**LINKED LIST date=29th nov 2024**

**1)CREATION OF LL**

class Node {

public:

int data;

Node\* next;

Node(int data1, Node\* next1) {

data = data1;

next = next1;

}

Node(int data1) {

data = data1;

next = nullptr;

}

};

**2)CREATION OF DOUBLY LINKED LIST**

Class Node{

Public:

Int data;

Node \*back;

Node\* next;

Node(int data1,Node\* back1,Node\* next1){

Data=data1;

Back=back1;

Next=next1;

}

Node(int data1){

Data=data1;

Back=nullptr;

Next=nullptr;

};

**3)INSERTION IN LL**

class Node {

public:

int data;

Node\* next;

Node(int data1, Node\* next1) {

data = data1;

next = next1;

}

Node(int data1) {

data = data1;

next = nullptr;

}

};

Node\* insertnodeatbeginning(Node\*head,int val){

Node \*temp=new Node(int val,head);

Return temp;

}

**4)deletion in ll**

class Node {

public:

int data;

Node\* next;

Node(int data1, Node\* next1) {

data = data1;

next = next1;

}

Node(int data1) {

data = data1;

next = nullptr;

}

};

**Node\* deleteatend(Node\* head){**

**Node \*temp=head;**

**while(temp->next->next!=NULL){**

**temp=temp->next;**

**}**

**delete temp->next;**

**temp->next=nullptr;**

**return head;**

**5)delete at middle**

class Node {

public:

int data;

Node\* next;

Node(int data1, Node\* next1) {

data = data1;

next = next1;

}

Node(int data1) {

data = data1;

next = nullptr;

}

};

**Node \*deleteatmiddle(Node \*head){**

**If(head ==NULL || head->next==NULL) return NULL;**

**Node\* slow=head;**

**Node\* fast=head;**

**Fast=fast🡪next->next;**

**While(fast!=NULL && fast->next!=NULL){**

**Slow=slow->next;**

**Fast=fast->next->next;**

**}**

**Slow->next=slow->next->next;**

**Return head;**

**6)insert in dll in beginning**

**Node \*insertinbeginning(Node \*head,data){**

**Node \*newnode=new Node(data);**

**If(head==NULL){**

**Head=newnode;**

**Return;**

**}**

**Temp->next=head;**

**Head->prev=temp;**

**Head=newnode;**

**Head->prev=NULL;**

**}**

**7)DELETE NODE IN DLL**

Class Node{

Public:

Int data;

Node \*back;

Node\* next;

Node(int data1,Node\* back1,Node\* next1){

Data=data1;

Back=back1;

Next=next1;

}

Node(int data1){

Data=data1;

Back=nullptr;

Next=nullptr;

};

**Node \*deleteatend(Node \*head){**

**If(head==null) return;**

**If(head->next==null){**

**Delete head;**

**Return;**

**}**

**Node \*temp=head;**

**While(temp->next->next!=NULL){**

**Temp=temp->next;**

**}**

**Delete temp->next;**

**Temp->next=nullptr;**

**Return head;**

**}**

**8)delete in middle in dll**

class Node {

public:

int data;

Node\* next;

Node(int data1, Node\* next1) {

data = data1;

next = next1;

}

Node(int data1) {

data = data1;

next = nullptr;

}

};

**Node \*deleteatmiddle(Node \*head){**

**If(head ==NULL || head->next==NULL) return NULL;**

**Node\* slow=head;**

**Node\* fast=head;**

**While(fast!=NULL && fast->next!=NULL){**

**Slow=slow->next;**

**Fast=fast->next->next;**

**}**

**Slow->prev->next=slow->next;**

**Slow->next->prev=slow->prev;**

**Return head;**