Date ....

Expt. No. .... Page No. ....

```
NODE deleterear(NODE first) {
NODE cur, prev;
if (first == NULL) {
printf ("cannot delete \n");
return first;
}

if (first->link == NULL) {

printf ("Item deleted %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 is %d", cur->info);

 free (cur);
 prev->link = NULL;
return first;
}
```

Teacher's Signature : ____________

Date ....

Page No. ....

Expt. No. ....

```
NODE insertpos( int item, int pos , NODE first) {

NODE temp , cur, prev;
int count ;
temp = getnode();
temp->info = item ;
temp->link= NULL;
if (first ==NULL && pos = =1) {

return temp ;}

if (first = = NULL) {

printf ("Invalid\n");
return first ;
}

if (pos= =1) {

temp->link= first ;
first =temp;
return temp; }

count = 1 ;
prev= NULL;
cur = first;
while (cur != NULL && count != pos) {
    prev= cur ;
    cur = cur -> link; count ++ ; }
```

Teacher's Signature :

Expt. No. ....   Date ....

Page No. ....

```
if ( count = = pos) {
  prev-> link = temp;
  temp -> link = cur;
  return first;
  }
  printf (" Invalid pos \n");
  return first; }

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

Teacher's Signature : ____________

Expt. No. ....

Date ....

Page No. ....

```
if (count == pos)
{  prev -> link = cur -> link;
 printf ("item deleted is %d \n", cur -> );
   free(cur);
   return first;
 }

else
printf ("invalid     pos \n");
return   first;
}
void  display (NODE  first)
{    NODE  temp;
   if ( first == NULL)
{ printf ("Empty");}
 for ( temp = first ; temp != NULL; temp = temp -> li
  { printf ( %d \n", temp -> info);
        } }
```

Teacher's Signature : ....

Date ....

Expt. No. .... Page No. ....

```
void main()
{ int item, choice, pos;
   NODE first = NULL;
 for(;;)
 { printf ("\n 1. Insert front 2. Insert rear \n
          3. Insert at specified location \n 4. Delete
          front \n 5. Delete rear \n 6. Delete
          at specified location \n 7. Display
           8. Exit \n");

    printf ("Enter 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;
 case2  : printf (" enter the item at rear \n").
          scanf (" %d, & item).
          first = insertrear(first, item);
           break;
```

Teacher's Signature : ____________

Expt. No. ....   Date ....
                                                            Page No. ....

```
case 3 : printf (" enter insertion at location)
           scanf (" %d ", &item);
            printf (" position:\n");
             scanf (" %d", &pos);

          first = insertpos(item , pos , first );
           break ;

case 4 :    first= deletefront (first );
             break;

case 5   :   first := deleterear (first );
              break ;                       initialize
                                             ↖
case 6 :   printf (" Enter pos : \n");   int pos1;
scanf    (" %d , & pos1 ) .

  first  =  deletepos ( pos1 , first ) ;
     break ;

Case 7   :   display (first ) ;
 break ;

default  :   exit (0) ;
 break ;
}
}
}
```

Teacher's Signature : ____________