ADS LAB-2 : 23rd Sept. 2020

Program: Memory efficient DLL.

K. ARUN TEJA
1BM18CS041
5A

```
Struct node {
    int data,
    struct node *npx
    };
typedef struct node *Node

Node XOR ( Node a, Node b) {
    return (Node) ((uintptr_t) a ^ (uintptr_t) b);
    }

void insert ( Node head, int data)

Node getNode() {
    Node p;
    p = (Node) malloc (sizeof (struct node));
    if ( p != null)
        return p;
    else {
        printf ( "Memory is not allocated");
        exit(0);
        }.
    }.

Void insert_ff( Node head, int data) {
    Node newn = getNode();
    newn → data = data;
    newn → npx = head;
    if ( head != null) {
        head → npx = XOR (newn, head → npx);
        }
    head = newn;
}.
```

```
void insert_r ( Node head, int data) {
    Node newn = getNode();
    newn → data = data;
    newn → npx = XOR( newn,
    Node curr = head;
    Node prev = null;
    Node next;
    while (curr != null) {
        next = XOR(prev, curr → npx);
        prev = curr;
        curr = next;
    }
    curr → npx = XOR(
    newn → npx = XOR(newn, curr)
    curr → npx = XOR(prev, curr → npx);
}
```