

Delete Tail

Given a pointer to the root of the linklist, your task is to delete the tail of the linklist and return the head of the updated list.

Input

1->2->3->4->5->6->7->8

Output

1->2->3->4->5->6->7

Solution: deleteTail.cpp

Kth Last Element

Implement a function that returns the Kth last of element from the linked list, in a single pass. You can assume K will be less than / equal to length of linked list.

(Hint : Use two pointers similar to Runner Technique)

Input

You will get head of the linked list.

1 -> 2 -> 3 -> 4 -> 5 ->6 ->7

K = 3

Output

5

Solution: kthLastElement.cpp

Alternate Merge

Given two linked lists, insert nodes of second list into first list at alternate positions of first list.

Input Format

In the function pointer to the start of the two linklists is passed.

Output Format

Return a pointer to the new merged list.

Sample Input

5->7->17->13->1112->10->2->4->6

Sample Output

5->12->7->10->17->2->13->4->11->6

Solution: alternateMerge.cpp

Bubble Sort on Linked List

Given a singly linked list, sort it using [bubble sort](#).

Input: 10->30->20->5

Output: 5->10->20->30

Solution: linkedListBubbleSort.cpp