



KIET Group of Institutions, Ghaziabad

Department of Computer Applications

(An ISO – 9001: 2015 Certified & 'A' Grade accredited Institution by NAAC)

Data Structure and Algorithm Analysis Lab

KCA 253: Session 2020-21

Experiment – No-9

Scheduled Date	Compiled Date	Submission Date
25-May-2021	25-May-2021	25-May-2021

Program : Write a Program to Implement Stack Using Linked list.

```
#include<stdio.h>
#include<stdlib.h>
typedef struct nodeType
{
    int info;
    struct nodeType *next;
}stack;
typedef enum { false , true} boolean;
void createstack(stack **ps);
void push(stack **ps,int value); //to add a new element
int pop(stack **ps); //to remove the last entering element
int peep(stack *ps); //to show the top element of stack
void disposestack(stack **ps);
boolean isempty(stack *ps); //to check stack is empty or not
void main()
{
    stack *top;
    int choice,element;
    createstack(&top);
    do
    {
        printf("-----Menu-----");
        printf("\n");
        printf("\n 1.Push");
        printf("\n 2.Pop");
        printf("\n 3.Peep");
        printf("\n 4.Exit");
        printf("\nEnter your choice");
        scanf("%d",&choice);
        switch(choice)
        {
            case 1:
                printf("enter element to push");
                scanf("%d",&element);
                push(&top,element);
                break;
            case 2:
                if(isempty(top))
                {
                    printf("stack is empty\n");
                    getch();
                }
                else
                {
                    printf("Value popped is %d\n",pop(&top));
                    printf("press any key to continue.....");
```



KIET Group of Institutions, Ghaziabad

Department of Computer Applications

(An ISO – 9001: 2015 Certified & 'A' Grade accredited Institution by NAAC)

Data Structure and Algorithm Analysis Lab

KCA 253: Session 2020-21

```
                getch();
            }
            break;
        case 3:
            if(isempty(top))
            {
                printf("stack is empty\n");
                getch();
            }
            else
            {
                printf("Value at top is %d\n",peek(top));
                printf("press any key to continue.....");
                getch();
            }
        }
    }
    while(choice !=4);
    disposestack(&top);
}
void createstack(stack **top){
    *top = NULL;
}
boolean isempty(stack *top)
{
    if(top==NULL)
        return true;
    else
        return false;
}
void push(stack **top,int value)
{
    stack *ptr;
    ptr = (stack *)malloc(sizeof(stack));
    if(ptr==NULL)
    {
        printf("\nunable to allocate to memory for new node....");
        printf("press any key to continue.....");
        getch();
        return;
    }
    ptr->info = value;
    ptr->next = *top;
    *top = ptr;
}
int pop(stack **top)
{
    int temp;
    stack *ptr;
    temp = (*top)->info;
    ptr=*top;
    *top=(*top)->next;
    free(ptr);
    return temp;
}
int peek(stack *top)
{

```



KIET Group of Institutions, Ghaziabad

Department of Computer Applications

(An ISO – 9001: 2015 Certified & 'A' Grade accredited Institution by NAAC)

Data Structure and Algorithm Analysis Lab

KCA 253: Session 2020-21

```
        return (top->info);
    }
    void disposestack(stack **top){
        stack *ptr;
        while(*top !=NULL){
            ptr = *top;
            *top =(*top)->next;
            free(ptr);
        }
    }
```

Output

```
-----Menu-----

1.Push
2.Pop
3.Peep
4.Exit
Enter your choice1
enter element to push12
-----Menu-----

1.Push
2.Pop
3.Peep
4.Exit
Enter your choice3
Value at top is 12
press any key to continue.....
-----Menu-----

1.Push
2.Pop
3.Peep
4.Exit
Enter your choice2
Value popped is 12
press any key to continue.....
```