STACK USING ARRAY IN C

```
#define  maxsize 13
```

```
int top=-1;
void push(int stack[],int key)
    {
        if(top==maxsize-1)
            {
                printf("stack is full\n");
            }
        else
        {
            stack[++top]=key;
        }
    }

void printstack(int stack[])
    {
        int i=0;
        for(i=0;i<=top;i++)
            {
            printf("%d\t",stack[i]);
            }
        printf("\n%d\n",top);
 }

 void Pop()
    {
        if(top==-1)
        {
            printf("\nstack is underflow:\n");
        }
        else
            {
                top=top-1;
            }
    }
int main()
    {
        int stack[maxsize];
        push(stack,10);
        push(stack,20);
        push(stack,30);
        push(stack,40);
        printstack(stack);
        Pop();
        printstack(stack);

        return 0;
    }
```

STACK USING LINKLIST IN C

```
#include<stdio.h>
#include<stdlib.h>
struct stack
    {
        int data;
        struct stack* next;
    };
void pop(struct stack** head)
    {
        struct stack* current,*parrent;

        if(*head==NULL)
        {
            printf("stack is underflow:\n");
        }
        else
        {   current=*head;
                if(current->next==NULL)
                {
                    *head=NULL;
                    printf("\n--NOW STACK IS
UNDERFLOW--\n");
                }
                else{
                while(current->next)
                    {   parrent=current;
                        current=current->next;
                    }
                parrent->next=NULL;
                free(current);
                }
        }
    }
void push(struct stack** head,int key)
    {
        struct stack* current;
        struct stack* newnode,*parrent;
        newnode=(struct stack*)malloc(sizeof(struct stack));
        newnode->data=key;
        newnode->next=NULL;

        if(*head==NULL)
            {
                *head=newnode;
            }
        else
```

```
      {
        current=*head;
        while(current)
        {   parrent=current;
            current=current->next;
        }

        parrent->next=newnode;
      }
  }

  void display(struct stack**head)
    {
      struct stack* current;
      current=*head;
      printf("\n-----stack values are-----\n");
       if(current==NULL)
        {
          printf("\n--NOW STACK IS
UNDERFLOW--\n");
        }
      while(current)
      {
        printf("%d\t",current->data);
        current=current->next;
      }
```

```
        }

int main()
   {
   int choice=0,key;
   struct stack* head=NULL;

   printf("\n*********Stack operations using linked
list*********\n");
   printf("\n--------------------------------------------------\n");
   while(choice != 4)
   {
      printf("\n\nChose one from the below options...\n");
      printf("\n1.Push\n2.Pop\n3.Show\n4.Exit");
      printf("\n Enter your choice \n");
      scanf("%d",&choice);
      switch(choice)
      {
         case 1:
         {
            printf("enter the value:\n");
            scanf("%d",&key);
            push(&head,key);
            break;
         }
         case 2:
```

```
        {
            pop(&head);
            break;
        }
        case 3:
        {
            display(&head);
            break;
        }
        case 4:
        {
            printf("Exiting....");
            break;
        }
        default:
        {
            printf("Please Enter valid choice ");
        }
    };
}
    return 0;
}
```