```
1.  //Source code for stack operations, using linked list:
2.  //Source code for stack operations, using linked list:
3.  #include <stdio.h>
4.  //#include <conio.h>
5.  #include <stdlib.h>
6.  struct stack
7.  {
8.  int data;
9.  struct stack *next;
10. };
11. void push();
12. void pop();
13. void display();
14. typedef struct stack node;
15. node *start=NULL;
16. node *top = NULL;
17. node* getnode()
18. {
19. node *temp;
20. temp=(node *) malloc( sizeof(node)) ;
21. printf("\n Enter data ");
22. scanf("%d", &temp -> data);
23. temp -> next = NULL;
24. return temp;
25. }
26. void push(node *newnode)
27. {
28. node *temp;
29. if( newnode == NULL )
30. {
31. printf("\n Stack Overflow..");
32. return;
33. }
34. if(start == NULL)
35. {
36. start = newnode;
37. top = newnode;
38. }
39. else
40. {
41. temp = start;
42. while( temp -> next != NULL)
43. temp = temp -> next;
44. temp -> next = newnode;
45. top = newnode;
46. }
47. printf("\n\n\t Data pushed into stack");
48. }
49. void pop()
50. {
51. node *temp;
52. if(top == NULL)
53. {
54. printf("\n\n\t Stackunderflow");
55. return;
56. }
57. temp = start;
58. if( start -> next == NULL)
59. {
60. printf("\n\n\t Popped element is %d ", top -> data);
61. start = NULL;
62. free(top);
63. top = NULL;
```

```
}
else
{
while(temp -> next != top)
{
temp = temp -> next;
}
temp -> next = NULL;
printf("\n\n\t Popped element is %d ", top -> data);
free(top);
top = temp;
}
}
void display()
{
node *temp;
if(top == NULL)
{
printf("\n\n\t\t Stack is empty ");
}
else
{
temp = start;
printf("\n\n\n\t\t Elements in the stack: \n");
printf("%5d ", temp -> data);
while(temp != top)
{
temp = temp -> next;
printf("%5d ", temp -> data);
}
}
}
char menu()
{
char ch;
//clrscr();
        printf("\n \tStack operations using pointers.. ");
        printf("\n ----------**********----------\n");
        printf("\n 1. Push ");
        printf("\n 2. Pop ");
        printf("\n 3. Display");
        printf("\n 4. Quit ");
        printf("\n Enter your choice:");
          ch = getchar();
        return ch;
        }
        void main()
        {
        char ch;
        node *newnode;
        do
        {
        ch = menu();
        switch(ch)
        {
        case '1' :
        newnode = getnode();
        push(newnode);
        break;
        case '2' :
        pop();
        break;
        case '3' :
```

```
127.        display();
128.        break;
129.        case '4':
130.        return;
131.        }
132.        //getch();
133.        } while( ch != '4' );
134.        }
135.
```