DATE: PAGE:

23/11/20

LAB-6

Adilya Satish
1BM19CS191

Linked List Implementation.

Q.

Ans

```
#include <stdio.h>
#include <stdlib.h>
struct node
{
  int data;
  struct node *~~head~~ next;
} struct node *head;

void insert_beg();
void insert_last();
void insert_random();
void delete_beg();
void delete_last();
void delete_random();
void display();
int item
void insert_beg() {
      if (ptr == NULL) {
          printf("overflow");
      }
} else {
      ptr->data = item;
      ptr->next = head;
      head = ptr;
      printf("Inserted");
}
}
```

DATE: PAGE:

```
void insert_last(){
    if (ptr == NULL){
        printf(Overflow)
    }
    else{
        if (head == NULL){
        ptr->next = NULL;
        head = ptr;
        printf("Inserted")
    }
    else
    { temp = head;
      while (temp->next != NULL){
          temp = temp->next;
      }
    }
  }
}
void insert_random() {
    int locat;
    if (ptr == NULL)
        printf(Overflow);

    else{
        printf("Enter value");
        scanf("%d", &item);
        ptr->data = item;
        printf ("Enter location:");
        scanf ("%d", &locat);
        temp = head;
```

```
for (i=0; i<locat; i++)
{
    temp = temp->next;
    if (temp=NULL){
        cetiven false;
    }
} else
ptr->next= temp->next;
temp->next=ptr;
}
}

void delete_beg(){
    if(head==NULL)
        cetiven false
    else{
        ptr=head;
        head= ptr->next;
        free(ptr);
    }
}

void delete_last(){
    if(head->next==NULL){
        head=NULL;
        free(head);
    }
    else if(ptr->next!=NULL){
        ptr1=ptr
        ptr=ptr->next
```

DATE: PAGE:

```
else {
    ptr = head;
    while (ptr->next != NULL)
        ptr1 = ptr;
        ptr = ptr->next;
    ptr->next = NULL;
    free(ptr)
  }
}.
void delete_random(){
    struct node *ptr, *ptr1
    int locat;
    printf("u Enter location");
    scanf("u %d", &locat);
    ptr = head;
    for(i=0; i<locat; i++){
        ptr1 = ptr
        ptr->next = ptr;
        free(ptr)
        printf("u Deleted Node");
        if (ptr==NULL){
            printf("Deletion not possible");
        }
        ptr1->next = ptr->next;
        free(ptr);
    }
```

DATE: PAGE:

```
void display () {
    if (ptr == NULL) {
        printf (Empty);
    }
    else {
        while (ptr != NULL) {
            printf ("%d", ptr -> data);
            ptr = ptr -> next;
        }
    }
}
```