Singly and Doubly Linked Lists!

William Fiset

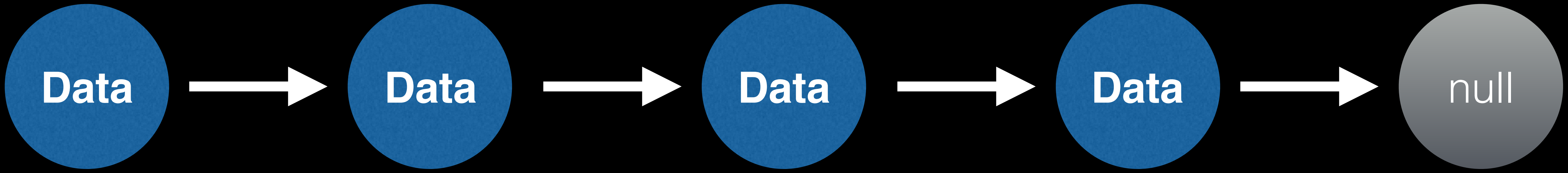
Outline

- **Discussion about Singly & Doubly Linked Lists**
    - What is a linked list?
    - Where are linked lists used?
    - Terminology
    - Singly Linked vs. Doubly Linked
- **Implementation Details**
    - How to insert new elements
    - How to remove elements
- **Complexity analysis**
- **Code Implementation (Doubly linked list)**

Discussion

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 are linked lists used?

- Used in many List, Queue & Stack implementations.
- Great for creating circular lists.
- Can easily model real world objects such as trains.
- Used in separate chaining, which is present certain Hashtable implementations to deal with hashing collisions.
- Often used in the implementation of adjacency lists for graphs.

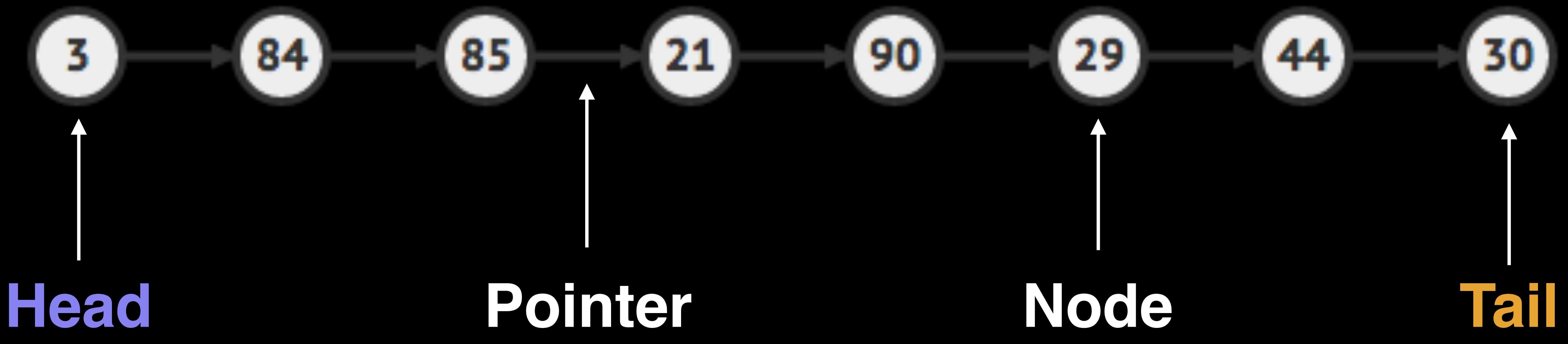
Terminology

Head: The first node in a linked list

Tail: The last node in a linked list

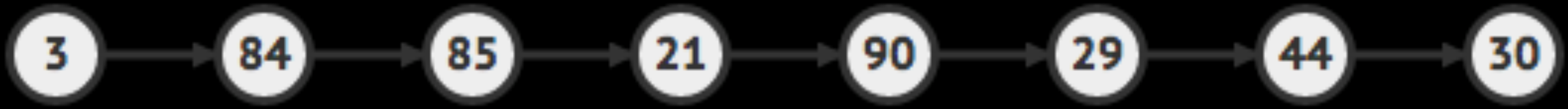
Pointer: Reference to another node

Node: An object containing data and pointer(s)

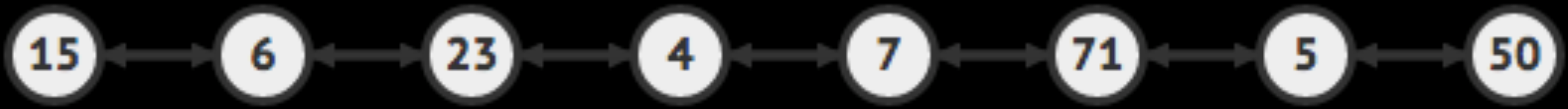


Singly vs Doubly Linked Lists

Singly linked lists only hold a reference to the next node. In the implementation you always maintain a reference to the **head** to the linked list and a reference to the **tail** node for quick additions/ removals.



With a doubly linked list each node holds a reference to the next and previous node. In the implementation you always maintain a reference to the **head** and the **tail** of the doubly linked list to do quick additions/removals from both ends of your list.



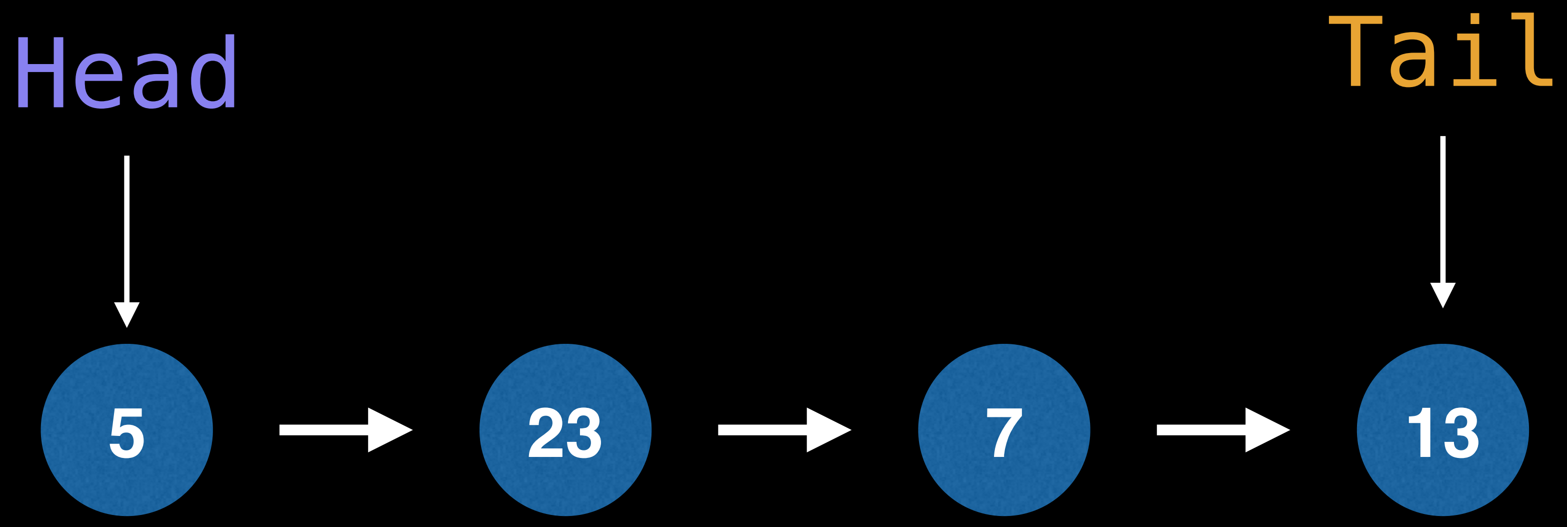
Singly & Doubly Linked lists Pros and Cons

	Pros	Cons
Singly Linked	Uses less memory Simpler implementation	Cannot easily access previous elements
Doubly Linked	Can be traversed backwards	Takes 2x memory

Implementation details

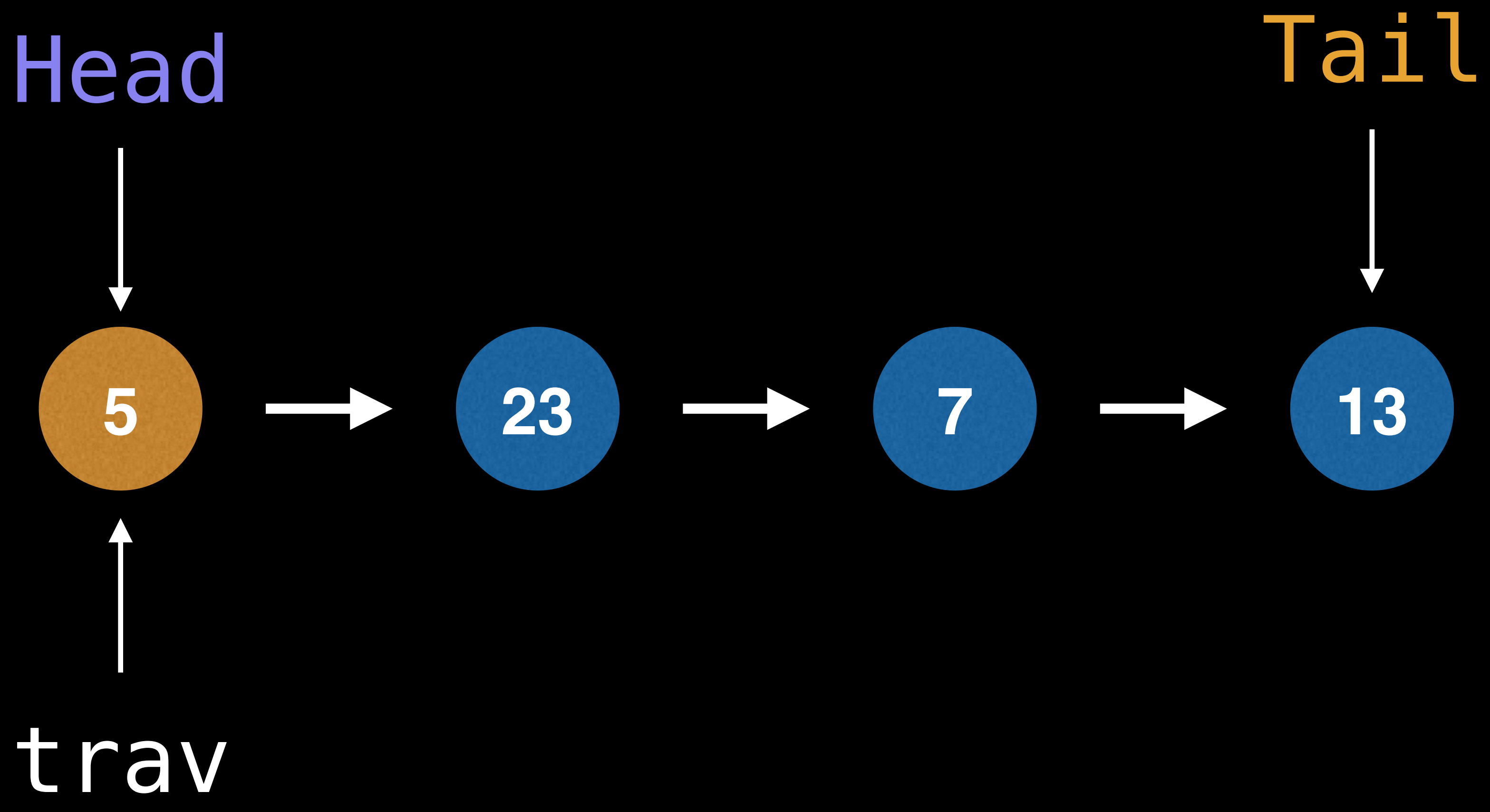
Inserting Singly Linked List

Insert 11 where the third node is.



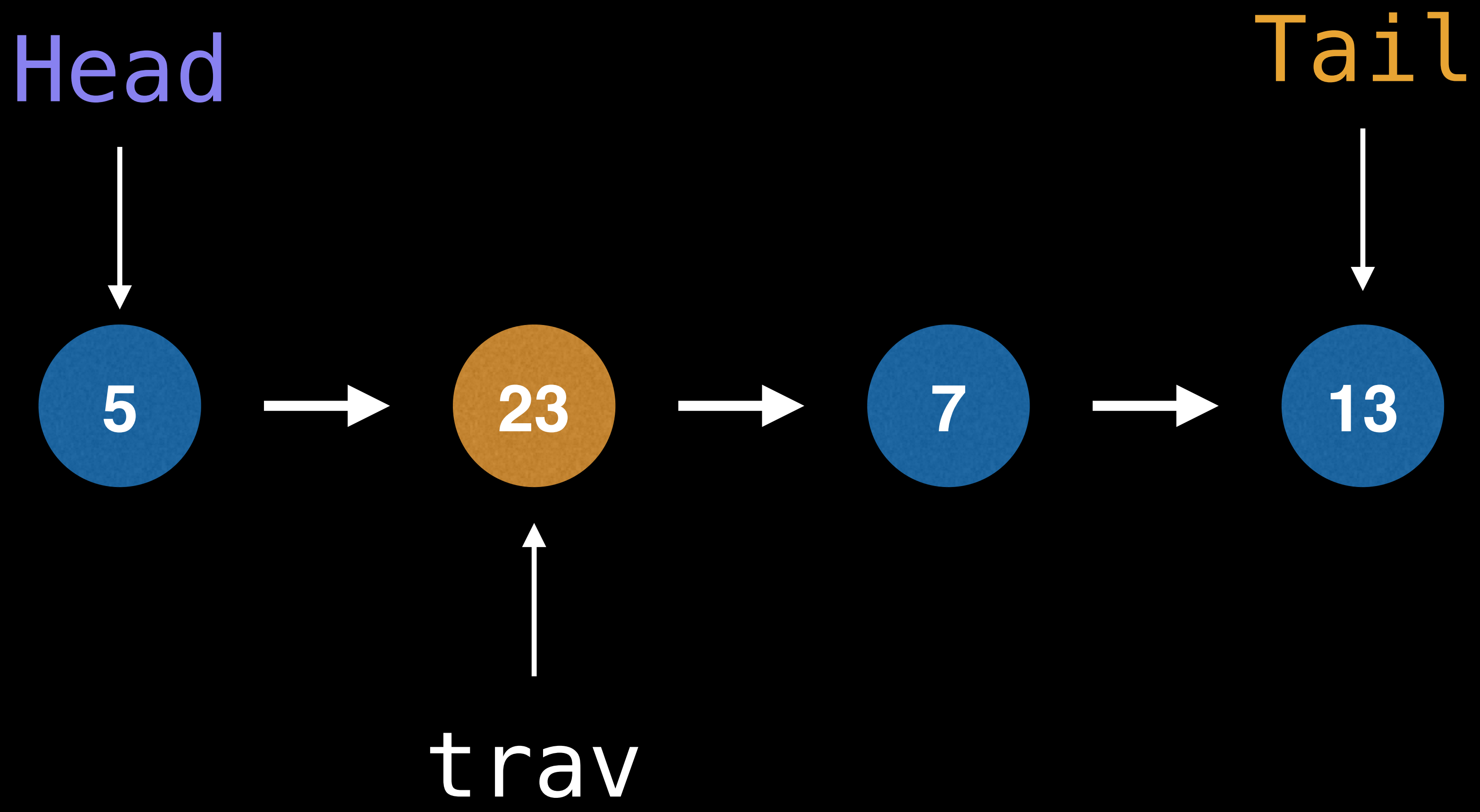
Inserting Singly Linked List

Insert 11 where the third node is.



