DAYANAND

1BM19CS043

```
#include <stdio.h>
#include < stdlib.h>
  struct node {
     int data;
      struct  node *next; };
    void  qinsert();
    void  qdisplay();
    void  qdel();
   void spush();
   void spop();
   void  sdisplay();
    struct  node *rear=NULL, *front= NULL, *top=NULL;
    int main (int  argc, char  **argv)
    { int  choice;
        Printf"Enter the  choice\n1.stack\n2.queue\n");
         Scanf("%d", &choice);
     if(choice ==1)
    do{ Printf( "\n1. Push \n2.Display \n3.pop\n");
      Printf ("\nEnter  your  choice:");
       Scanf ("%d", &choice);
        switch( choice)
     { case 1: spush(); break;
        case 2: sdisplay; break;
        case 3:  spop;  break;
     default:  if(choice!=4)
              Printf("\n Invalid  Input");
  } } while (choice!=4);
    use if ( choice ==2)
    do{ Printf (" \n Queue  implementation  using
                 linked list \n");
```

```
Printf("\n1. Create \n2. Display \n3. Delete \n4. Exit \n");
 Printf("\n Enter your choice :");
   scanf("%d", &choice);
    switch (choice)
    { case 1: qinsert(); break;
      case 2: qdisplay(); break;
      case 3: qdel(); break;
      default: if (choice!=4)
           Printf("\nInvalid input");
    }
      while (choice!= 4); }
    Void qinsert()
{ struct node *newnode;
  newnode =(struct node*)malloc(sizeof (struct node));
   Printf(" Enter the element :\n");
  scanf ("%d", &newnode->data);
   newnode->next= NULL;
    if (rear== NULL)
  { rear= newnode;
     front= newnode;
  }
  else rear-> next= newnode;
        rear =newnode;
    }
    void qdel()
    { if (front==NULL)
   { Printf(" Queue is empty\n"); return; }
    else { Printf ("Deleted ele is %d", front-> data);
      if (front==rear)
   { Printf(" Queue is empty\n");
       front=NULL; rear=NULL;
    }
```

```
else
 front=front->next; }}
  void qdisplay() {
     struct node *temp;
      if (front==NULL)
      { printf("queue is empty");
        return;
}
 temp=front;
  while(temp!=NULL)
{ printf("%d", temp->data);
    temp=temp->next; }}
  void spush()
{ int item;
   struct node *newnode;
Printf(" Enter the element\n");
  scanf("%d", &item);
 newnode = (struct node *) malloc(sizeof(structnode));
  newnode-> data = item;
   newnode -> next=NULL;
    if(top== NULL)
      top=newnode;
      else
       newnode->next=top;
       top=newnode;
}
 void spop(){
    if (top==NULL)
      printf(" stack- is empty");
 else { printf(" element removed is %d :", top->data);
    top=top->next; }}
```

```
void display()
{
  struct node *temp;
     temp=top;
     if(top==NULL)
      printf("stack is empty");
     while(temp!=NULL)
     { printf("%d",temp->data);
        temp=temp->next;
     }
}
```