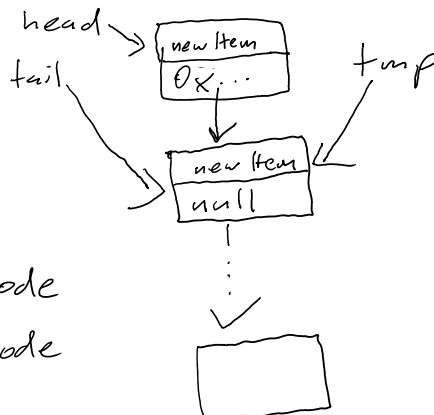
- 2) Public:

A) A constructor that initializes both pointers to null.

B) A grow method for allocating nodes dynamically (and also fill the node payload).

If LL is empty:

- 1) Allocate node and point head
- 2) Point tail to same node
- 3) Write contents to node member
- 4) Write null to head → next (or tail → next)



Else (LL not empty)

- 1) Use tmp pointer and allocate new node
 - 2) Write contents to node member
 - 3) tmp \rightarrow next point to null
 - 4) Write tail \rightarrow next = tmp
 - 5) Tail pointer point to last node
- C) A destructor - gets called automatically when object goes out of scope.
- L \rightarrow deallocate dynamic memory
- D) Display function
- Traverse & cout