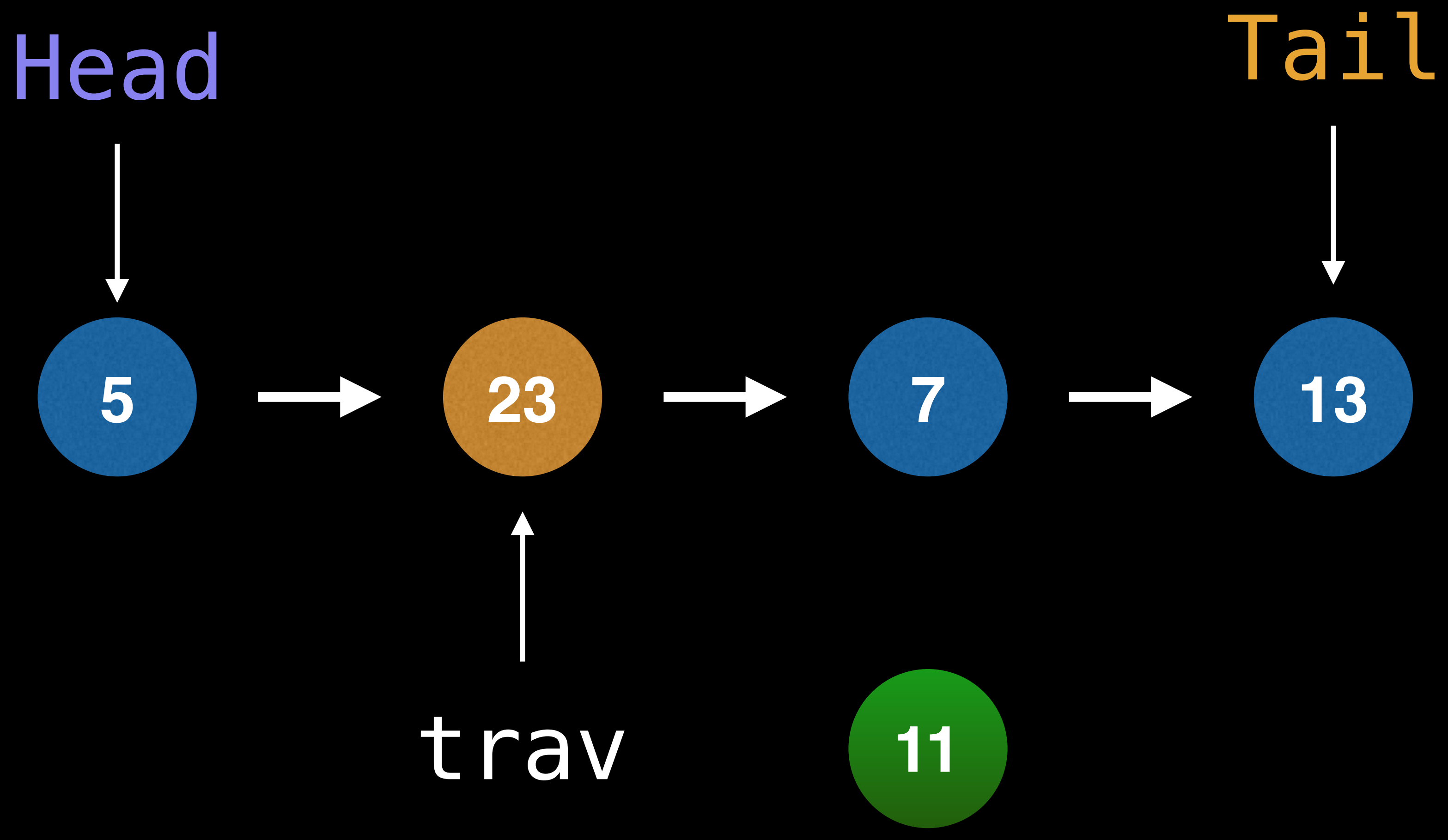
Inserting Singly Linked List

Insert 11 where the third node is.



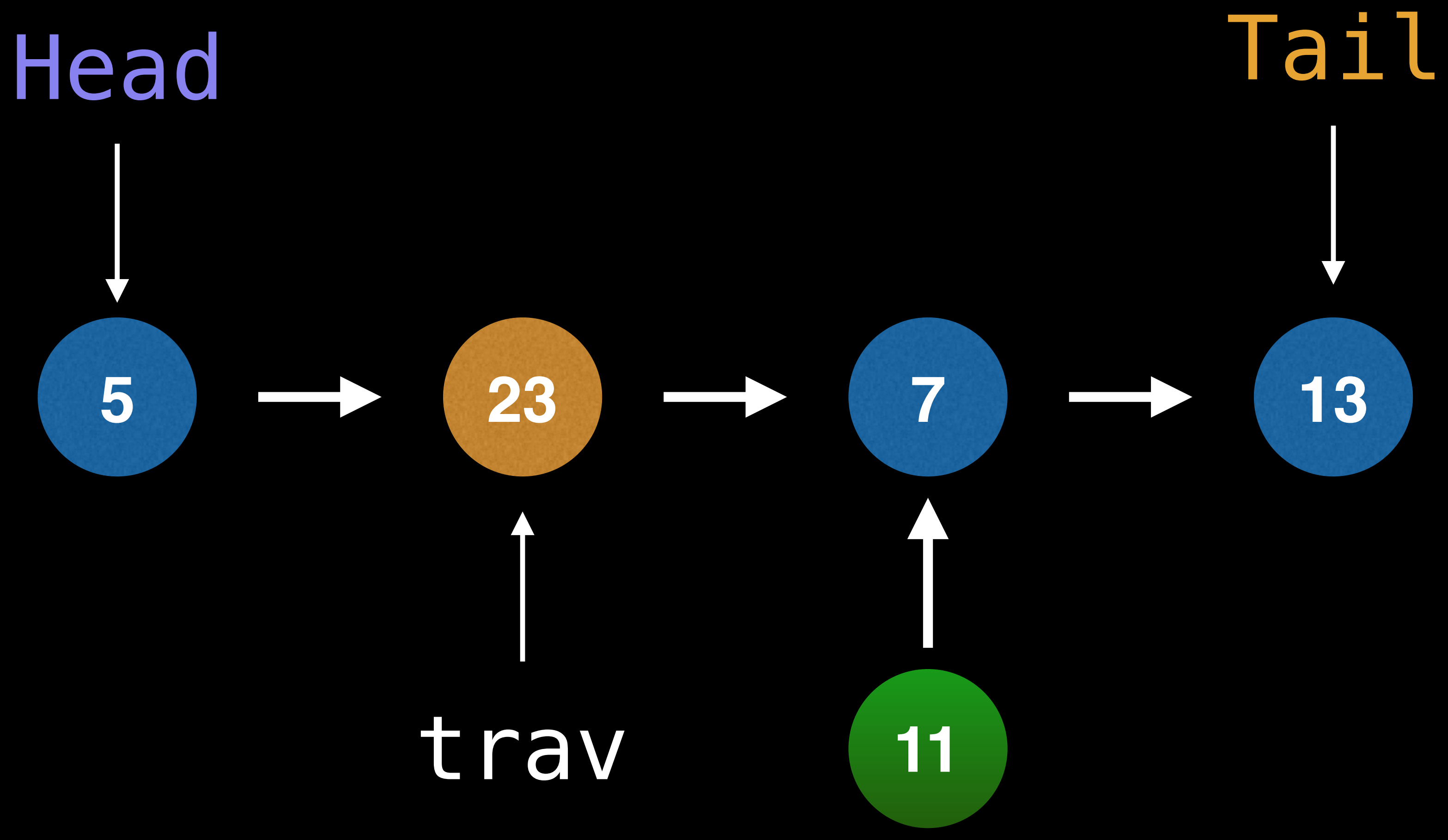
Inserting Singly Linked List

Insert 11 where the third node is.



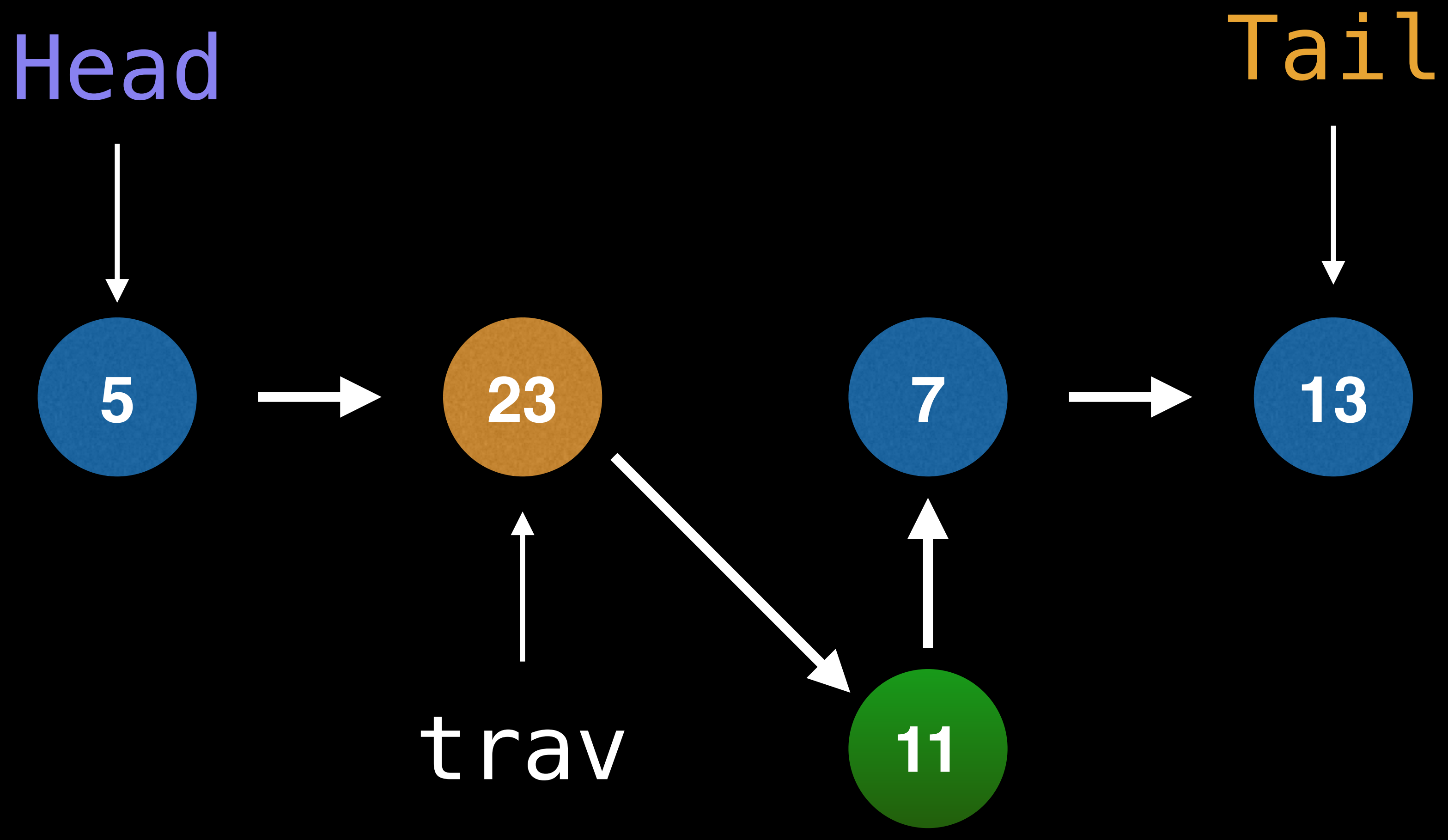
Inserting Singly Linked List

Insert 11 where the third node is.



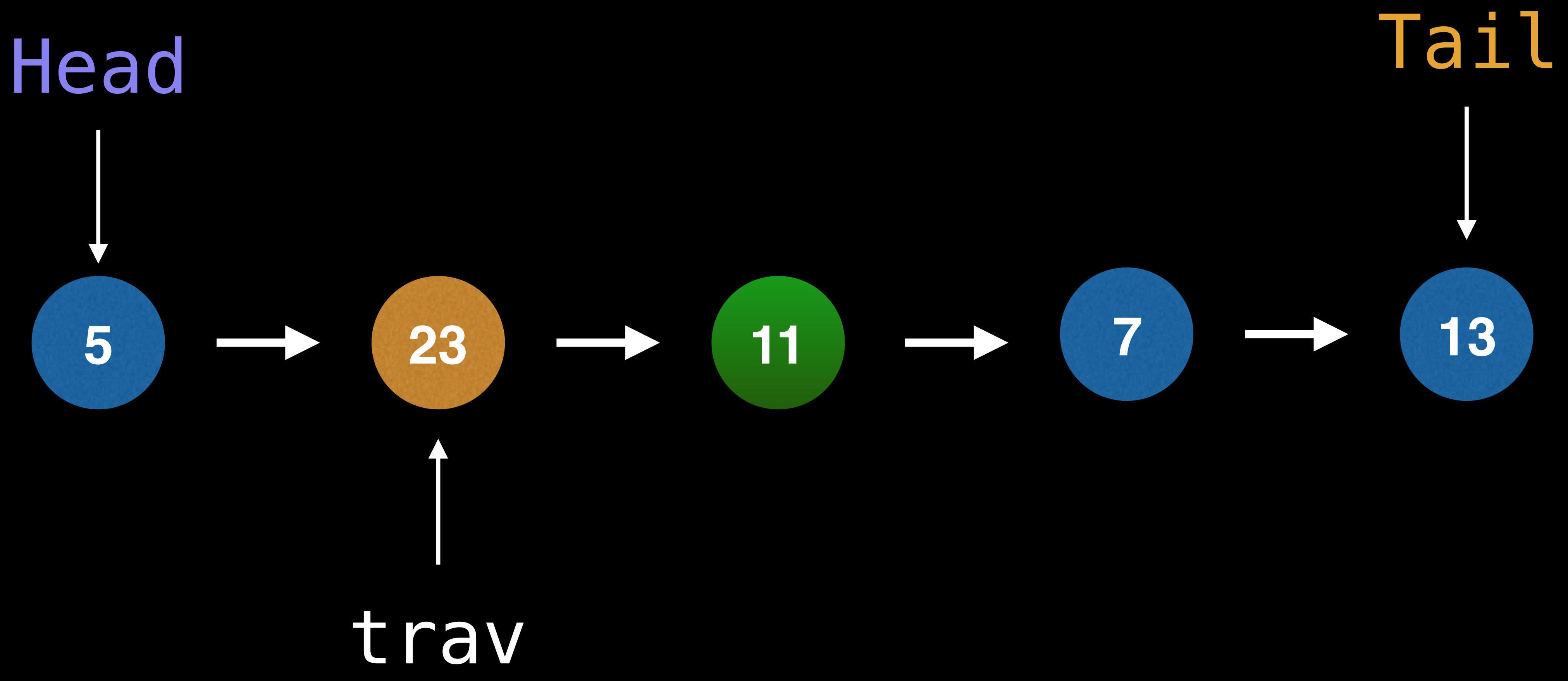
Inserting Singly Linked List

Insert 11 where the third node is.



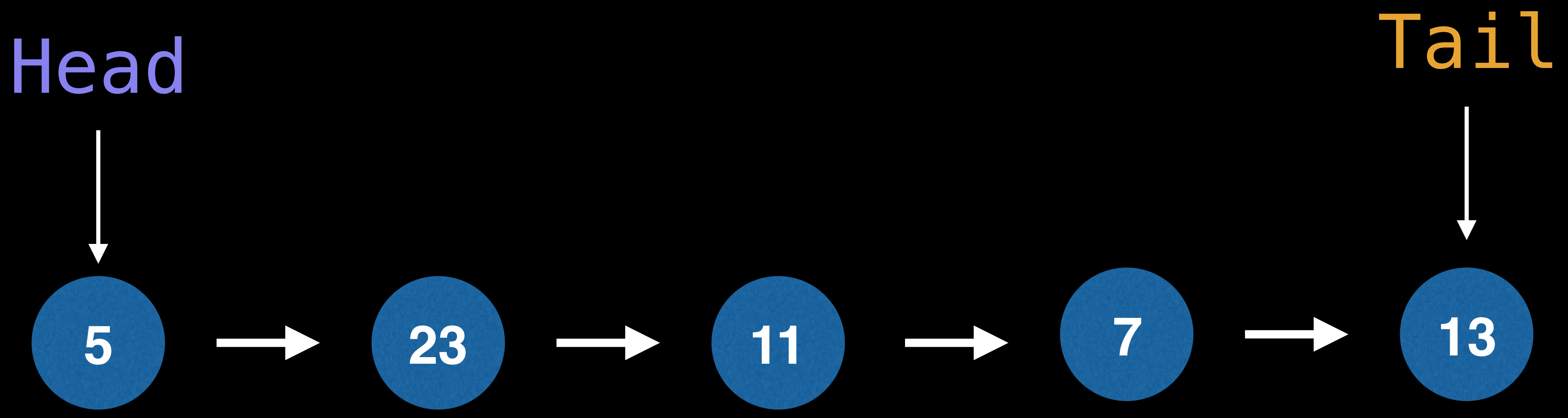
Inserting Singly Linked List

Insert 11 where the third node is.



