```
#include <stdio.h>

#include <stdlib.h>

typedef struct node {

    int data;

    struct node *prev, *next;

} node;

node* create() {

    node *head = NULL, *p, *q = NULL;

    int i, n;

    printf("Enter number of nodes:\n");

    scanf("%d", &n);

    printf("Enter data:\n");

  for (i = 0; i < n; i++) {

        p = (node*)malloc(sizeof(node));

        scanf("%d", &p->data);

        p->next = NULL;

        p->prev = q;

        if (head == NULL) {

            head = p;

        } else {

            q->next = p;

        }

        q = p;

    }

    return head;

}

void display(node* head) {

    node* p;

    if (head == NULL) {

        printf("List is empty\n");

        return;
    }
    printf("Elements in the list:\n");
    for (p = head; p != NULL; p = p->next) {
        printf("%d\t", p->data);
    }
    printf("\n");
}
node* insert_front(node* head, int x) {
    node* p = (node*)malloc(sizeof(node));
    p->data = x;
    p->prev = NULL;
    p->next = head;
    if (head != NULL)
        head->prev = p;
    return p;
}
node* insert_rear(node* head, int x) {
    node *p = (node*)malloc(sizeof(node)), *q;
    p->data = x;
    p->next = NULL;

    if (head == NULL) {
        p->prev = NULL;
        return p;
    }

    q = head;
    while (q->next != NULL)
        q = q->next;

    q->next = p;
    p->prev = q;
    return head;
}

node* delete_front(node* head) {
    if (head == NULL) {
        printf("List is empty\n");
        return head;
    }
    printf("Item deleted: %d\n", head->data);
    node* temp = head;
    head = head->next;
    if (head != NULL)
        head->prev = NULL;
    free(temp);
    return head;
}

node* delete_rear(node* head) {
    if (head == NULL) {
        printf("List is empty\n");
        return head;
    }
    node* p = head;
    if (p->next == NULL) {
        printf("Item deleted: %d\n", p->data);
        free(p);
        return NULL;
    }
    while (p->next != NULL)
        p = p->next;
    printf("Item deleted: %d\n", p->data);
    p->prev->next = NULL;
    free(p);
    return head;
}

int main() {
    node* head = NULL;
    head = create();
    display(head);
    head = insert_front(head, 10);
    display(head);
    head = insert_rear(head, 20);
    display(head);
    head = delete_front(head);
    display(head);
    head = delete_rear(head);
    display(head);
    return 0;
}
```