```
/**********************SUBHAS  NATH*******************/
/*****************************************************/
/************PROGRAM OF LINEAR LINKED LIST************/

#include<stdio.h>
#include<conio.h>
#include<stdlib.h>
#define NULL 0
struct link
 {
  int info;
  struct link *next;
 };
typedef struct link node;
void main()
 {
  node *head;
  clrscr();
  void create(node *);
  void show(node *);
  node *insert_fst(node *);
  node *insert_lst(node *);
  head=(node *)malloc(sizeof(node)); //CREATE STORAGE FOR FIRST NODE
  printf("\n\tCreate the linked list:\n");
  create(head);
  printf("\n\tDisplay the linked list:\n");
  show(head);
  printf("\n\n\tInsertion at 1st position.\n");
  head=insert_fst(head);
  show(head);
  printf("\n\n\tInsertion at last position.\n");
  head=insert_lst(head);
  show(head);
  getch();
 }

/********************CREATE LINKED LIST*********************/

void create(node *temp)
 {
  char ans;
  printf("\nEnter the info:-   ");
  scanf("%d",&temp->info);
  temp->next=NULL;
  printf("\nWant Another?  ");
  fflush(stdin); //CLEAR THE BUFFER
  ans=getchar();
  if(ans!='y')
   return;
  else
   {
    temp->next=(node *)malloc(sizeof(node));//CREATE STORAGE FOR NEXT NODE
    create(temp->next);  //RECURSIVE CALL TO CREATE NEXT NODE
   }
 }
```

```
/********************DISPLAY LINKED LIST********************/

void show(node *temp)
 {
  if(temp->next==NULL)
   printf("%d",temp->info); //DISPLAY LAST NODE
  else
   {
    printf("%d->",temp->info);//DISPLAY CURRENT INFO
    show(temp->next);   //RECURSIVE CALL TO DISPLAY NEXT INFO
   }
 }


/******************INSERTION AT FIRST POSITION***************/

node *insert_fst(node * temp)
 {
  node * list;
  list=(node *)malloc(sizeof(node));
  printf("\nEnter the element:-  ");
   scanf("%d",&list->info);
  list->next=temp;
  temp=list;
  return(temp);
 }



/******************INSERTION AT LAST POSITION*****************/

node *insert_lst(node *temp)
 {
  node *last,*first;
  first=temp;
  while(temp->next!=NULL)
   {
    temp=temp->next;
   }
  if(temp->next==NULL)
   {
    last=(node *)malloc(sizeof(node));
    printf("\nEnter the element:-  ");
     scanf("%d",&last->info);
    temp->next=last;
    last->next=NULL;
   }
  return(first);
 }
```