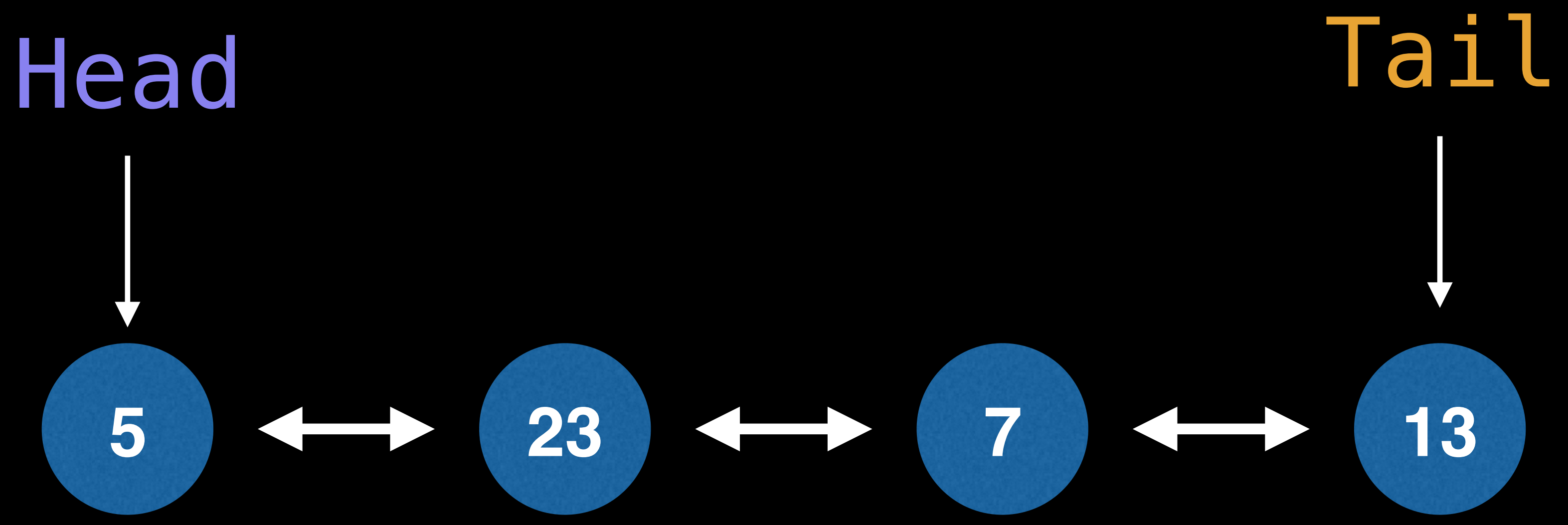
Inserting Singly Linked List

Insert 11 where the third node is.



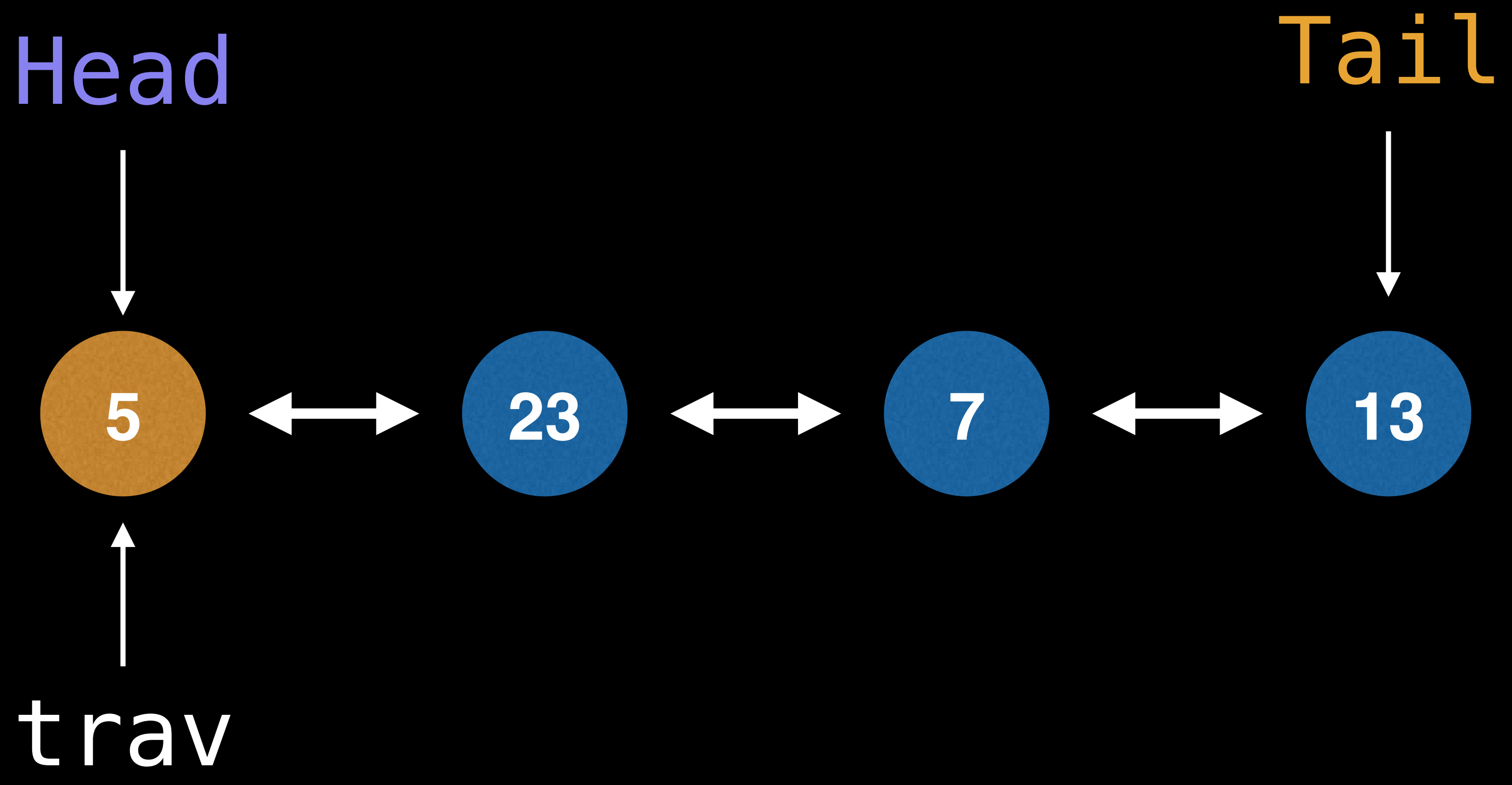
Inserting Doubly Linked List

Insert 11 where the third node is.



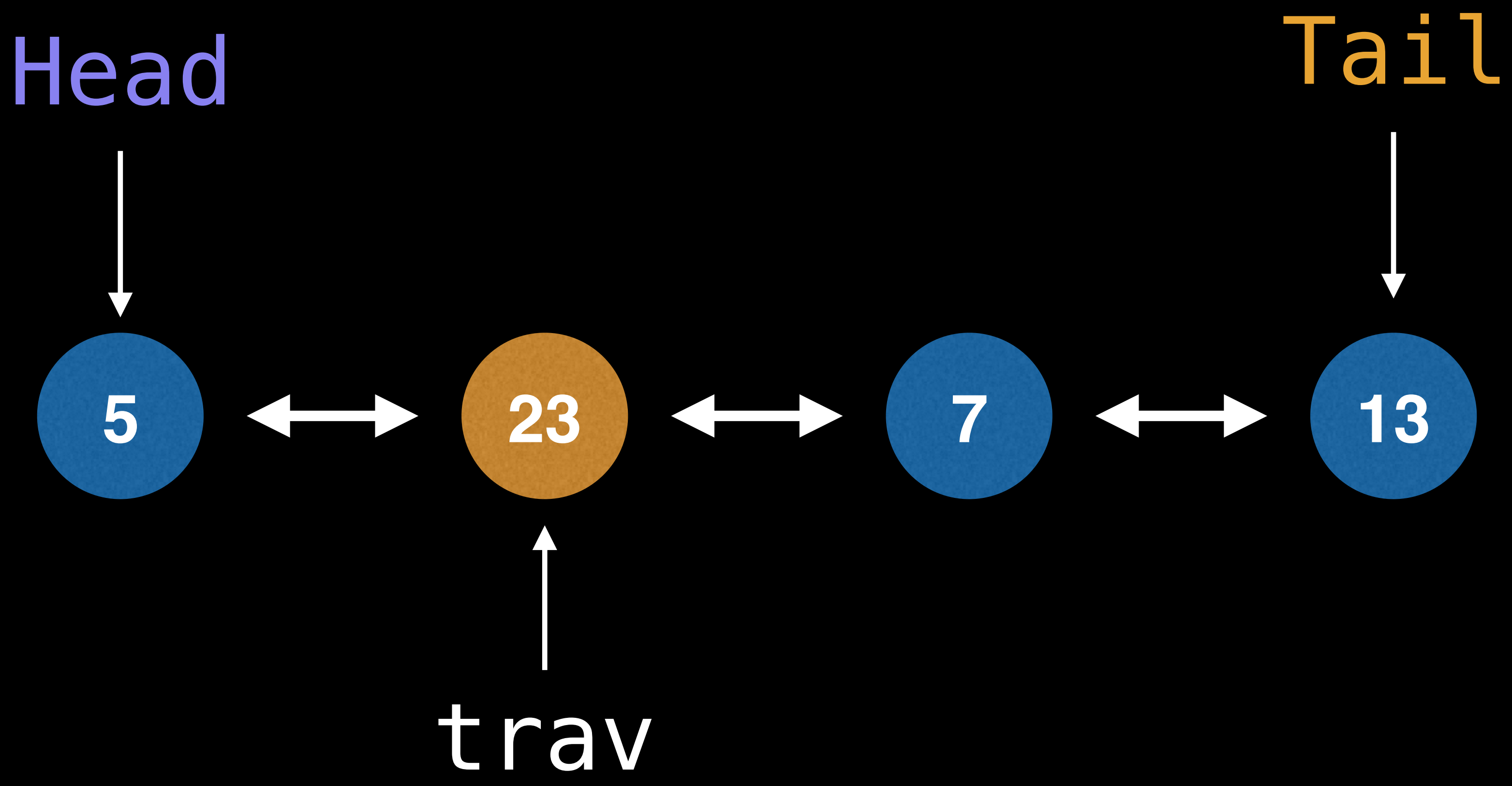
Inserting Doubly Linked List

Insert 11 where the third node is.



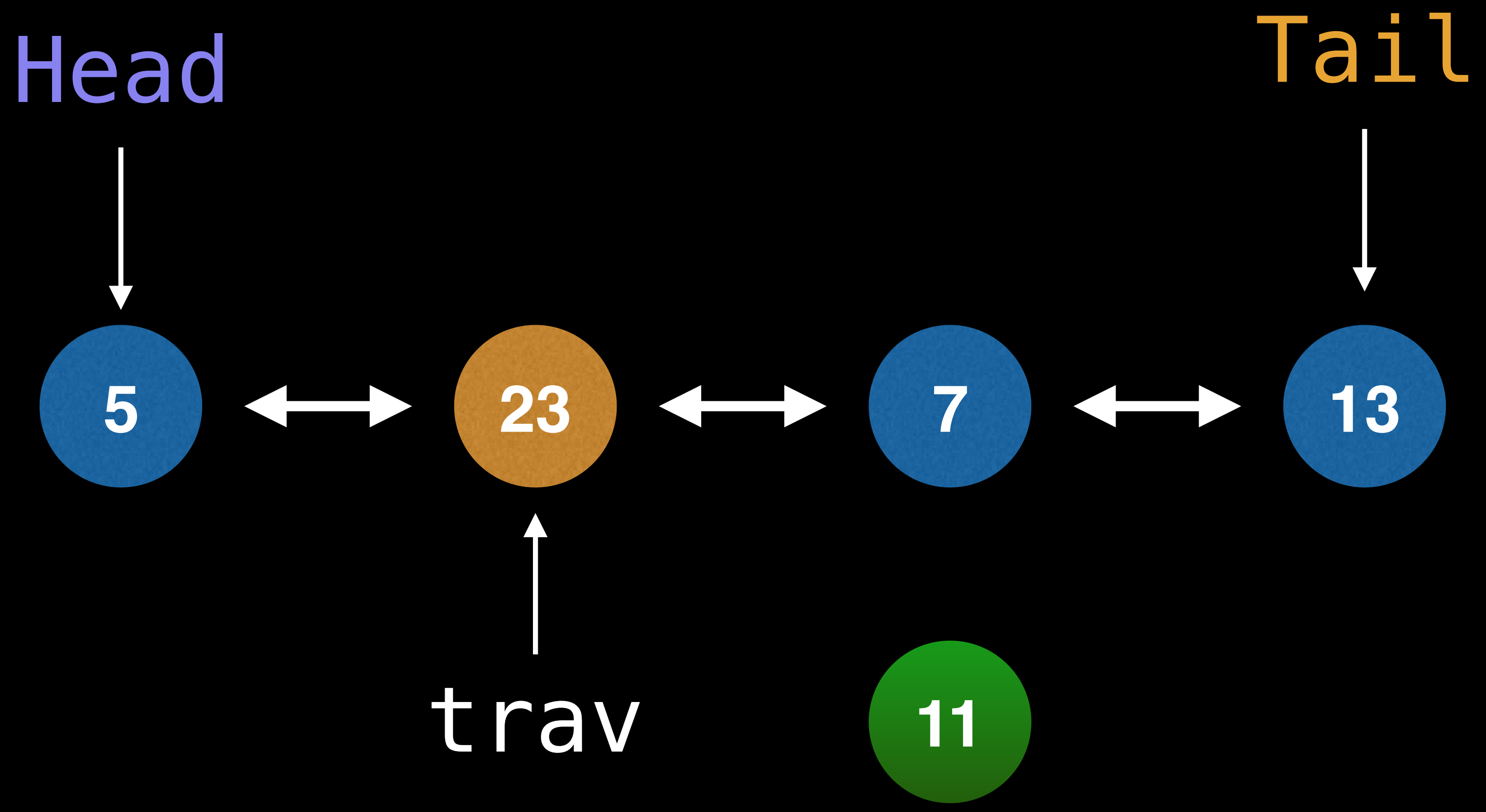
Inserting Doubly Linked List

Insert 11 where the third node is.



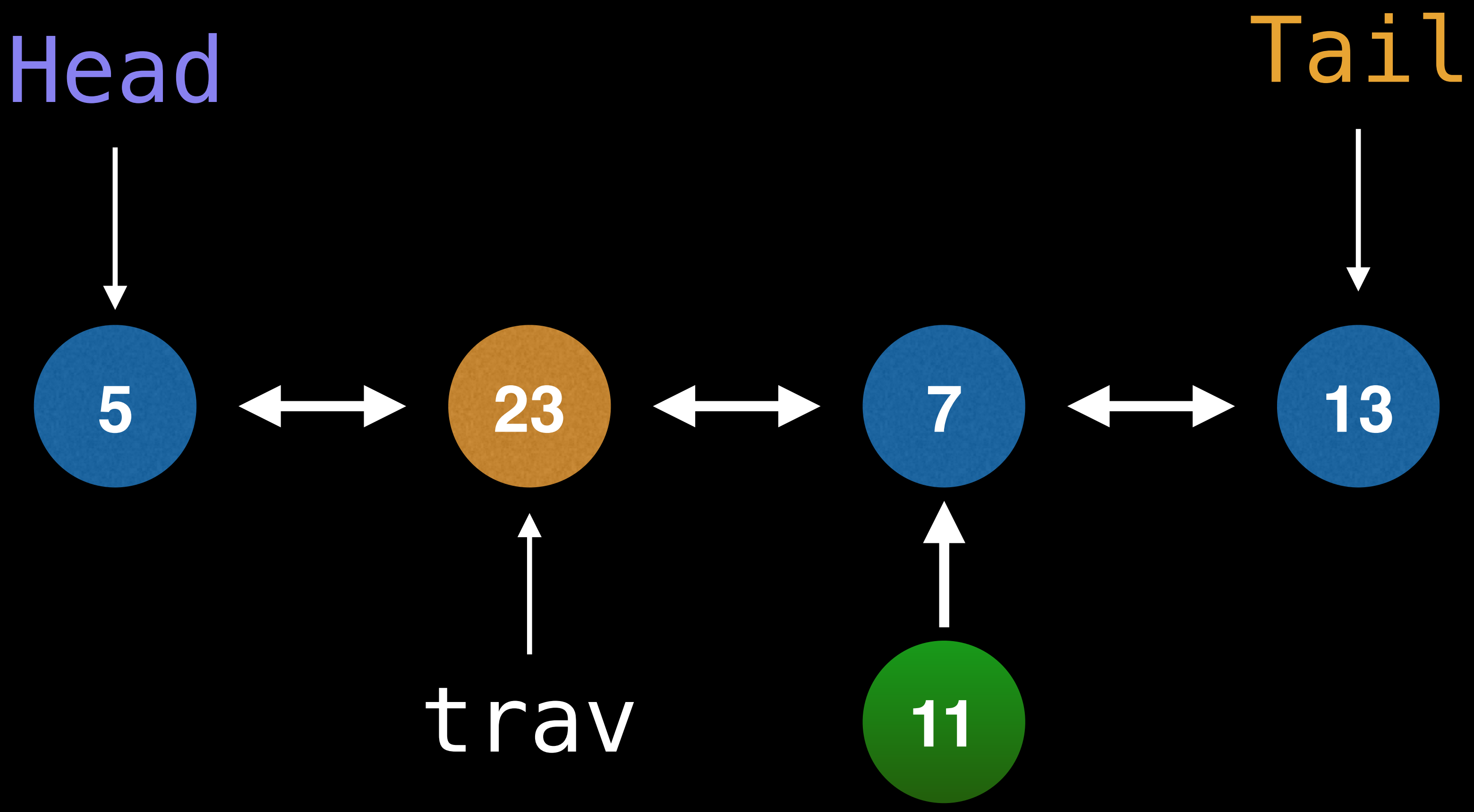
Inserting Doubly Linked List

Insert 11 where the third node is.



