

Aim:

Write a C program that implements a stack using linked list. The program should have **push()**, **pop()**, **display()**, **isEmpty()** and **peek()** functions that uses switch cases and works on the choice of the user.

Source Code:**StackListPopPeek.c**

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

struct stack {
    int data;
    struct stack *next;
};

typedef struct stack *stk;
stk top = NULL;
stk temp, preptr;

void pop() {
    // write your code here to pop an element from the stack
    int el; temp = top;
    if(temp == NULL){
        printf("Stack is underflow.\n");return;
    }
    printf("Popped value = %d\n", top->data);
    top = top-> next;free(temp);
}

void peek() {
    // write your code here to find the peek of the stack
    if(top==NULL) {printf("Stack is underflow.\n");return;
    }
    printf("Peek value = %d\n",top->data);
}

void isEmpty() {
    // write your code here to find whether the stack is empty or not
    if (top==NULL)printf("Stack is empty.\n");
    else printf("Stack is not empty.\n");
}

void push(int x) {
    // write your code here to push an element into the stack
    stk new = (stk)malloc(sizeof(stk));
    new -> data = x;new -> next = NULL;
    if (top==NULL) top=new;
    else{
        new -> next = top;
        top = new;
    }
}
```

```

    printf("Successfully pushed.\n");
}

void display() {
    // write your code here to display the stack elements
    temp=top;
    if(temp == NULL){
        printf("Stack is empty.\n");return;
    }
    printf("Elements of the stack are : ");
    while(temp!=NULL){
        printf("%d ",temp->data);
        temp = temp -> next;
    }
    printf("\n");
}

int main() {
    int op, x;
    while(1) {
        printf("1.Push 2.Pop 3.Display 4.Is Empty 5.Peek 6.Exit\n");
        printf("Enter your option : ");
        scanf("%d",&op);
        switch(op) {
            case 1:
                printf("Enter element : ");
                scanf("%d",&x);
                push(x);
                break;
            case 2:
                pop();
                break;
            case 3:
                display();
                break;
            case 4:
                isEmpty();
                break;
            case 5:
                peek();
                break;
            case 6:
                exit(0);
        }
    }
}

```

Execution Results - All test cases have succeeded!

Test Case - 1
User Output
1.Push 2.Pop 3.Display 4.Is Empty 5.Peek 6.Exit 1
Enter your option : 1
Enter element : 10

Successfully pushed. 1
1.Push 2.Pop 3.Display 4.Is Empty 5.Peek 6.Exit 1
Enter your option : 1
Enter element : 20
Successfully pushed. 1
1.Push 2.Pop 3.Display 4.Is Empty 5.Peek 6.Exit 1
Enter your option : 1
Enter element : 30
Successfully pushed. 2
1.Push 2.Pop 3.Display 4.Is Empty 5.Peek 6.Exit 2
Enter your option : 2
Popped value = 30 3
1.Push 2.Pop 3.Display 4.Is Empty 5.Peek 6.Exit 3
Enter your option : 3
Elements of the stack are : 20 10 4
1.Push 2.Pop 3.Display 4.Is Empty 5.Peek 6.Exit 4
Enter your option : 4
Stack is not empty. 5
1.Push 2.Pop 3.Display 4.Is Empty 5.Peek 6.Exit 5
Enter your option : 5
Peek value = 20 2
1.Push 2.Pop 3.Display 4.Is Empty 5.Peek 6.Exit 2
Enter your option : 2
Popped value = 20 2
1.Push 2.Pop 3.Display 4.Is Empty 5.Peek 6.Exit 2
Enter your option : 2
Popped value = 10 3
1.Push 2.Pop 3.Display 4.Is Empty 5.Peek 6.Exit 3
Enter your option : 3
Stack is empty. 4
1.Push 2.Pop 3.Display 4.Is Empty 5.Peek 6.Exit 4
Enter your option : 4
Stack is empty. 6
1.Push 2.Pop 3.Display 4.Is Empty 5.Peek 6.Exit 6
Enter your option : 6

Test Case - 2
User Output
1.Push 2.Pop 3.Display 4.Is Empty 5.Peek 6.Exit 4
Enter your option : 4
Stack is empty. 5
1.Push 2.Pop 3.Display 4.Is Empty 5.Peek 6.Exit 5
Enter your option : 5
Stack is underflow. 3
1.Push 2.Pop 3.Display 4.Is Empty 5.Peek 6.Exit 3
Enter your option : 3
Stack is empty. 1
1.Push 2.Pop 3.Display 4.Is Empty 5.Peek 6.Exit 1
Enter your option : 1
Enter element : 45

Successfully pushed. 1
1.Push 2.Pop 3.Display 4.Is Empty 5.Peek 6.Exit 1
Enter your option : 1
Enter element : 67
Successfully pushed. 3
1.Push 2.Pop 3.Display 4.Is Empty 5.Peek 6.Exit 3
Enter your option : 3
Elements of the stack are : 67 45 5
1.Push 2.Pop 3.Display 4.Is Empty 5.Peek 6.Exit 5
Enter your option : 5
Peek value = 67 3
1.Push 2.Pop 3.Display 4.Is Empty 5.Peek 6.Exit 3
Enter your option : 3
Elements of the stack are : 67 45 2
1.Push 2.Pop 3.Display 4.Is Empty 5.Peek 6.Exit 2
Enter your option : 2
Popped value = 67 2
1.Push 2.Pop 3.Display 4.Is Empty 5.Peek 6.Exit 2
Enter your option : 2
Popped value = 45 2
1.Push 2.Pop 3.Display 4.Is Empty 5.Peek 6.Exit 2
Enter your option : 2
Stack is underflow. 6
1.Push 2.Pop 3.Display 4.Is Empty 5.Peek 6.Exit 6
Enter your option : 6

