**ROLL NO:-45**

**NAME : Harshit Atul Chilvirwar**

**PRACTICAL NO:-**

**PRACTICAL NAME :- IMPLEMENTATION OF CIRCULAR LINKED LIST.**

#include "iostream.h"

#include "conio.h"

class NODE

{

public:

int data;

NODE \*next;

};

class LIST

{

NODE \*start,\*end;

public:

LIST();

void ADD\_FIRST(int);

void ADD\_LAST(int);

int DEL\_FIRST();

int DEL\_LAST();

void VIEW\_FWD();

int IS\_EMPTY();

};

LIST::LIST()

{

start = NULL;

end = NULL;

}

void LIST::ADD\_FIRST(int ele)

{

NODE \*NN;

NN= new NODE();

NN->data = ele;

NN->next = NULL;

if(start==NULL)

{

start = NN;

end = NN;

end->next = NN;

}

else

{

NN->next = start;

start = NN;

end->next = NN;

}

}

/

void LIST::ADD\_LAST(int ele)

{

NODE \*NN;

NN= new NODE();

NN->data = ele;

NN->next = NULL;

if(start==NULL)

{

NN->next = NN;

start = NN;

end = NN;

}

else

{

end->next = NN;

NN->next = start;

end = NN;

}

}

int LIST::DEL\_FIRST()

{

if(start == NULL)

{

cout<<endl<<"List is empty";

return NULL;

}

else

{

int ele = start->data;

NODE \* TEMP = start;

if(start->next == start)

{

start = NULL;

end = NULL;

}

else

{

start = start->next;

end->next = start;

}

delete TEMP;

return ele;

}

}

int LIST::DEL\_LAST()

{

if(start == NULL)

{

cout<<endl<<"List is empty";

return NULL;

}

else

{

int ele = end->data;

NODE \* TEMP = end;

if(start->next == start)

{

start = NULL;

end = NULL;

}

else

{

NODE \*prev = start;

while(prev->next != end)

{

prev = prev->next;

}

prev->next=start;

end=prev;

}

delete TEMP;

return ele;

}

}

int LIST::IS\_EMPTY()

{

if(start == NULL)

return 1;

else

return 0;

}

void LIST::VIEW\_FWD()

{

if(start == NULL)

{

cout<<endl<<"List is empty";

return;

}

cout<<endl<<"List elements are : ";

cout<<start->data<<" ";

NODE \*ptr = start->next;

while(ptr != start )

{

cout<<ptr-> data<<" ";

ptr=ptr->next;

}

}

void MENU()

{

int ele, opt, pos;

LIST obj;

do

{

cout<<endl<<"================\n";

cout<<endl<<"1 Add at First";

cout<<endl<<"2 Add at Last";

cout<<endl<<"3 Delete from First";

cout<<endl<<"4 Delete from Last";

cout<<endl<<"5 List All (FWD)" ;

cout<<endl<<"6 Exit";

cout<<endl<<"================\n";

cout<<endl<<"Enter your choice : ";

cin>>opt;

switch(opt)

{

case 1:

cout<<endl<<"Enter element : ";

cin>>ele;

obj.ADD\_FIRST(ele);

obj.VIEW\_FWD();

break;

case 2:

cout<<endl<<"Enter element : ";

cin>>ele;

obj.ADD\_LAST(ele);

obj.VIEW\_FWD();

break;

case 3:

if(!obj.IS\_EMPTY())

{

ele = obj.DEL\_FIRST();

cout<<endl<<"Delted element = "<<ele;

obj.VIEW\_FWD();

}

else

cout<<endl<<"List is empty";

break;

case 4:

if(!obj.IS\_EMPTY())

{

ele = obj.DEL\_LAST();

cout<<endl<<"Delted element = "<<ele;

obj.VIEW\_FWD();

}

else

cout<<endl<<"List is empty";

break;

case 5:

obj.VIEW\_FWD();

break;

case 6:

return;

default:

cout<<endl<<"invalid input";

}

}while(1);

}

void main()

{

clrscr();

MENU();

getch();

}