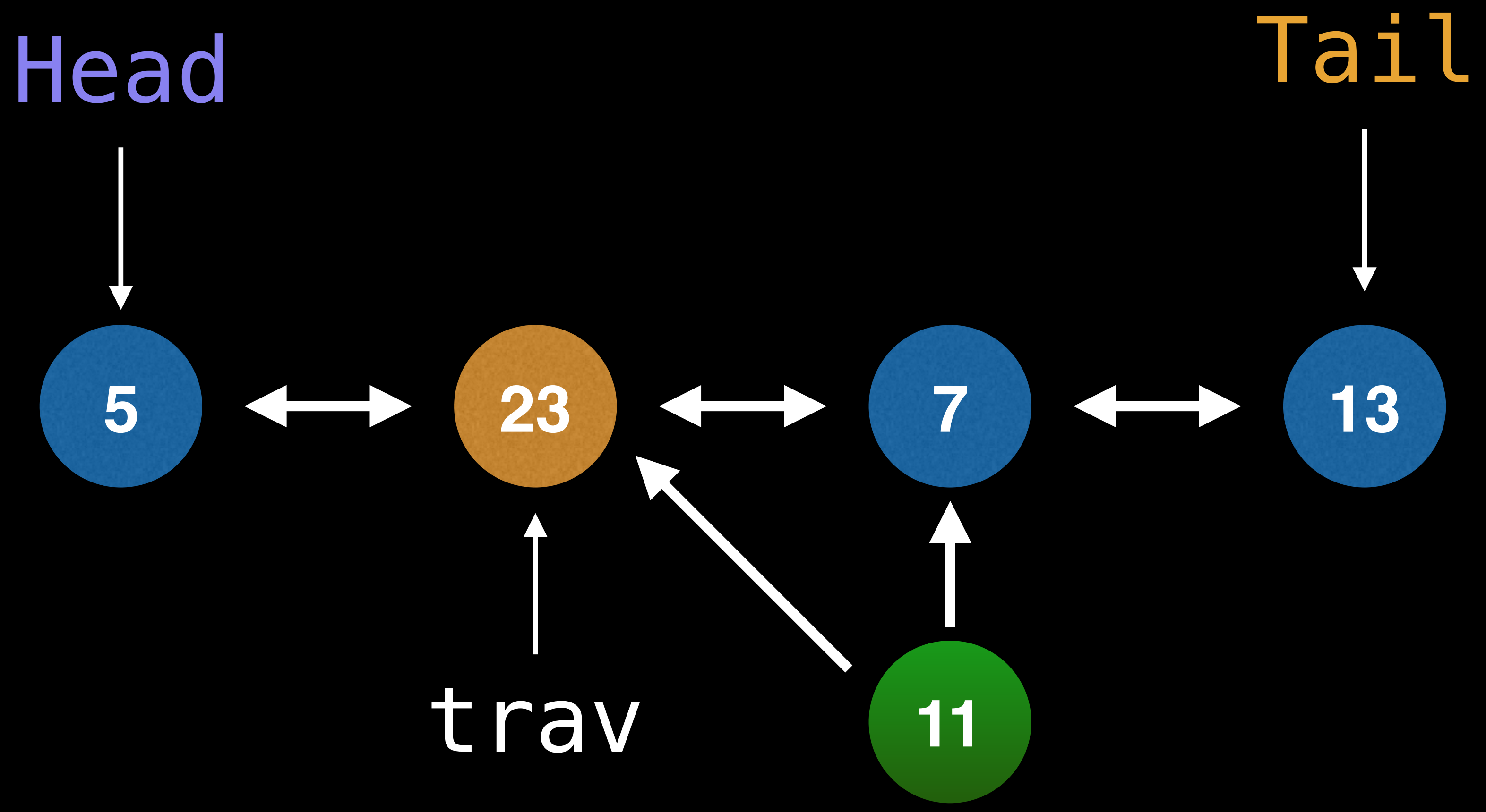
Inserting Doubly Linked List

Insert 11 where the third node is.



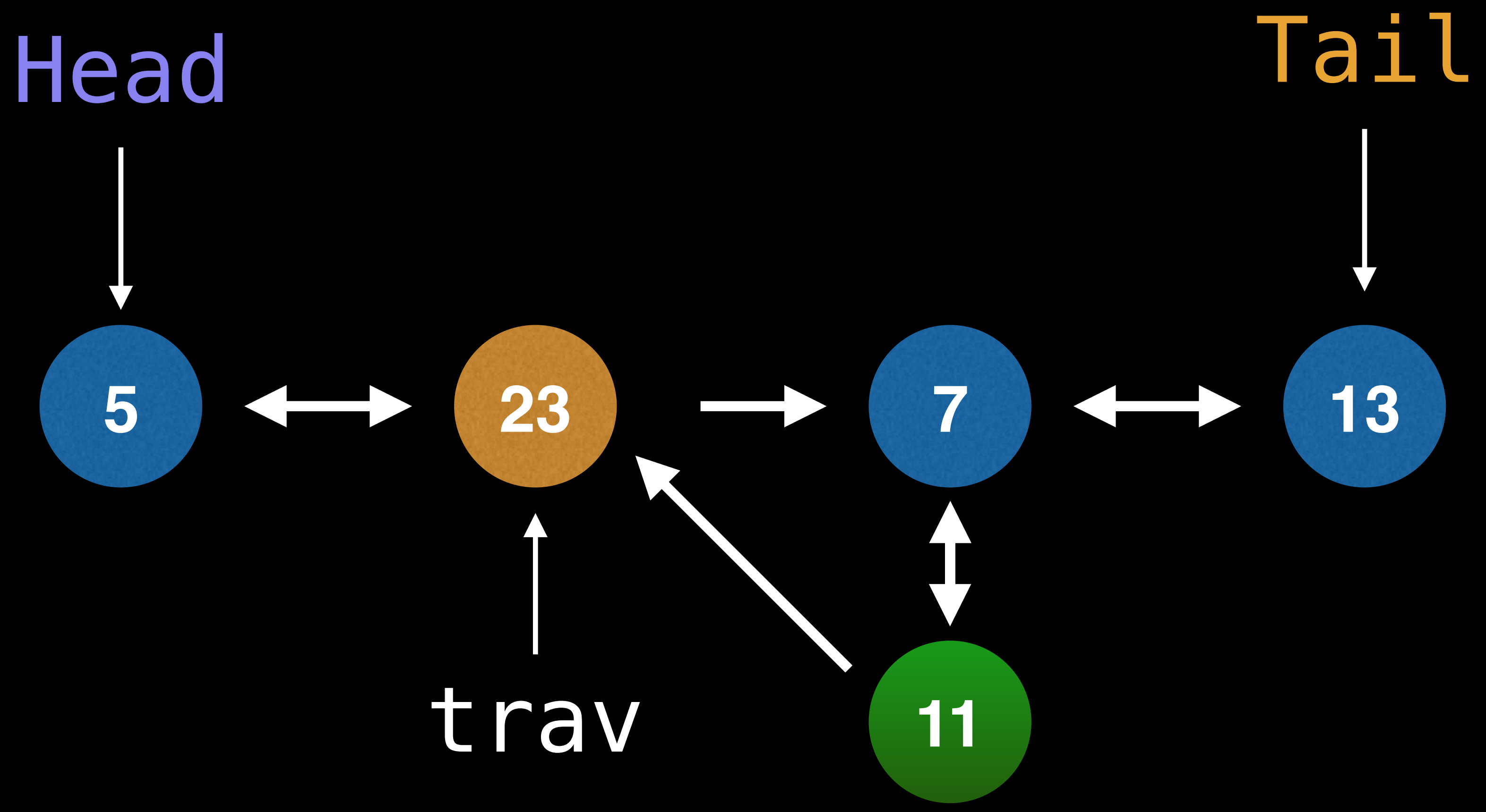
Inserting Doubly Linked List

Insert 11 where the third node is.



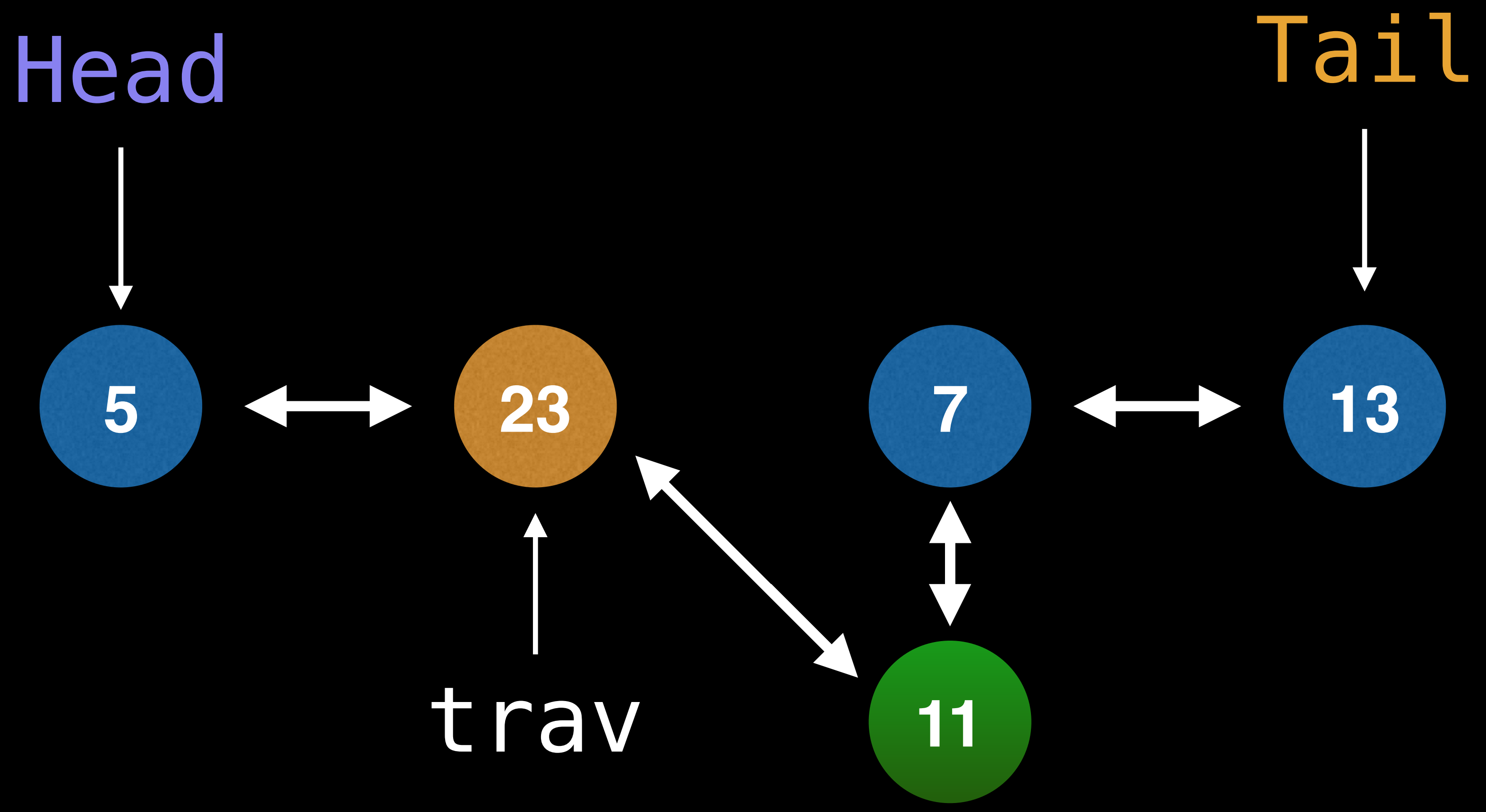
Inserting Doubly Linked List

Insert 11 where the third node is.



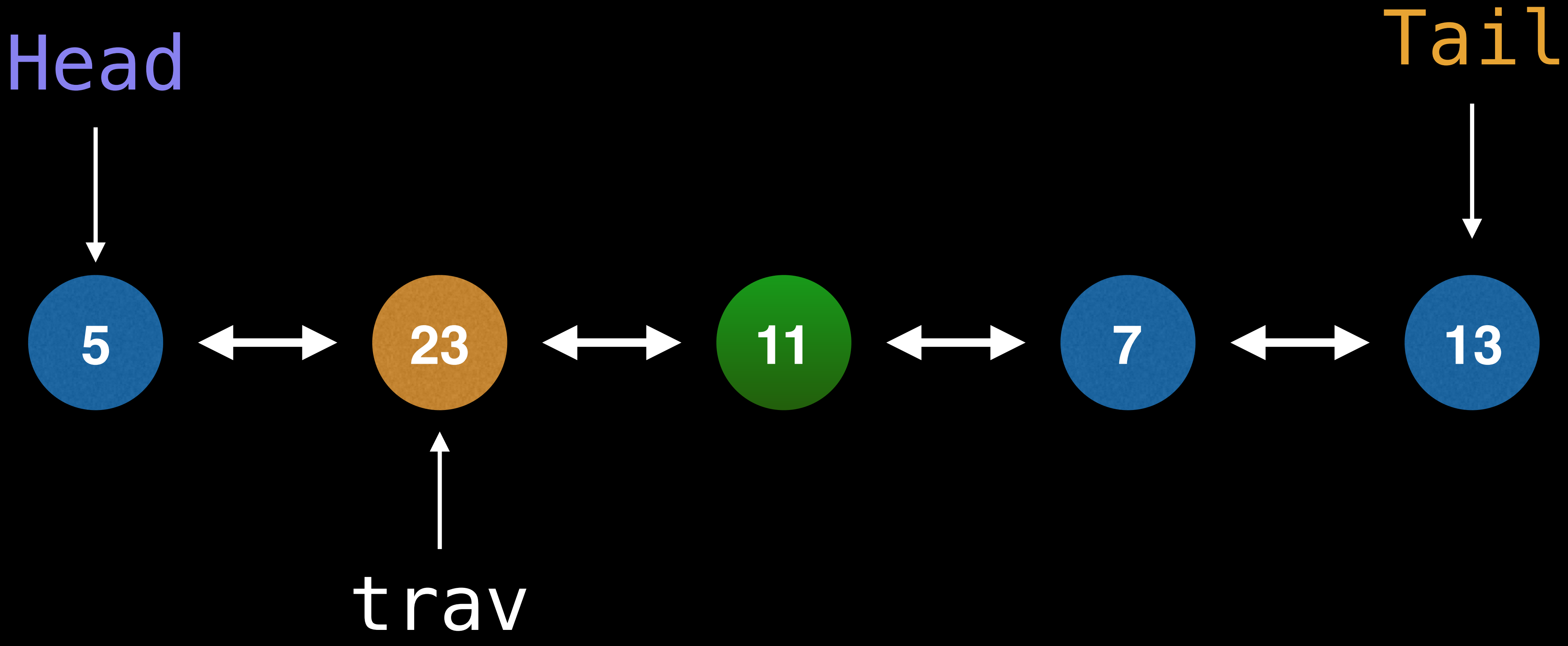
Inserting Doubly Linked List

Insert 11 where the third node is.



