22TH JAN 2023

FIRST :-

Delete all occurrences of a given key in a doubly linked list : : Medium

You are given the head of a doubly Linked List and a Key. Your task is to delete all occurrences of the given key and return the new DLL.

Example:

```
Input:
2<->2<->10<->8<->4<->2<->5<->2
2
Output:
10<->8<->4<->5
Explanation:
All Occurences of 2 have been deleted.
```

Your Task:

Complete the function void deleteAllOccurOfX(struct Node** head_ref, int key), which takes the reference of the head pointer and an integer value key. Delete all occurrences of the key from the given DLL.

Expected Time Complexity: O(N).

Expected Auxiliary Space: O(1).

Constraints:

$1<=$Number of Nodes$<=10^4$

$-10^3<=$Node Value $<=10^3$

```
class Solution {
  public:

  void deleteNode(struct Node** head_ref, struct Node* curr){
        if (*head_ref == NULL || curr == NULL)
            return;

        if (*head_ref == curr)
            *head_ref = curr->next;

        if (curr->next != NULL)
            curr->next->prev = curr->prev;

        if (curr->prev != NULL)
            curr->prev->next = curr->next;

        free(curr);
    }



    void deleteAllOccurOfX(struct Node** head_ref, int x) {
        // Write your code here
       struct Node* current = *head_ref;
        struct Node* next;
        while(current!=NULL){
            if(current->data == x){
                next  = current->next;
                deleteNode(head_ref, current);
                current = next;
            }else{
                current = current->next;
            }
        }


    }
};
```

-: Done for the today :-