

White board 9

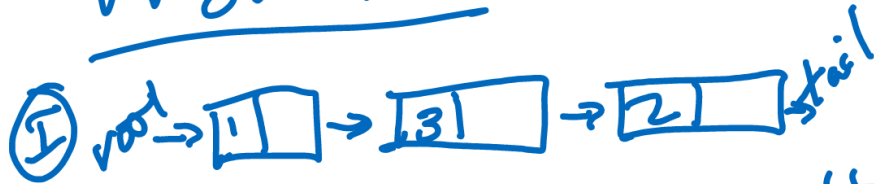
White board 9 Loop Detection

Problem Domain

Return a true/false for
a method that evaluates
a loop list. Method takes
in no arguments
(make a property of
linked lists class)

Big O required to stay
at $O(1)$ (no added
memory use allowed)

Visual

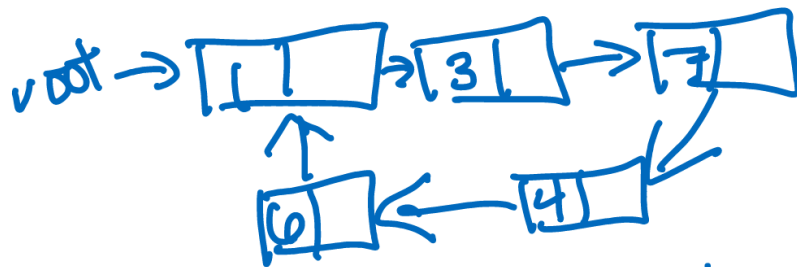


② false ~~no duplicate nodes~~
∴ not a looped LinkedList (LL)

case 1

case 2

①



② true - looped list
because end of
LL points to root/
head

Algo rithm

- set a pointer to first node

- set 2nd pointer to 3rd node
while (node.next != null) (+2)

if (first node.next != 3rd node)
increment (each one)

keep going looping till null
return false;

if (pointer A == pointer B)
return true

Code

```
public class Boolean hasLoop() {  
    LinkedList testList =  
        new LinkedList();
```

```
    ListNode pointerNodeA =  
        new ListNode();
```

```
    pointerNodeA =  $\emptyset$ ;
```

```
    pointerNodeB =  
        pointerNodeA + 2;
```

```
    while (pointerNodeA != null) {
```

```
        if (pointerNodeA ==  
            pointerNodeB) {
```

```
            return true
```

```
}
```

```
else if (pointer node A !=  
         pointer node B) {  
    pointer node A++;
```

```
}  
return false;
```

```
} // while loop close
```

```
} // method close
```