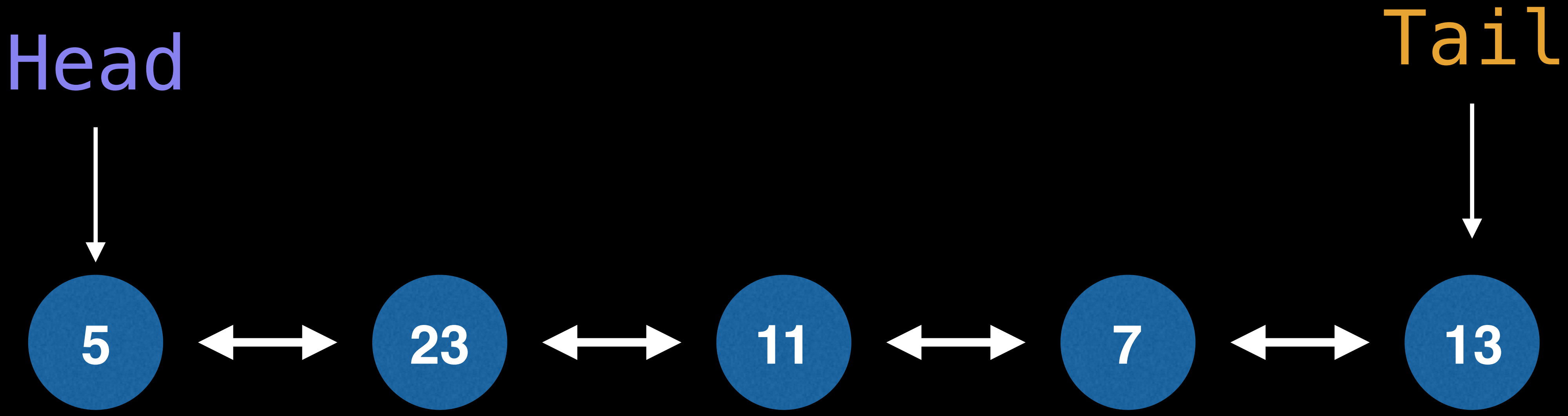
Inserting Doubly Linked List

Insert 11 where the third node is.



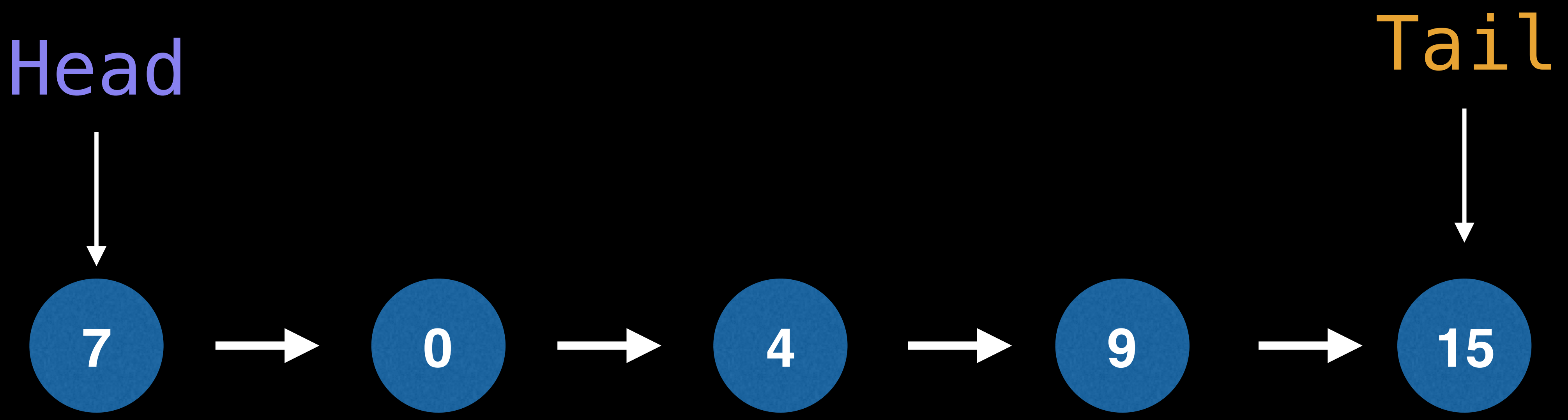
Inserting Doubly Linked List

Insert 11 where the third node is.



