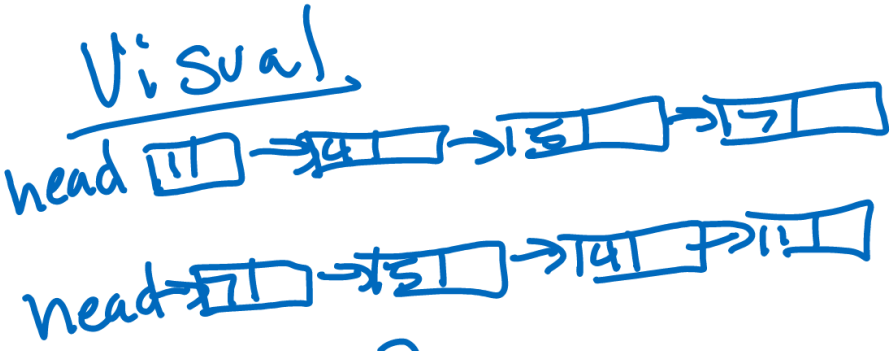
Whiteboard 10

Problem Domain

Reverse a singly linked list without using any built in methods. Aim to have an O(n) space solution.
*note no returning

Visual



so how?

Array.size-1 = this.root

this.root.next = Array.sig-2

while loop? while Array.sig>0

Algorithm (LL)

- find size of LinkedList using previously coded method
- Create a while loop that decrements size until its zero
- set LL root to LL to size -1.
- go through loop till all ListNodes are relinked using next property
- return Reversed list

Big O

this solution keeps O(n) space the same (no new list) and has a single while loop so time is also O(n)

Pseudo Code

```
ll = linkedList
int size = ll.size() - 1;
this.root := size - 1
int cycle = ll.size()
while (cycle < 0)
    assign this.root.next = size--
then return list
```

Code

```
public static void returnLinkedList(){
    int size = LinkedList.size() - 1;
    this.root = LinkedList[size];
    int cycle = LinkedList.size();
    this.root.next = LinkedList[size-1];
    // now new root has new connection to
        next node
    ListNode current = this.root.next;
    while(cycle - 2 < 0){
    current.next = LinkedList[size-2];
      current = current.next
       size--;
    }
    System.Out.println(LinkedList);
}
```

Tests

test for empty list

test for list only one node

" " " two nodes

test for LinkedList with at least four nodes

* use before each to setup LinkedLists using prepend method for tests