6.3 Program that Implements priority Queue using Linked list.

#include <conio.h> #include <iostream> using namespace std;

class queue

{

private:

int data; queue \*next;

public:

};

void add(int); void display(); int remove();

void priority(void);

queue \*front; queue \*rear;

void queue::add(int num)

{

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

{

front=rear=new queue; front->data=num; front->next=NULL;

}

else

{

rear->next=new queue; rear->next->data=num; rear->next->next=NULL; rear=rear->next;

}

}

void queue::display()

{

queue \*q=front; if((front==NULL)&&(rear==NULL))

{

cout<<"\nStack empty";

}

else

{

while(q!=NULL)

{

cout<<q->data<<" "; q=q->next;

}

}

}

int queue::remove()

{

int num; queue \*temp;

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

{

return -1;

}

else

{

temp=front; num=front->data; front=front->next; delete(temp);

}

return num;

}

void queue :: priority(void)

{

int num;

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

{

}

else

{

cout<<"Queue Is Empty";

if(front->data > rear->data)

{

}

}

int main()

{

}

else

{

}

num = remove(); cout<<num;

cout<<"Enter The Number:"; cin>>num;

add(num);

queue z; char c='y'; int num,op; while(c=='y')

{

cout<<"\n1.Add"; cout<<"\n2.Display"; cout<<"\n3.Remove"; cout<<"\n4.Priority"; cout<<"\n Enter an option : "; cin>>op;

switch(op)

{

case 1:

case 2:

cout<<"\nEnter value: "; cin>>num;

z.add(num);

cout<<"\n\tDo you want to continue: "; break;

cout<<"Queue values:\t";

case 3:

z.display();

cout<<"\n\tDo you want to continue: "; break;

num=z.remove();

cout<< num <<" removed";

cout<< "\n\tDo you want to continue "; break;

case 4:

}

z.priority();

cout<< "\n\tDo you want to continue "; break;

c=getch();

}

}