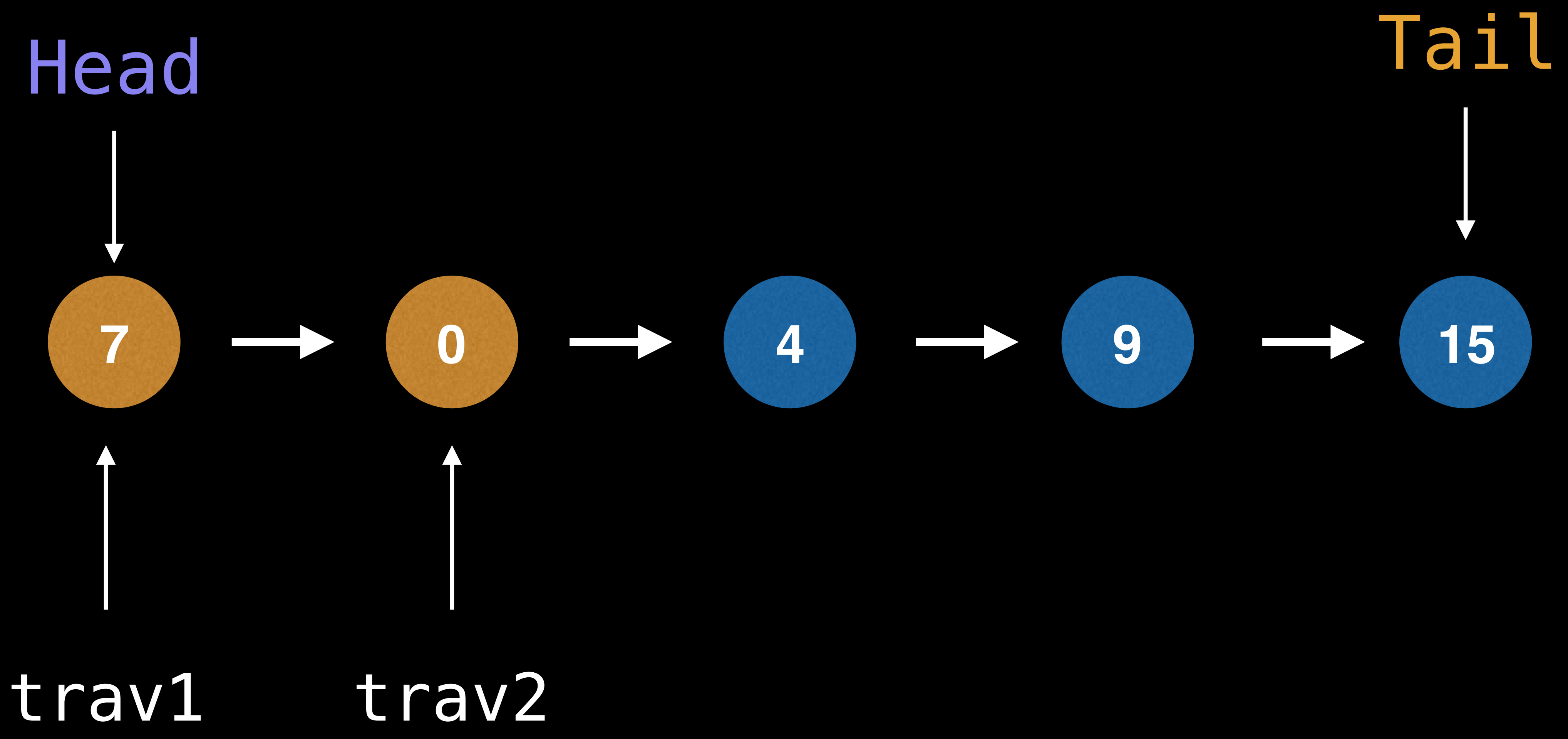
Removing from Singly Linked List

Remove 9 from the following SLL



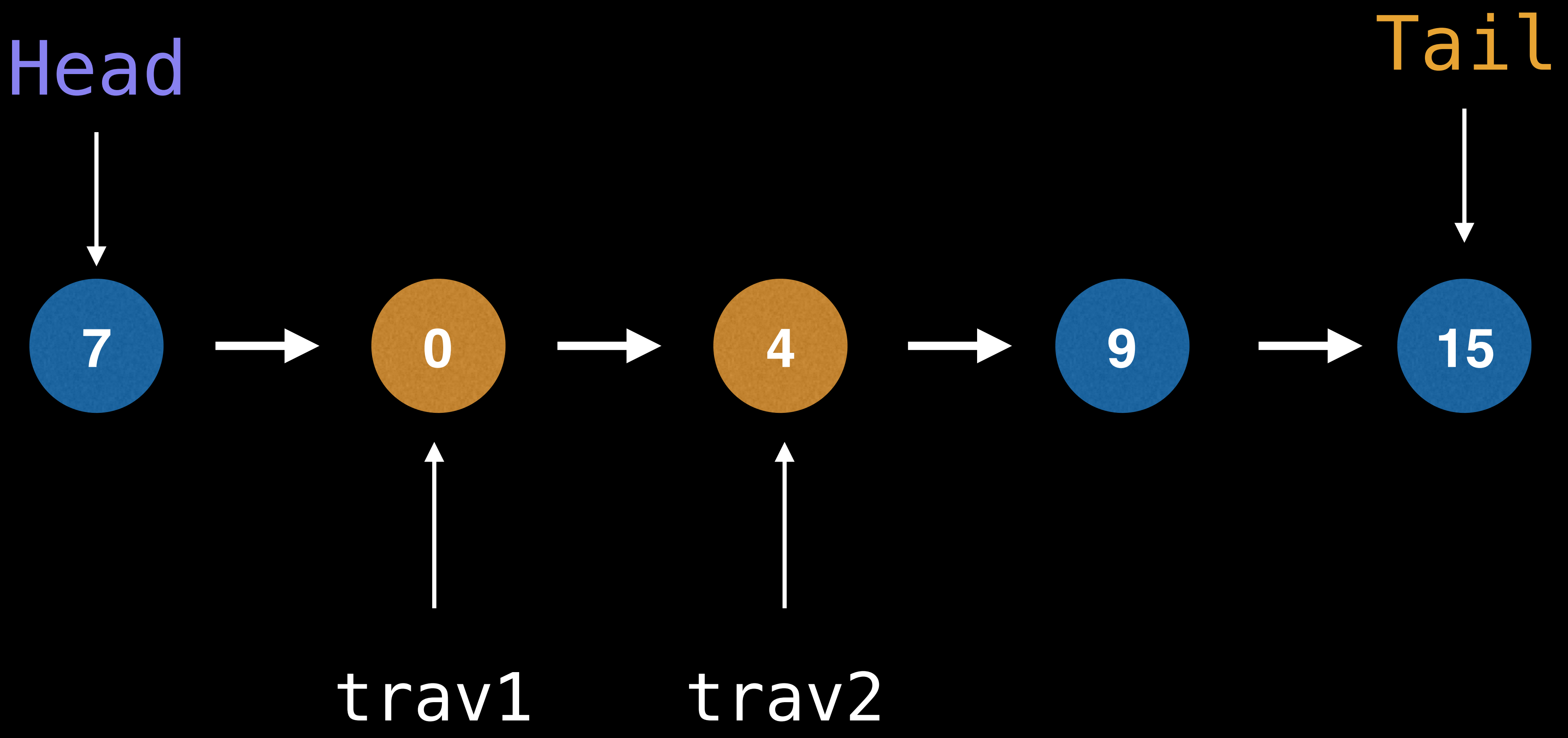
Removing from Singly Linked List

Remove 9 from the following SLL



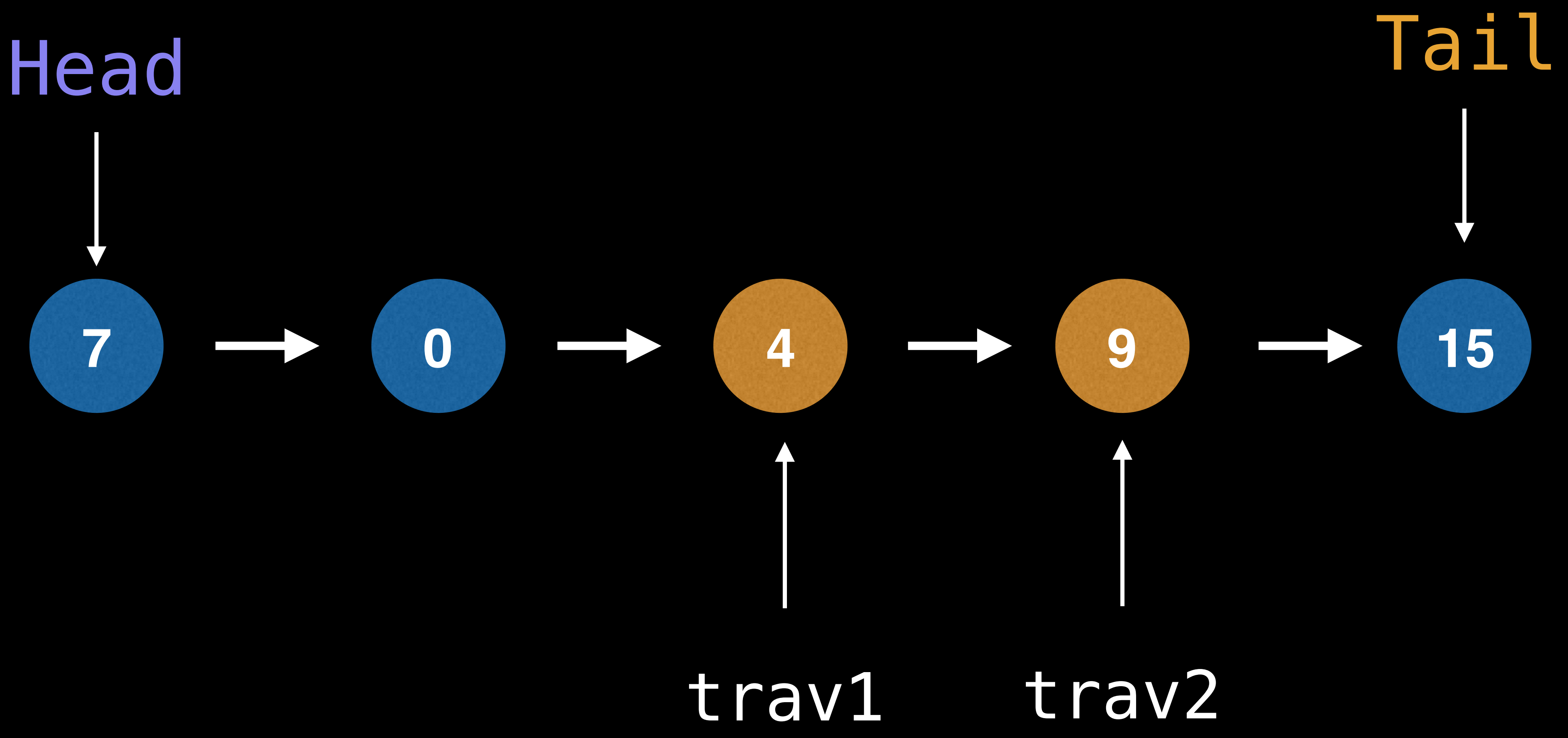
Removing from Singly Linked List

Remove 9 from the following SLL



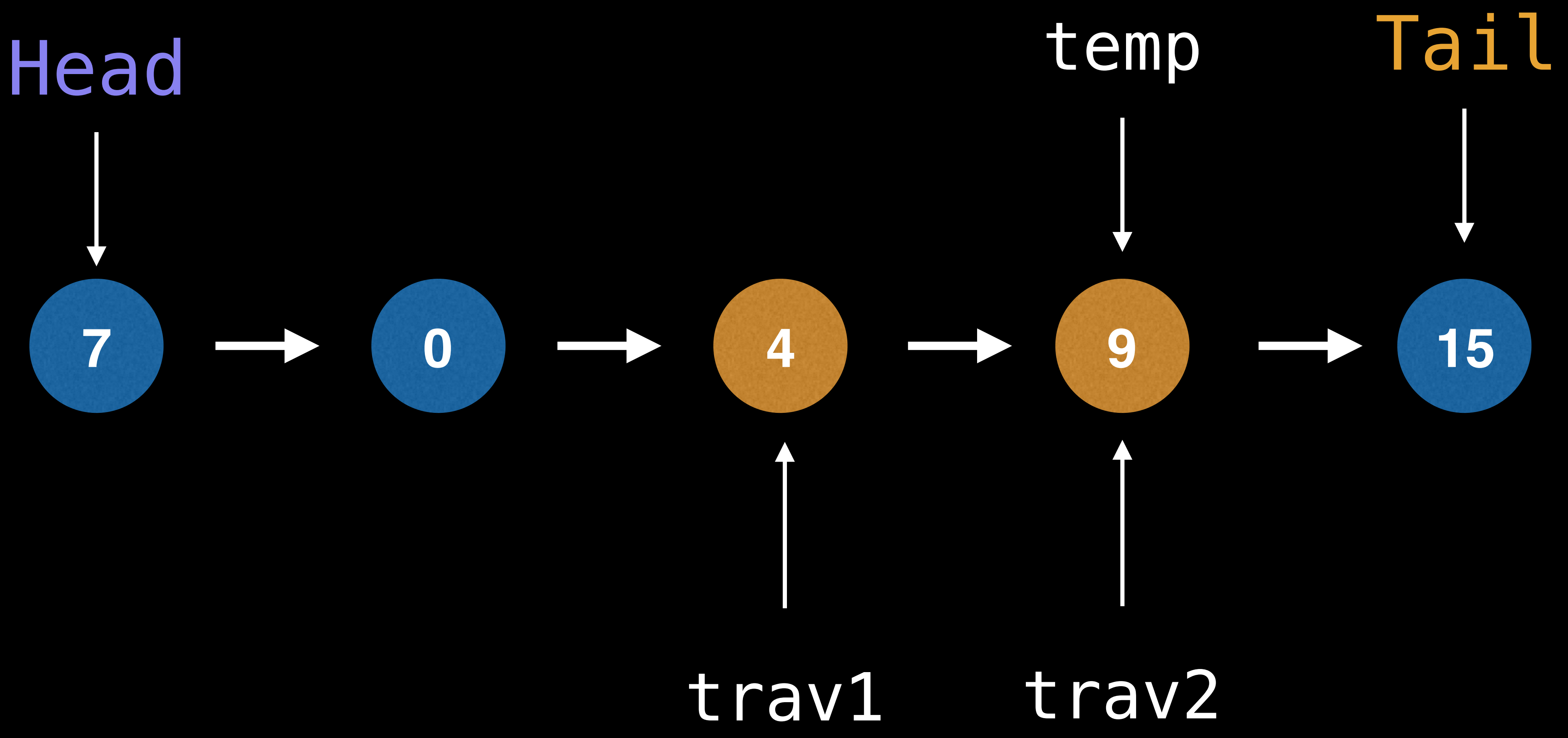
Removing from Singly Linked List

Remove 9 from the following SLL



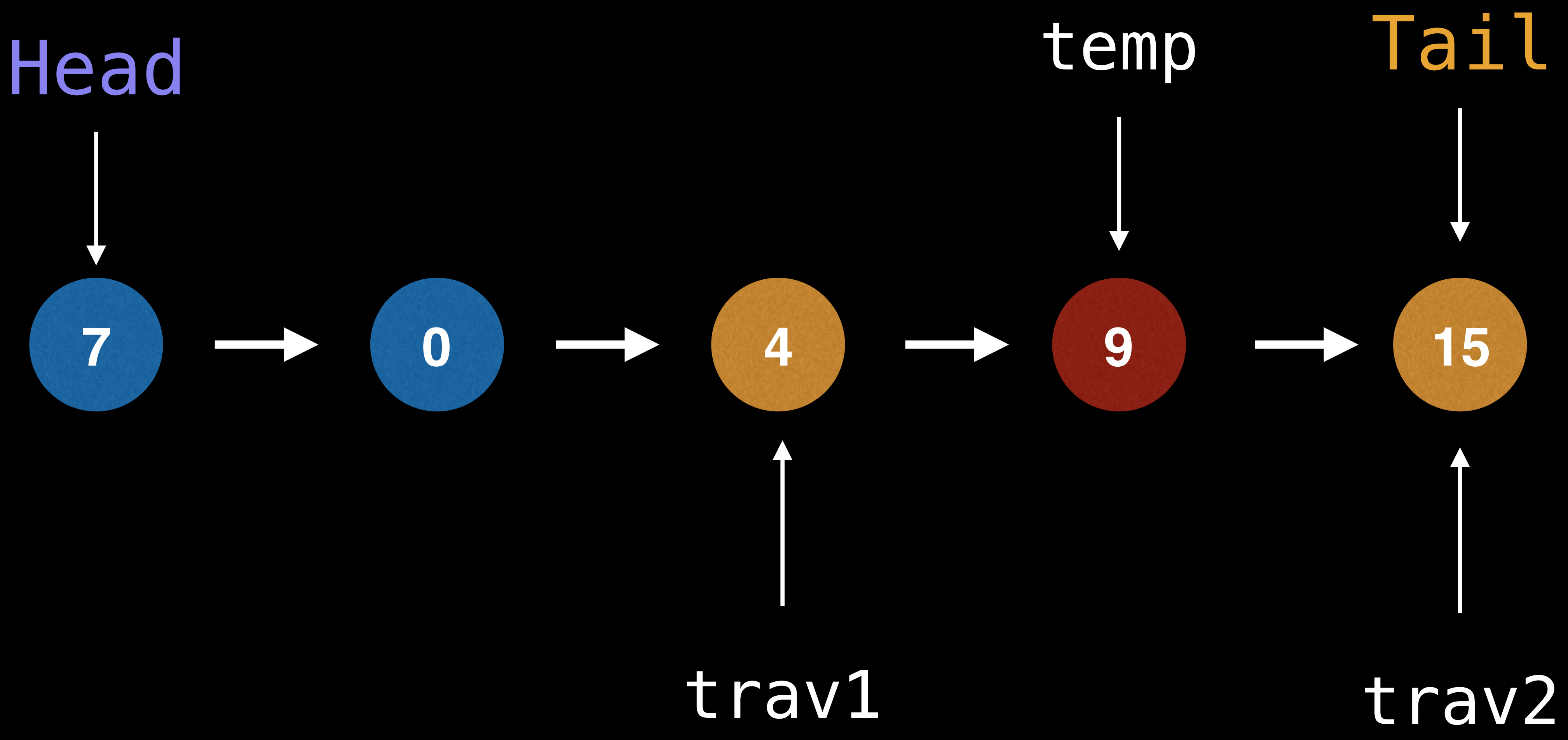
Removing from Singly Linked List

Remove 9 from the following SLL



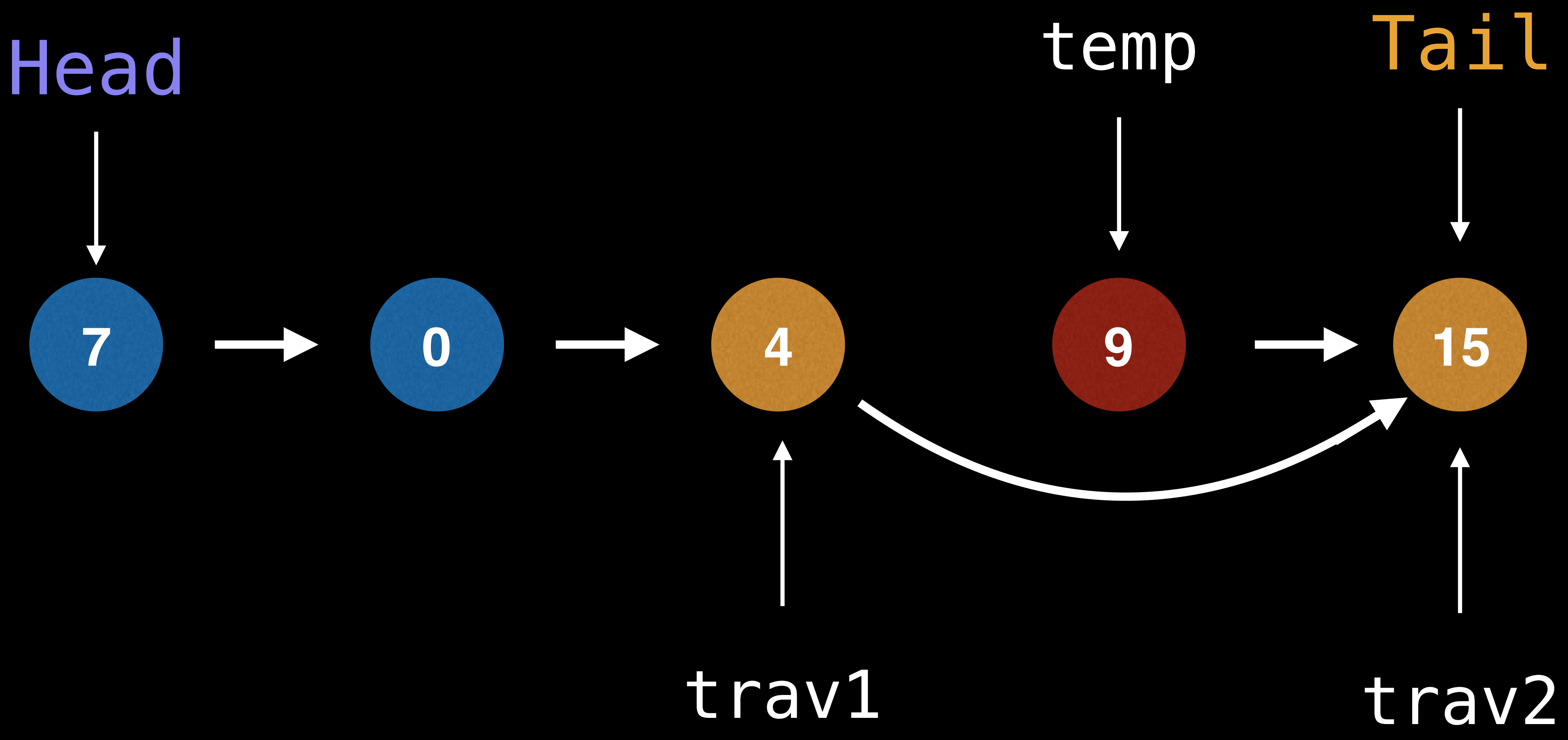
Removing from Singly Linked List

Remove 9 from the following SLL



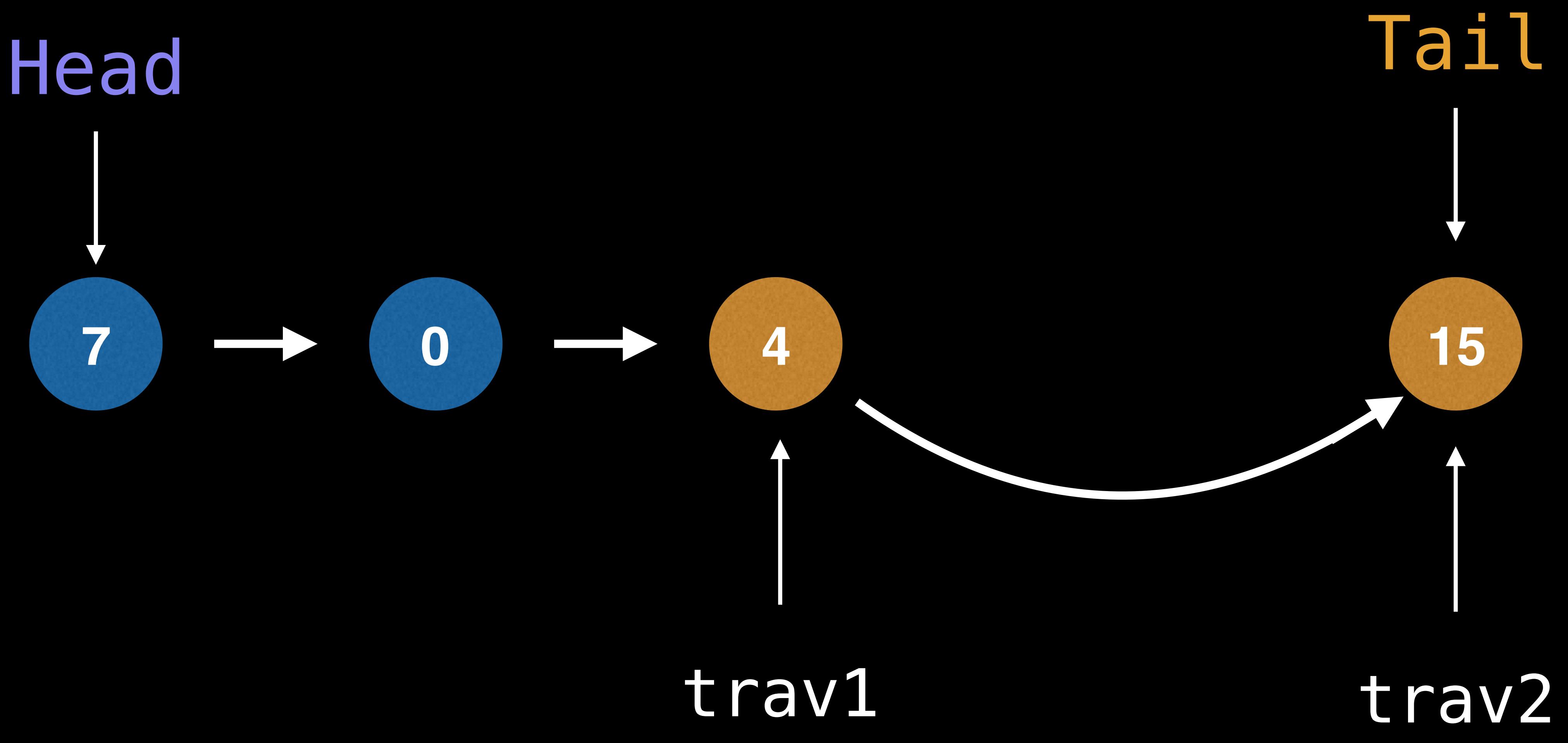
Removing from Singly Linked List

Remove 9 from the following SLL



