**// Create a Queue ADT using Linked List Data structure and perform its //operations**

#include<iostream.h>

#include<stdio.h>

#include<conio.h>

// Structure of Node

struct Node

{

int data;

Node \*link;

};

Node \*front = NULL;

Node \*rear = NULL;

Node \*temp;

//function to enter elements in queue

void enqueue ( int val)

{

Node \*temp = new Node();

temp->data= val;

temp->link = NULL;

//If inserting the first element/node

if(( front == NULL ) && (rear==NULL))

{

front = rear= temp;

}

else

{

rear ->link = temp;

rear = temp;

}

}

//function to delete/remove element from queue

void dequeue ( )

{

temp=front;

if((front==NULL)&&(rear==NULL))

cout<<"Queue is empty\n";

else if(temp->link!=NULL)

{

temp=temp->link;

cout<<"Value deleted from queue is:"<<front->data;

delete(front);

front=temp;

}

else

{

cout<<"Values deleted from queue is:"<<front->data;

delete(front);

front=rear =NULL;

}

}

//function to display queue

void displayQueue()

{

if((front==NULL)&&(rear==NULL))

cout<<"Queue is empty\n";

else

{

cout<<"Queue Elements are:";

Node \*temp = front;

while(temp!=NULL)

{

cout<<temp->data<<"-> ";

temp=temp->link;

}

cout<<"NULL";

}

}

//Main Function

void main()

{

clrscr();

int option, val;

cout<<"\n QUEUE ADT OPERATIONS USING LINKED LIST \n";

cout<<"1) Insert an element into queue"<<endl;

cout<<"2) delete the element from queue"<<endl;

cout<<"3) Display the queue"<<endl;

cout<<"4) Exit"<<endl;

while(option!=4)

{

cout<<"\nEnter Choice:"<<endl;

cin>>option;

switch (option)

{

case 1:

{

cout<<"Enter Value to be insert:\n";

cin>>val;

enqueue(val);

break;

}

case 2:

{

dequeue();

break;

}

case 3:

{

displayQueue();

break;

}

case 4:

{

cout<<"Exiting..."<<endl;

break;

}

default:

{

cout<<"Invalid Choice"<<endl;

