AMACSS Study Group

Linked Lists (again)

A clearer idea of what can be a linked list

We can create any node object (heck, we could call it a potato object. But don't do that.) with the following properties:

- Holds data, or "cargo"
- contains a reference to the next and/or previous node(s)
    - i.e a pointer to the next and/or previous node(s)

Once we link those nodes together, that's a linked list. Simple as that.

Last week we talked about doubly linked lists. Let's go back and look at the implementation

How to traverse a linked list

When traversing a linked list, there are certain things we know that helps us get around.

For example, we know that the head of the list has nothing before it.
We also know that the tail of the list has nothing after it.

> From this we can tell that if the next pointer of our node is None, then we are at the end of the list

Lets see a diagram of this…

If we are here, then we know we are at the end of the list

Head

If we check the .next.next value of this Node, we will see that it is None. Therefore this is the second last node in the list

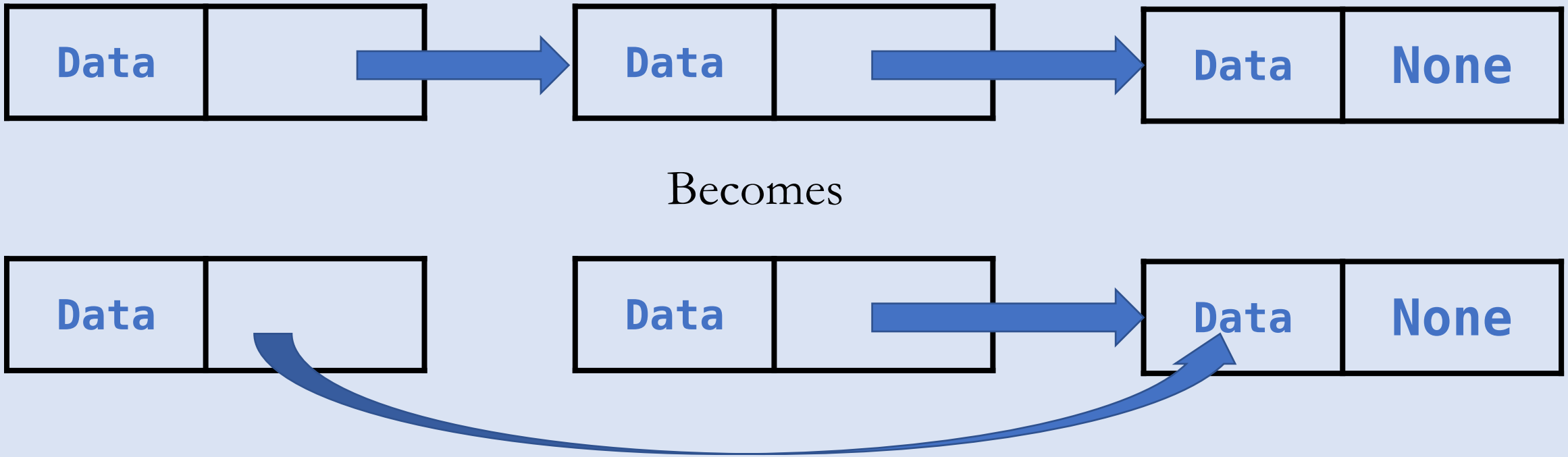
Current Node

Removing nodes from a linked list

To remove a node from a linked list we need to do a couple of things:

- remove the pointer to the node
- change the pointer of the previous node to point to the next node

If there is nothing pointing to a node, then it is no longer in our list.



Try implementing this: remove a node in a list at a specific index

Note: A Linked List node class can be found on the website in the week 3 code, just import that if you don't want to create your own Node class

Try the following questions for the remainder of thee study group:

- find the middle node of the list, and print its data
- find the second last node of the list, and print its data
- remove the head of the list, print the value of the new head
- add a node at a specific index
- add a node to the end of list, print the entire list

Note: A Linked List node class can be found on the website in the week 3 code, just import that if you don't want to create your own Node class

Please leave feedback at https://goo.gl/tZnNj0