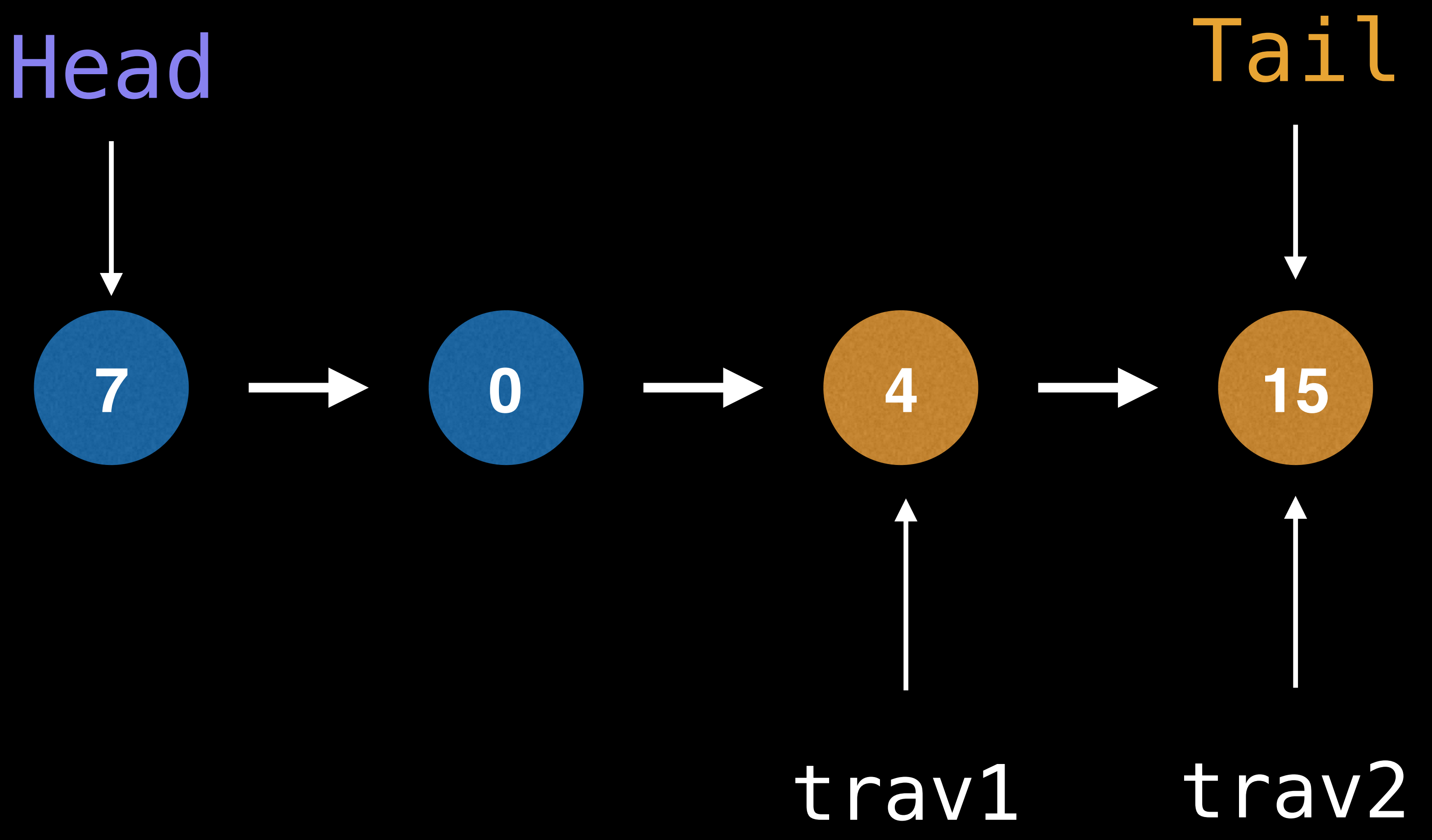
Removing from Singly Linked List

Remove 9 from the following SLL



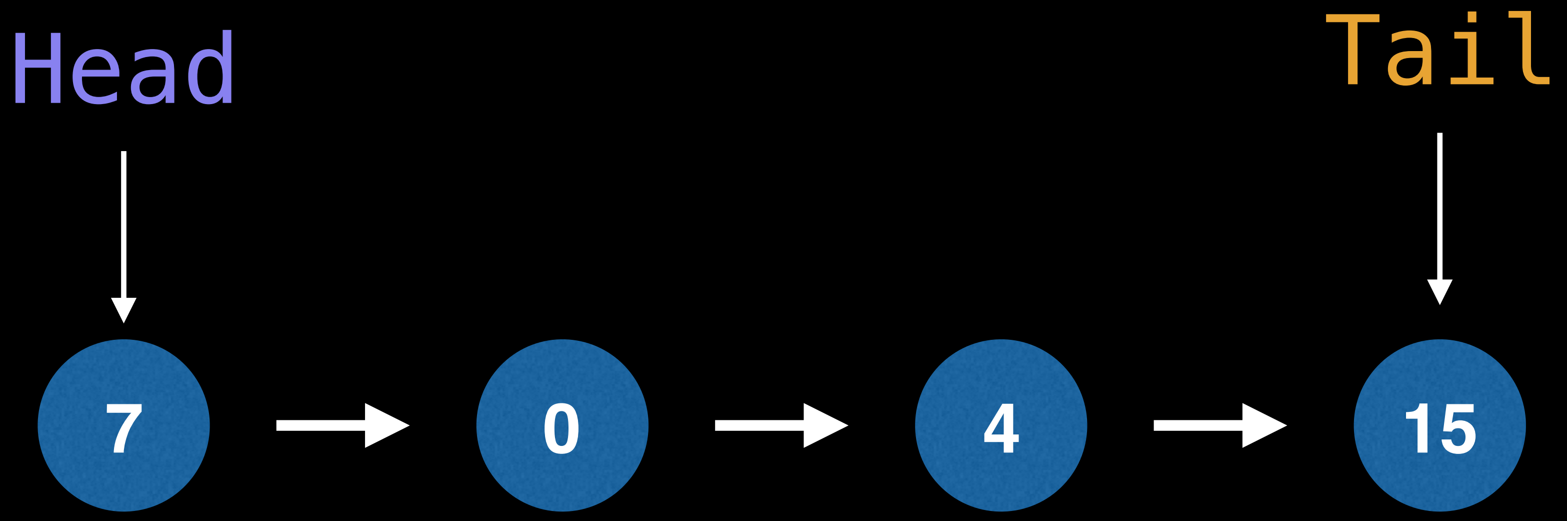
Removing from Singly Linked List

Remove 9 from the following SLL



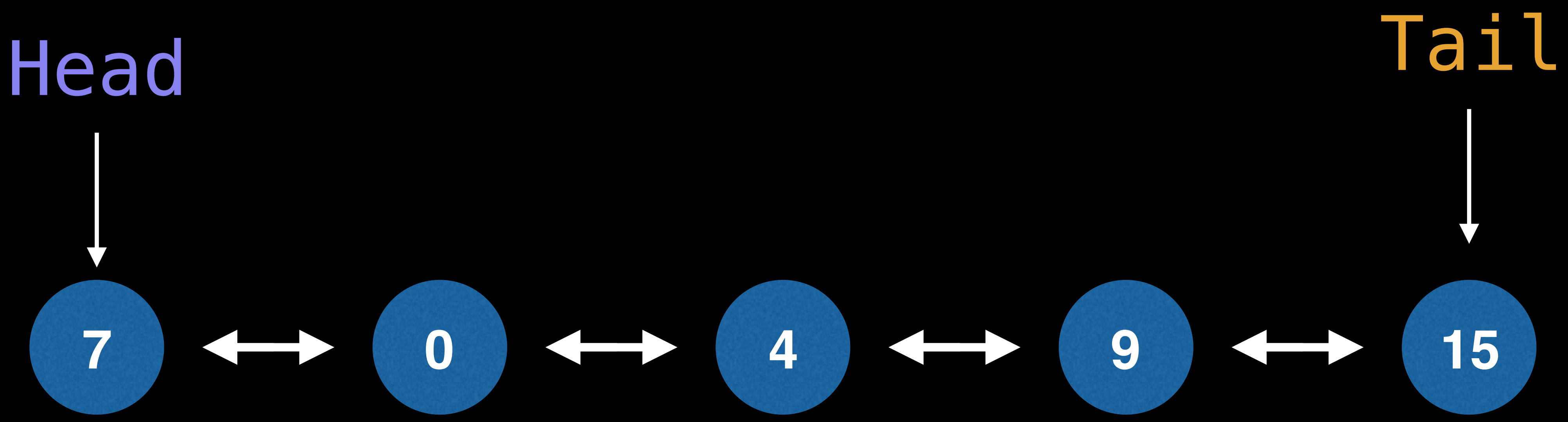
Removing from Singly Linked List

Remove 9 from the following SLL



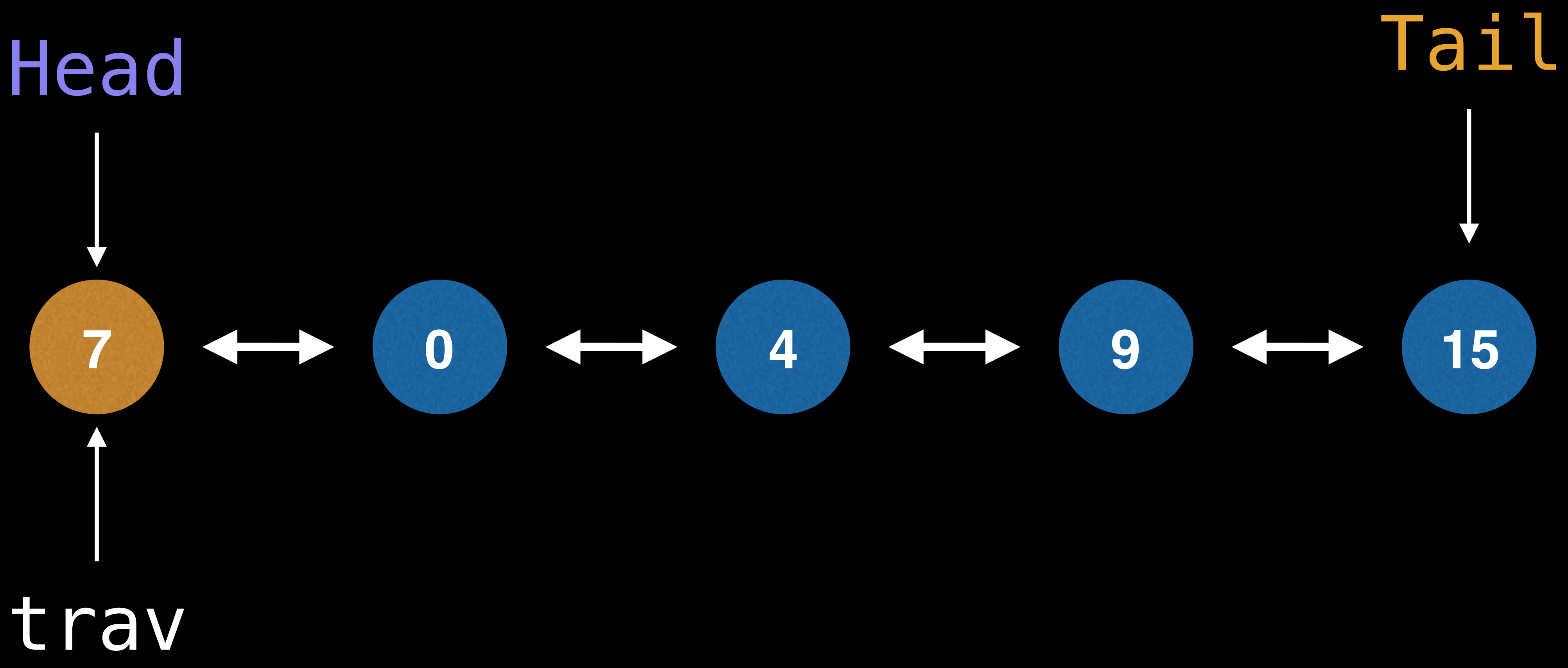
Removing from Doubly Linked List

Remove 9 from the following DLL



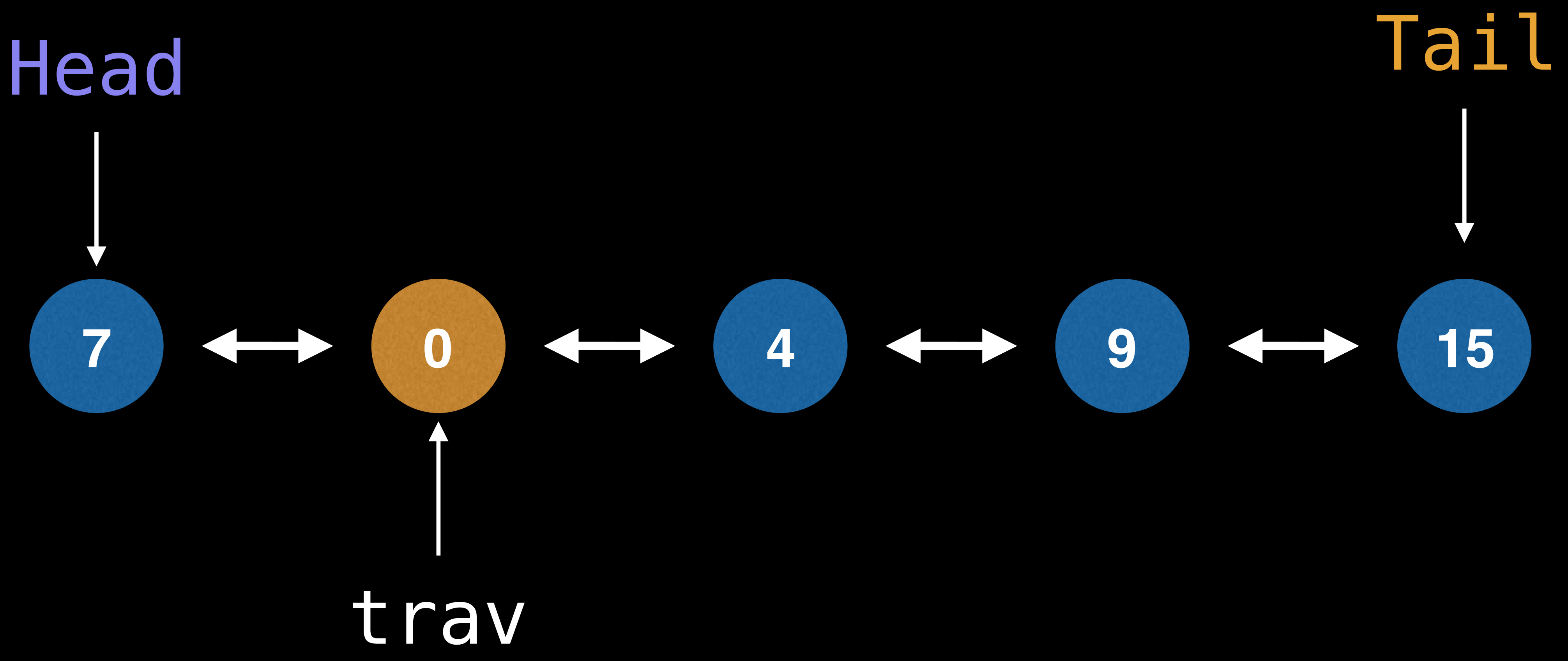
Removing from Doubly Linked List

Remove 9 from the following DLL



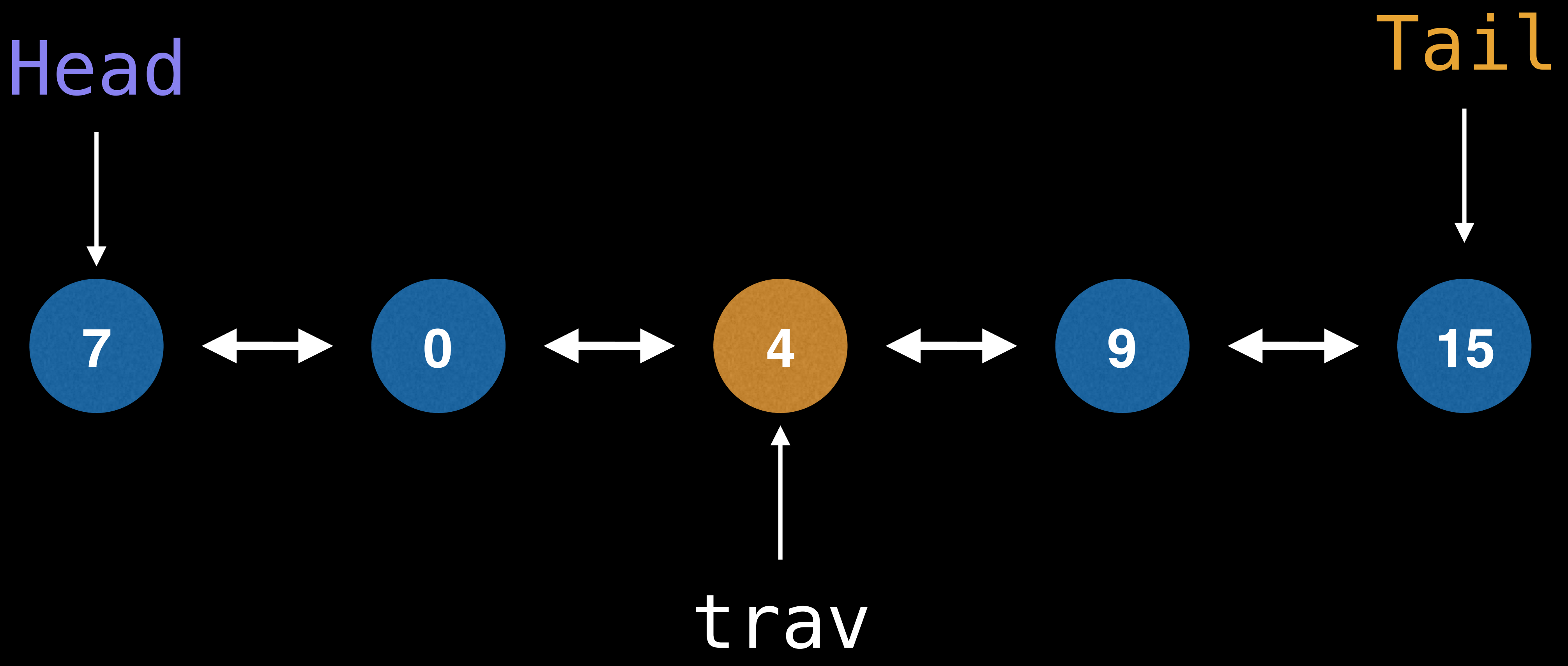
Removing from Doubly Linked List

Remove 9 from the following DLL



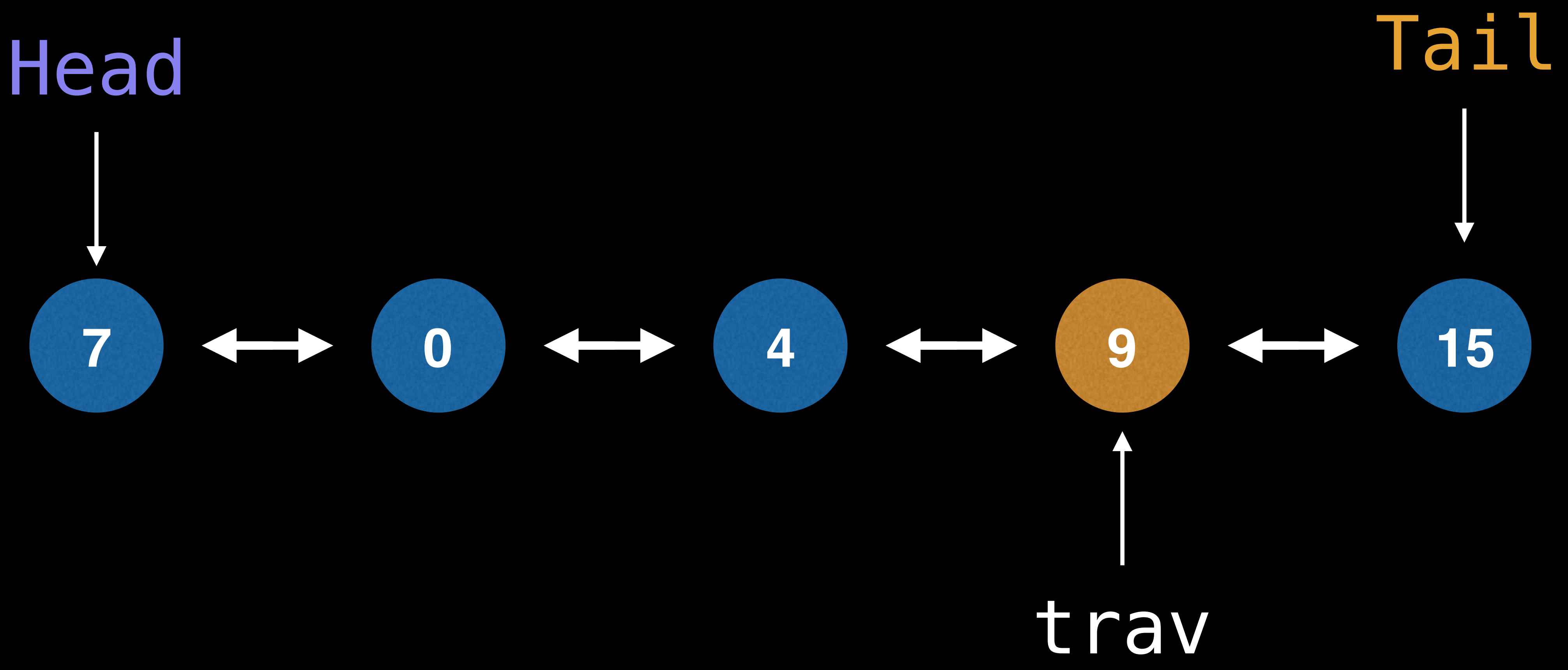
Removing from Doubly Linked List

Remove 9 from the following DLL



