Day-6

Implementation of linked List:

Code:

```
#include <stdio.h>
#include<stdlib.h>
struct Node{
    int data;
    struct Node *next;
};
void display();
struct Node* head;
void insertStart(int data){
    struct Node* nn=(struct Node*)malloc(sizeof(struct Node));
    if(head==NULL){
        nn->data=data;
        nn->next=NULL;
        head=nn;
    }
    else
    {
        nn->data=data;
        nn->next=head;
        head=nn;
    }
}
void insertindex(int index,int data){
    struct Node* nn=(struct Node*)malloc(sizeof(struct Node));
```

```
    struct Node* h=head;
    if(index==0)
        insertStart(data);
    else
    {
        nn->data=data;
        for(int i=0;i<index-1;i++){
            h=h->next;
        }
        nn->next=h->next;
        h->next=nn;

    }
}
void display(){
    struct Node* h=head;
    while(h->next!=NULL){
        printf("%d ",h->data);
        h=h->next;
    }
    printf("%d\n",h->data);
    return ;
}
int size(){

    int c=0;
   struct Node *h=head;
```

```
    while(h->next!=NULL){
        c++;
        h=h->next;
    }
    c++;
    return c;
}
void insert(int data){
    struct Node *h=head;
    struct Node *nn=(struct Node*)malloc(sizeof(struct Node));
    if(h==NULL){
        insertStart(data);
        return ;
    }
    while(h->next!=NULL){
        h=h->next;
    }
    nn->data=data;
    h->next=nn;
    nn->next=NULL;
    return ;
}
int main()
{
    insertStart(10);
    insertindex(1,23);
    insertindex(2,203);
```

```
    insertindex(3,230);
    insert(34);
    printf("\nSize of Linked List is %d\n",size());
    display();
    return 0;
}
```

Output:

1. **Delete a Node in the List.**

Code:

```
#include<stdio.h>
#include<stdlib.h>
struct Node{
    int data;
    struct Node *next;
};
struct Node *head;
void delete(int);
void insert(int data){
    struct Node *h=head;
    struct Node *nn=(struct Node*)malloc(sizeof(struct Node));
    nn->data=data;
    nn->next=NULL;
    if(h==NULL){

        head=nn;
```

```
        return ;
    }
    while(h->next!=NULL){
        h=h->next;
    }
    h->next=nn;
    return ;
}
void display(){
    struct Node* h=head;
    while(h->next!=NULL){
        printf("%d ",h->data);
        h=h->next;
    }
    printf("%d\n",h->data);
    return ;
}
void delete(int delete){
    // printf("Entering...");
    struct Node *h=head;
    int index=-1;
    int i=0;
    while(h!=NULL){
        i++;
        if(h->data==delete){
            index=i;
            break;
```

```
        }

        h=h->next;
    }
    if(index==-1){
        printf("Element not found\n");
        return ;
    }
    struct Node *h1=head;
    for(int i=1;i<index-1;i++){
        h1=h1->next;
    }
    struct Node *del=h1->next;
    h1->next=del->next;



}
int main(){
    struct Node s;
    printf("Ente the number of nodes you want to create\n");
    int n;
    scanf("%d",&n);
    for(int i=1; i<=n; i++){
        int data;
        scanf("%d",&data);
        insert(data);
    }
```

```
    printf("displaying the nodes:\n");
    display();
    printf("Enter the Node data you want to delete:\n");
    int de;
    scanf("%d",&de);
    delete(de);
    printf("After deleting..\n");
    display();
}
```

Output:

2. Write a program to create a structure with fields like name and age and insert that date into a nodes.

Code:

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>

struct Node {
    char name[20];
    int age;
    struct Node *next;
};

struct Node *head;
```

```
void insert(char name[], int age) {
    struct Node *h = head;
    struct Node *nn = (struct Node*)malloc(sizeof(struct Node));
    nn->age = age;
    strcpy(nn->name, name);
    nn->next = NULL;

    if (h == NULL) {
        head = nn;
        return;
    }

    while (h->next != NULL) {
        h = h->next;
    }

    h->next = nn;
}

void display() {
    printf("Checking..\n");
    struct Node* h = head;
    while (h != NULL) {
        printf("\nName: %s and  Age: %d\n", h->name, h->age);
        h = h->next;
    }
```

```
}

int main() {
    printf("Enter the number of nodes you want to create:\n");
    int n;
    scanf("%d", &n);

    for (int i = 1; i <= n; i++) {
        int age;
        char name[20];
        printf("Enter age for node %d\n: ", i);
        scanf("%d", &age);
        printf("Enter name for node %d\n: ", i);
        scanf("%s", name);
        insert(name, age);
    }

    printf("\nDisplaying the nodes:\n");
    display();

    return 0;
}
```

Output:

```
Enter the number of nodes you want to create:
3
Enter age for node 1
: 23
Enter name for node 1
: Gowtham
Enter age for node 2
: 34
Enter name for node 2
: Mahesh
Enter age for node 3
: 45
Enter name for node 3
: James

Displaying the nodes:
Checking..

Name: Gowtham and  Age: 23

Name: Mahesh and  Age: 34

Name: James and  Age: 45
```