## Lab6:-

WAP to Implement Singly Linked List with following operations
a) a) Create a linked list. b) Deletion of first element, specified element and last element in the list. c) Display the contents of the linked list.

```
Program:-
#include<stdio.h>
#include<stdlib.h>
struct node
{
int info;
struct node *link;
};
typedef struct node *NODE;
NODE getnode()
{
NODE x;
x=(NODE)malloc(sizeof(struct node));
if(x==NULL)
printf("mem full\n");
exit(0);
}
return x;
```

```
}
void freenode(NODE x)
{
free(x);
}
NODE insert_front(NODE first,int item)
{
NODE temp;
temp=getnode();
temp->info=item;
temp->link=NULL;
if(first==NULL)
return temp;
temp->link=first;
first=temp;
return first;
}
NODE delete_front(NODE first)
{
NODE temp;
if(first==NULL)
```

```
{
printf("List is empty!Can't delete an item'\n");
return first;
}
temp=first;
temp=temp->link;
printf("The Item deleted at front-end is = %d\n",first->info);
free(first);
return temp;
}
NODE insert_rear(NODE first,int item)
{
NODE temp,cur;
temp=getnode();
temp->info=item;
temp->link=NULL;
if(first==NULL)
return temp;
cur=first;
while(cur->link!=NULL)
cur=cur->link;
cur->link=temp;
return first;
}
```

```
{
NODE cur, prev;
if(first==NULL)
{
printf("List is empty cannot delete\n");
return first;
}
if(first->link==NULL)
{
printf("Item deleted is %d\n",first->info);
free(first);
return NULL;
}
prev=NULL;
cur=first;
while(cur->link!=NULL)
{
prev=cur;
cur=cur->link;
}
printf("Item deleted at rear-end is %d",cur->info);
free(cur);
prev->link=NULL;
return first;
}
NODE delete_pos(int pos,NODE first)
{
```

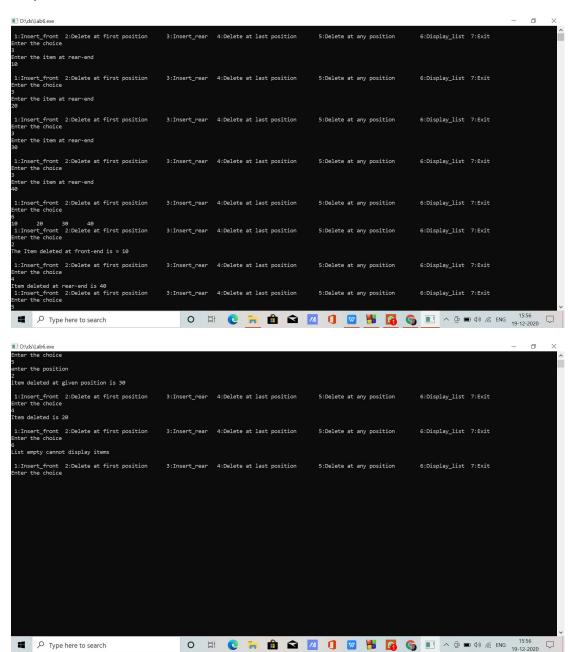
```
NODE cur;
NODE prev;
int count,flag=0;
if(first==NULL || pos<0)
{
printf("invalid position\n");
return NULL;
}
if(pos==1)
{
cur=first;
first=first->link;
freenode(cur);
return first;
}
prev=NULL;
cur=first;
count=1;
while(cur!=NULL)
{
if(count==pos){flag=1;break;}
count++;
prev=cur;
cur=cur->link;
}
if(flag==0)
{
```

```
printf("invalid position\n");
return first;
}
printf("item deleted at given position is %d\n",cur->info);
prev->link=cur->link;
freenode(cur);
return first;
}
void display(NODE first)
{
NODE temp;
if(first==NULL)
printf("List empty cannot display items\n");
for(temp=first;temp!=NULL;temp=temp->link)
 {
 printf("%d\t",temp->info);
}
}
int main()
{
int item, choice, pos;
NODE first=NULL;
for(;;)
```

```
{
printf("\n 1:Insert_front\t 2:Delete at first position\t 3:Insert_rear\t 4:Delete at last
position\t 5:Delete at any position\t 6:Display_list\t 7:Exit\n");
printf("Enter the choice\n");
scanf("%d",&choice);
switch(choice)
{
 case 1:printf("Enter the item at front-end\n");
      scanf("%d",&item);
      first=insert_front(first,item);
      break;
 case 2:first=delete_front(first);
      break;
 case 3:printf("Enter the item at rear-end\n");
      scanf("%d",&item);
      first=insert_rear(first,item);
      break;
 case 4:first=delete_rear(first);
      break;
 case 5:printf("enter the position\n");
      scanf("%d",&pos);
      first=delete_pos(pos,first);
      break;
 case 6:display(first);
      break;
default:exit(0);
      break;
}
```

}

## Output Screenshot:-



Written pictures:-

PAGE NO. DATE: / /
#moludicatdio.n>
#include 2 etallib h> 13 UM - dalil a gross
struct node (JULIS SURIE)
\$ CAMPAGE OF THE P
int into
struct node*link;
1; Charit country
typodet etrut node* NODE;
Nobe gernodil)
None x
I = (No De) mallor (size of (struct mode));
if (x = NITT)
if (x=null)
pf ("mem Jull mus)
pf ("mem full m")
Julium & Comment of the state o
3 11 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
void freehod (Nobe of)
3 man (m)
NODE invort-front (Nobe first, int item)
*
NODE temp; temp = getnodil):
temp = getnodil):

bruak; int item, choice, pos; case 5's Nobe from = NOUS tow (?;) pt ("ensur the position to delete in"):
scant (" x d", & pos); first = delete-per (por, first); break; pt " ins. insert from in 2 invitous in 3 deletion case 6', display (first); of first element mot deletion of east element weat; In detion @ specified position me duplay defaul: exit (0); bruat? of ("enter the enoral");

st ("td", schoid);

ewitch (choid) tempt into = item; NODE delete-front (NODE thrust) temp -> link = NULL's (1100 == turit) fi No DE terres (won== their) to return tempi temp > link = first; firest = temp; pt (Mix is empty (n1); surum finsts terest arrive; Node must - man (Node first, int item) temp = first temp = temp > knki pt 1 1 tem deleted = x d in ", find sinp); NODE temp, wil; temp= getrode(); pres (finst); temp -> info = item; return temps (1JUN== HULL) H sutwin temp; Node delete\_mos (Note first) cun firm; (11UN = 14HIL < 1000) dirter Node wer, pour; ever = cur-> link; if (first = NULL) cus >link = temp; pp ("list is emphysn"); return first; return first) of (forst-> HAR = = NULL) for (" item deleted is it in ", from > inpole survin NULL;

case & : pt 1" entor the "term @ pront and "

can 2: pt 1" enter the item @ sear\_end in

+ crest = item - forent ( first, item):

: ( mi); wife nave truen = tereit

case 3: first = delike-front (first)

ease A; first = delete-ouar (first);

( " y.d", 4 item);

et (" 1.d", 41 rem);

bruak;

preak.

break.

void display ( NODE first)

if (thous=NULL)
P+ (" list is empM In");

pf ("xd m", temp >info);

temp=temp>link)

for ("the examply temp = first; temp? = NULL;

None temp;

int main ()