Linked List Examples
Use only one iteration, find n’th node from the end of a Linked List

![Diagram of a linked list with the head marked and the n=4 node indicated.](image-url)
Use only one iteration, find n’th node from the end of a Linked List

- Two pointers p1=head; p2= head
- First move p1 to n nodes from head.
- Now move both pointers one by one until p1 reaches end.
- P2 stops at target
Find the intersection point of two Linked Lists
Find the intersection point of two Linked Lists

n1 = length(l1)
n2 = length(l2)
d = |l1 - l2|
Move p1 d times
Move both p1 and p2 until they meet
Find the split point of two Linked Lists
Detect if there is a cycle in a Singly Linked List
boolean hasLoop(Node first) {
    if (first == null)  return false;
    Node slow, fast; // create two references.
    slow = fast = first;
    while (true) {
        slow = slow.next; // 1 hop.
        if (fast.next != null)
            fast = fast.next.next; // 2 hops.
        else
            return false; // next node null => no loop.
        if (slow == null || fast == null) // if either hits null..no loop.
            return false;
        if (slow == fast) // if the two ever meet...we must have a loop.
            return true;
    }
}
Remove the cycle in a Singly Linked List
Clone a linked list with next and random pointer