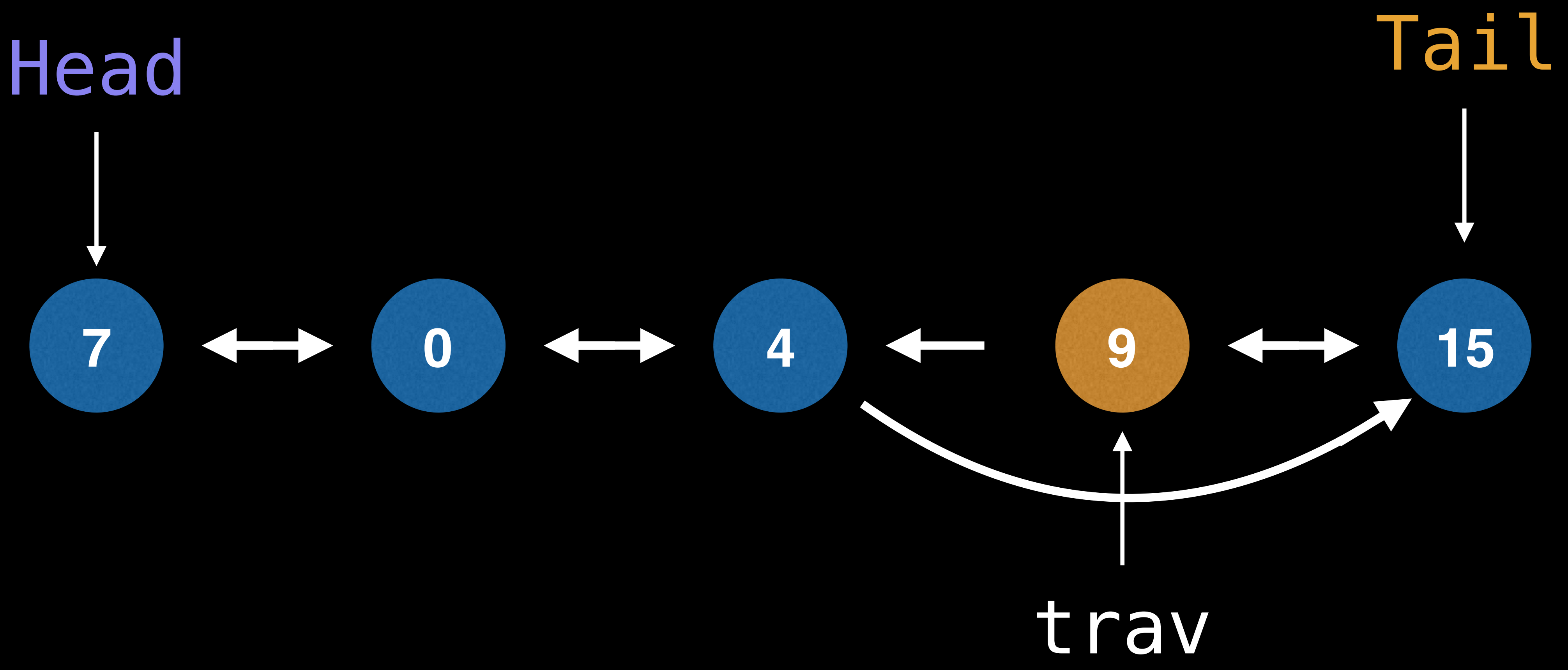
Removing from Doubly Linked List

Remove 9 from the following DLL



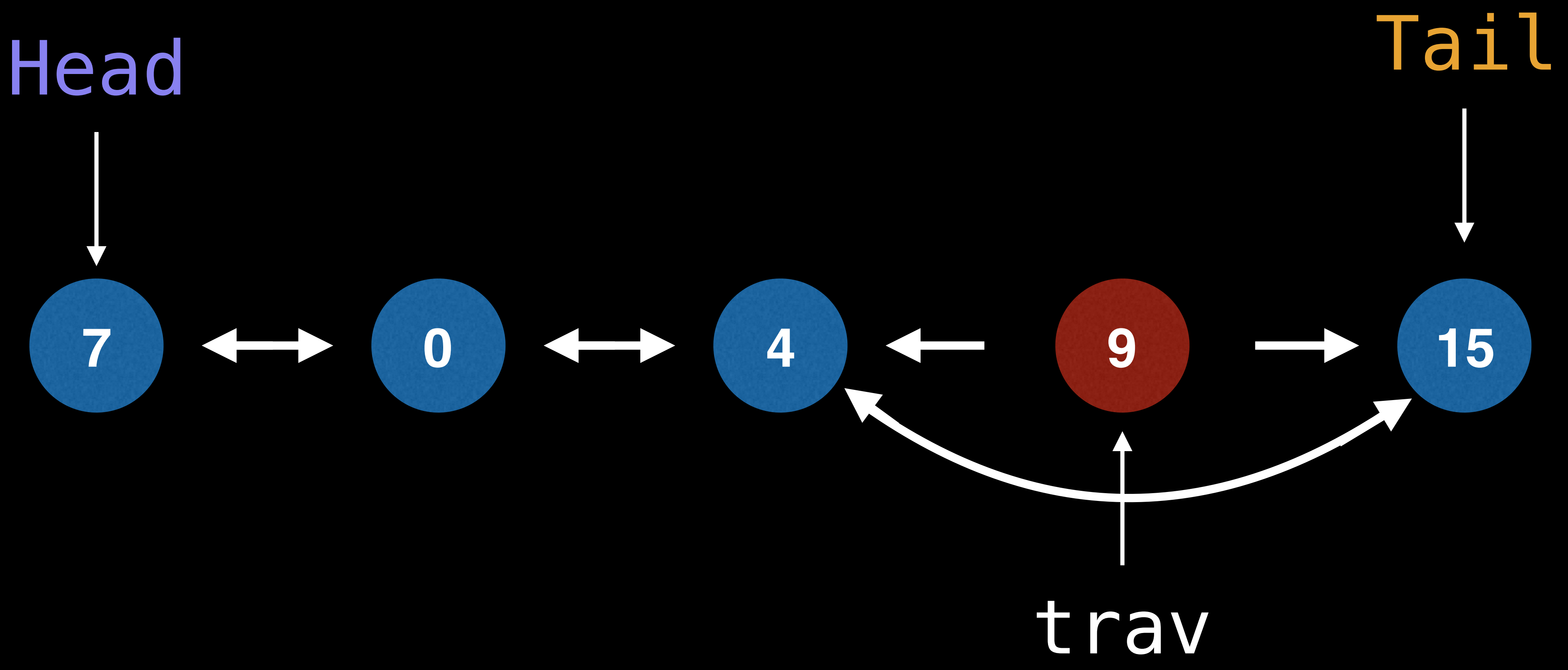
Removing from Doubly Linked List

Remove 9 from the following DLL



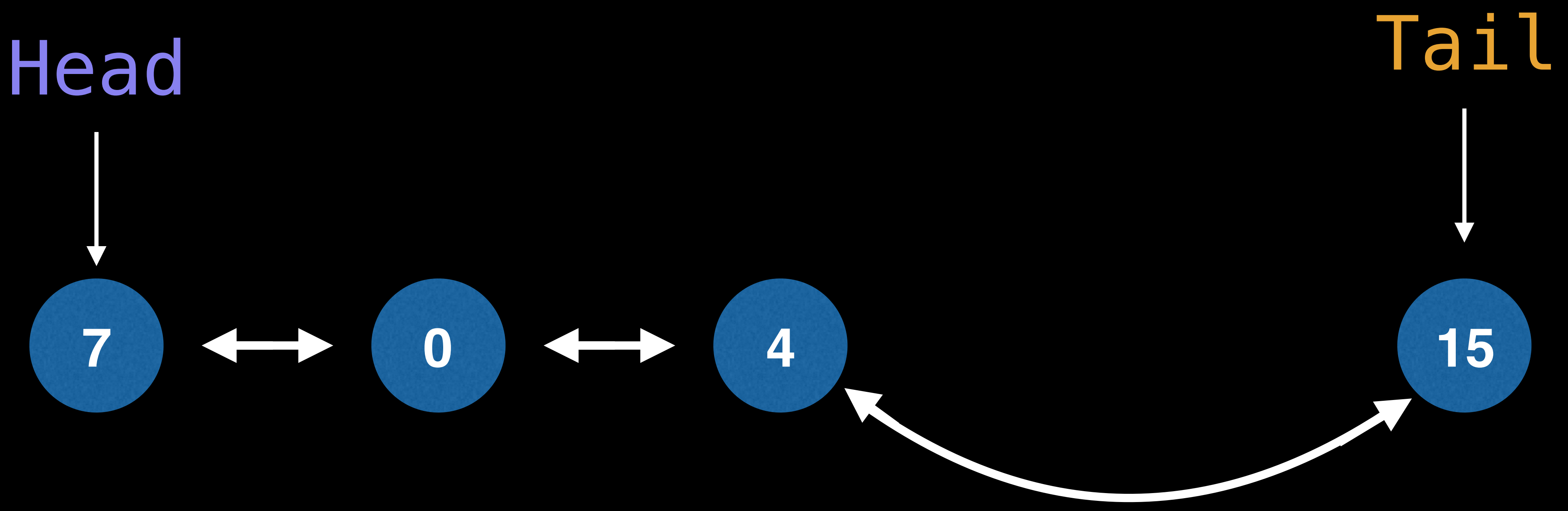
Removing from Doubly Linked List

Remove 9 from the following DLL



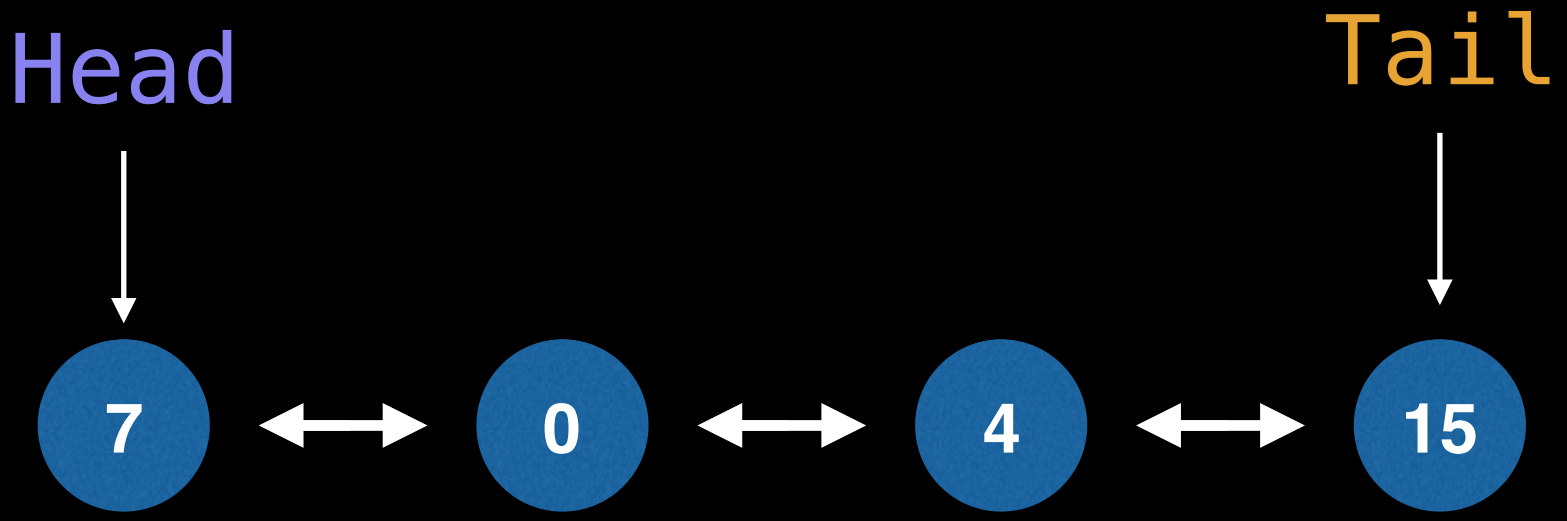
Removing from Doubly Linked List

Remove 9 from the following DLL



Removing from Doubly Linked List

Remove 9 from the following DLL



Complexity Analysis

Complexity

	Singly Linked	Doubly Linked
Search	O(n)	O(n)
Insert at head	O(1)	O(1)
Insert at tail	O(1)	O(1)

Complexity

	Singly Linked	Doubly Linked
Remove at head	O(1)	O(1)
Remove at tail	O(n)	O(1)
Remove in middle	O(n)	O(n)