XOR - Linked List

Amogh Karhadkar
IBM18CS014

```
void insert_front(Node** head, int val){
    if (*head == NULL){
        *head = new Node();
        (*head) -> data = val;
        (*head) -> next = NULL;
        return;
    }
    Node* newnode = new Node();
    newnode -> data = val;
    newnode -> next = (*head);      // as NULL xor *head = *head
    (*head) -> next = (Node*)((uintptr_t)(a) ^ (uintptr_t)(b));
    (*head) = newnode;
}

void insert_end(Node** head, int val){
    if(*head == NULL){
        *head = new Node();
        (*head) -> data = val;
        (*head) -> next = NULL;
        return;
    }
    Node* newnode = new Node();
    newnode -> data = val;
    newnode -> next = NULL;

    Node* curr = *head;
```


Amy

Amogh Kathadkar
IBM18CS014

```
Node * prev = NULL;
~~while(curr != NULL){~~
Node* next = (Node*)((uintptr_t)(prev) ^ (uintptr_t)(curr));
                                                    next
while(next != NULL){

    prev = curr
    curr = next
    next = (Node*)((uintptr_t)(prev) ^ (uintptr_t)(curr->next));

}

newnode->next = curr;
curr->next = (Node*)((uintptr_t)(prev) ^ (uintptr_t)(newnode));

}
```


dcy