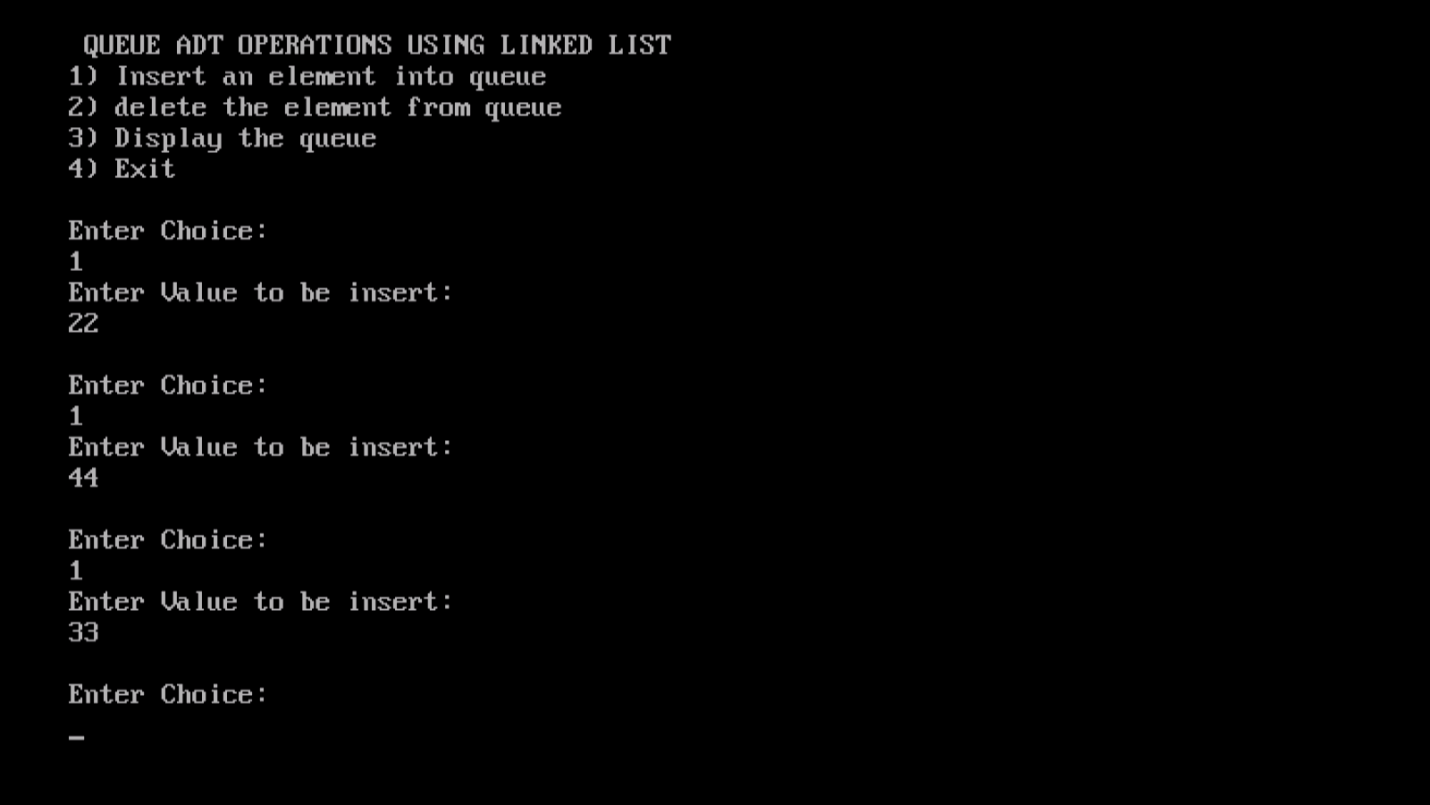
}

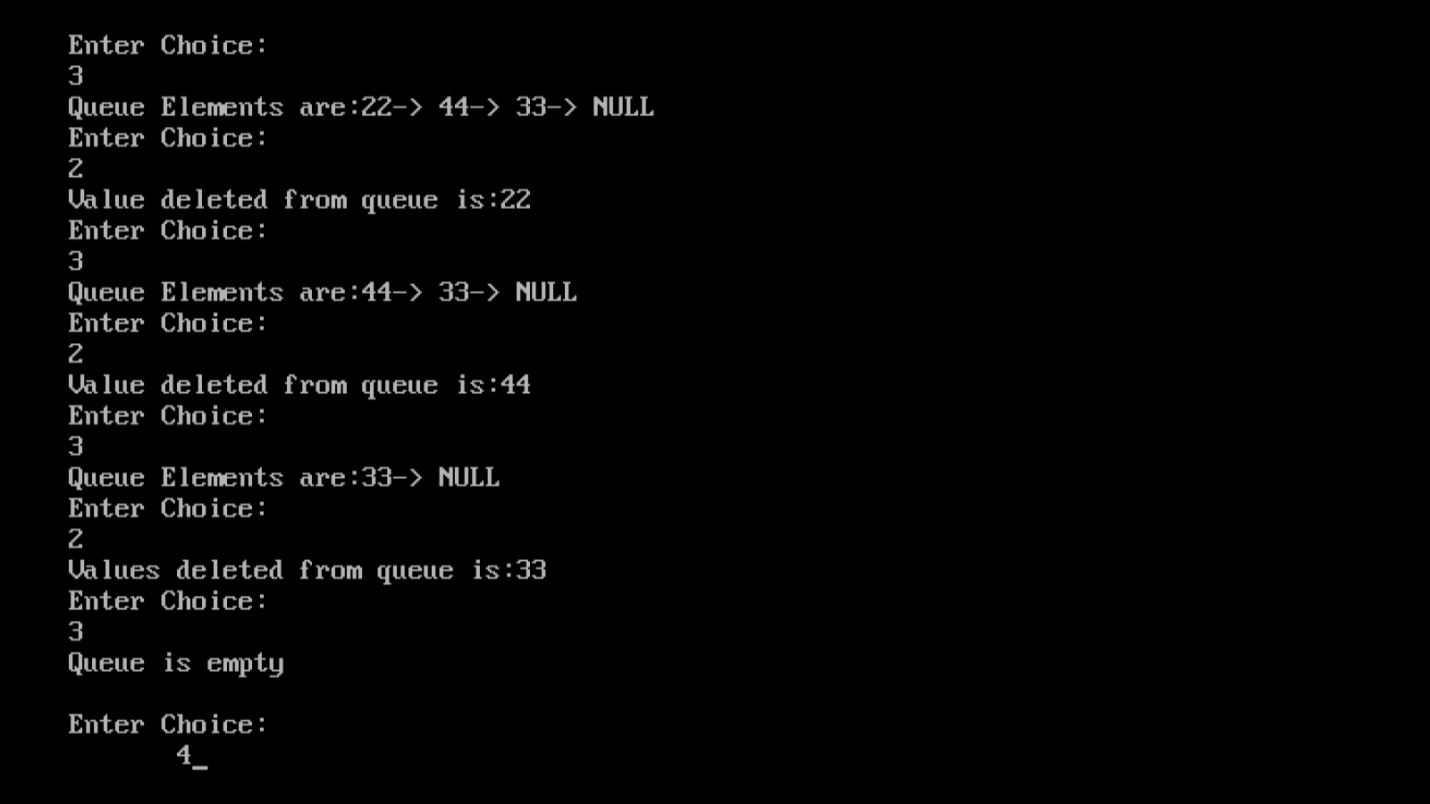
}

}

}

**OUTPUT:**





**//This code was contributed by G.Sundar Manikandan- MCA Section ’B’**