Problem 5: Write a (C) program to create a *dynamic list* of integers. Unlike array, the size of the list is not fixed at the beginning and this list can grow dynamically. Memory needs to be allocated while inserting a new element into the list.
Input the elements to insert into list. Walk the list and print the integers.
Next, delete all the odd valued integers. Print the new list (remember it may end up being a NULL list).
Note: You need to create a `node structure` which has at least two attributes: one to store the integer value, another as a pointer to a neighbouring node that contains the next integer and so on. The last node points to a NULL pointer indicating end of the list. Continue to read the elements to insert into list until the input received is equal to `-999`.

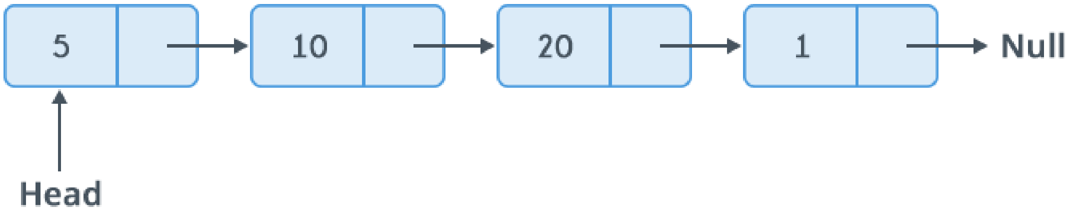

Figure 1: Dynamic list of 4 nodes

Sample Input 1:
3 6 5 9 13 -999
Output 1:
3 6 5 9 13
6

Sample Input 2:
47 65 23 91 -999
Output 2:
47 65 23 91
NULL