

```

#include<iostream>
using namespace std;

/*
@swasthiiii
*/

typedef struct node
{
    int data;
    struct node *ptr;
}NODE;

class stack_list
{
public :
    NODE *push(NODE *);
    NODE *pop(NODE *);
    void display(NODE *);
};

int main()
{
    stack_list sl;
    NODE *head=NULL;;
    int ch;
    int count=0;
    while(1)
    {
        cout<<"\n---menu---\n";
        cout<<"1.push\n";
        cout<<"2.pop\n";
        cout<<"3.display\n";
        cout<<"4.exit\n";
        cout<<"enter your choice:";
        cin>>ch;
        cout<<"\n";
        switch(ch)
        {
            case 1:if(count<5)
                {
                    head=sl.push(head);
                    count++;
                }
            else
            {
                cout<<"stack is full";
            }
        }
    }
}

```

```

        }
        break;
    case 2: head=s1.pop(head);
    if(count>0)
    {
        count--;
    }
    break;
    case 3: s1.display(head);
    break;
    case 4: exit(0);
}

}
return 0;
}

NODE *stack_list::push(NODE *head)
{
    int num;
    NODE *newnode;
    newnode=(NODE *)malloc(sizeof(NODE));
    cout<<"enter the element for the stack:\n";
    cin>>num;

    newnode->data=num;
    newnode->ptr=NULL;

    if(head==NULL)
    {
        head=newnode;
    }
    else
    {
        newnode->ptr=head;
        head=newnode;
    }
    return head;
}

NODE *stack_list::pop(NODE *head)
{
    NODE *temp;
    if(head==NULL)
    {
        cout<<"stack is empty";
    }
    else if(head->ptr==NULL)

```

```

    {
        cout<<"deleted:"<<head->data<<"\n";
        head=NULL;
        free(head);
    }
    else
    {
        temp=head;
        head=head->ptr;
        temp->ptr=NULL;
        cout<<"deleted:"<<temp->data<<"\n";
        free(temp);
    }
    return head;
}

void stack_list::display(NODE *head)
{
    NODE *temp;
    temp=head;
    if(head==NULL)
    {
        cout<<"stack is empty\n";
    }
    else
    {
        cout<<"stack elements\n";
        while(temp!=NULL)
        {
            cout<<temp->data<<"\n";
            temp=temp->ptr;
        }
    }
}
}

```

As we know in linked list we can create stack of unlimited size, but here I created count variable in main function whos size is 5 ,so only 5 nodes will be created ,and give the stack is full msg if you try to insert another node

If you want the N size stack then just do the following changes in main function(switch case);

```
int main()
{
    stack_list sl;
    NODE *head=NULL;;
    int ch;
    while(1)
    {
        cout<<"\n---menu---\n";
        cout<<"1.push\n";
        cout<<"2.pop\n";
        cout<<"3.display\n";
        cout<<"4.exit\n";
        cout<<"enter your choice:";
        cin>>ch;
        cout<<"\n";
        switch(ch)
        {
            case 1:head=sl.push(head);
                break;
            case 2:head=sl.pop(head);
                break;
            case 3:sl.display(head);
                break;
            case 4:exit(0);
        }
    }
    return 0;
}
```