

```

15. void first_insert(int data){
    if(isempty()){
        head=new nod(data);
        tail=head;
    }
    else{
        nod*temp=new nod(data);
        temp->next=head;
        temp->prev=NULL;
        temp->next->prev=temp;
        head=temp;
    }
}

```

```

16. void push(int data){
    if(isempty()){
        nod*temp=new nod(data);
        end=temp;
        front=temp;
    }
    else{
        nod*temp=new nod(data);
        end->next=temp;
        end=temp;
    }
}

```

```

17. void first_remove(){
    header=header->next;
}

```

```

18. void pop(){
    if(!isempty()) top=top->next;
}

```

```

19. void first_remove(){
    nod*temp=head;
    temp->next->prev=NULL;
    head=temp->next;
}

```

```

20.!!
    void pop(){
        if(front==NULL) return;
        nod*temp=front;
        front=front->next;
    }

```

```

21-23!. void insert(int data , int data1){
    nod*temp=head;
    while(temp!=NULL){
        if(temp->data==data){
            nod*it=new nod(data1);
            it->next=temp->next;
            temp->next=it;
            it->prev=temp;
            it->next->prev=it;
            break;
        }
        else temp=temp->next;
    }
}

```

```

22.void remove(int data){
    nod*temp=head;
    if(temp->data==data){
        head=head->next;
        head->prev=NULL;
    }
    else if(tail->data==data){
        tail=tail->prev;
        tail->next=NULL;
    }
    else {
        while(temp!=NULL)
            if(temp->next->data==data){
                temp->next=temp->next->next;
                temp->next->prev=temp;
                break;
            }
        else temp=temp->next;
    }
}

```

```

24. void insert(int data, int data1){
    nod*temp=head;
    while(temp!=NULL){
        if(temp->data==data){
            nod*it=new nod(data1);
            it->next=temp->next;
            temp->next=it;
            break;
        }
        else temp=temp->next;
    }
}

```