Write a program to implement singly linked list with the following operations.

a) Create a linked list.
b) Insertion of a node at first position, at any position and at end of the list.

Display the contents of the linked list.

```
#include <stdio.h>
#include <stdlib.h>

struct node{
   int info;
   struct node* next;
};

struct node* createlk(){
   struct node*p;
   struct node*start=NULL;
   struct node*last;
   int item;
   printf("enter -999 to exit\n");
   scanf("%d",&item);
   while(item!=-999){
      p=(struct node*)malloc(sizeof(struct node));
      p->info=item;
      if(start==NULL){
         p->next=NULL;
         start=p;
         last=p;
      }
```

```
        else{
          p->next=NULL;
          last->next=p;
          last=p;
        }
        scanf("%d",&item);
    }
    return start;

}

struct node* insertfirst(struct node*start,int item){
    struct node*p;
    p=(struct node*)malloc(sizeof(struct node));
    p->info=item;
    if(start==NULL){
      p->next=NULL;
      start=p;
    }
    else{
        p->next=start;
        start=p;
    }
    return start;
}

struct node* insertlast(struct node* start,int item){
    struct node* p,* last;
```

```
    p=(struct node*)malloc(sizeof(struct node));
    p->info=item;
    if(start==NULL){
       p->next=NULL;
       start=p;

    }
    else{
      last=start;
      while(last->next!=NULL){
        last=last->next;
      }
      p->next=NULL;
      last->next=p;



    }
    return start;

}

struct node* insertatposition(struct node* start,int item,int pos){
    struct node* p,* temp;
    p=(struct node*)malloc(sizeof(struct node));
    int size=1;
    p->info=item;
    if(pos==1){
       p->next=start;
```

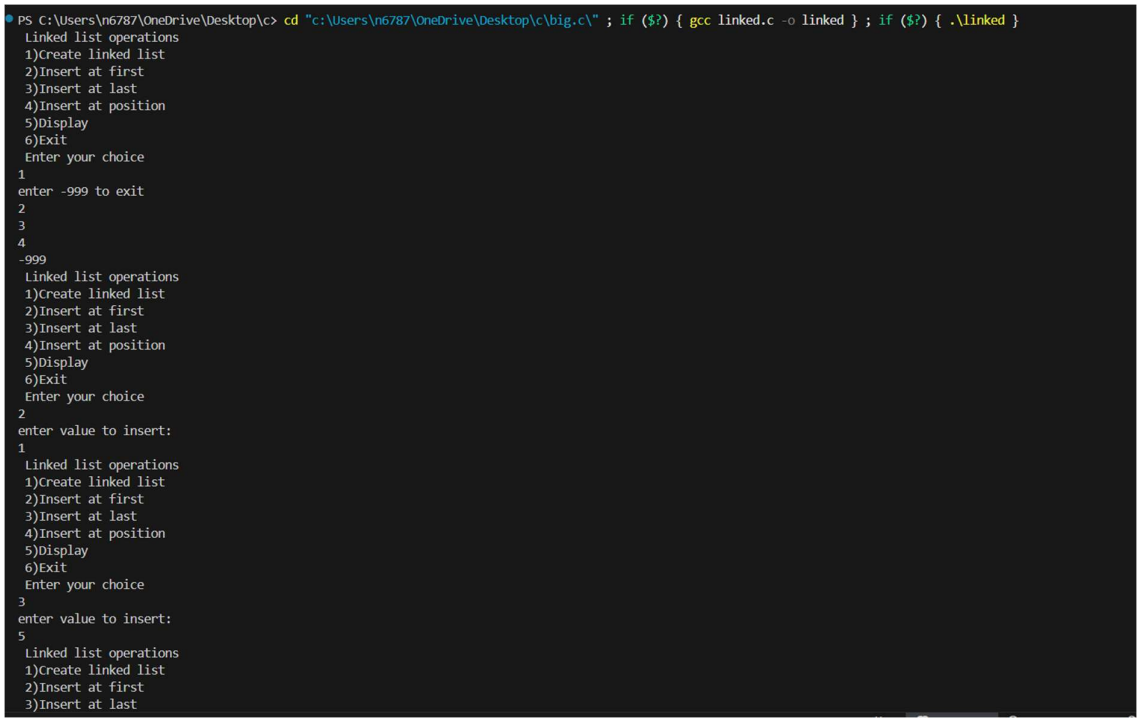
```
      start=p;
      return start;
    }

    else{
      temp=start;
      while(temp!=NULL ){
        if(size==pos-1){
          p->next=temp->next;
          temp->next=p;
          break;

        }
        size++;
        temp=temp->next;
      }
    }
    return start;
}

void displaylk(struct node*start){
  struct node*temp;
  if(start==NULL){
    printf("linked list is empty\n");
  }
  else{
    temp=start;
    printf("elements are\n");
```

```
    while(temp!=NULL){
        printf("%d\n",temp->info);
        temp=temp->next;
    }
  }
}

int main(){
    struct node* head=NULL;
    int choice;
    int val;
    while(1){
    printf(" Linked list operations \n");
    printf(" 1)Create linked list \n" );
    printf(" 2)Insert at first  \n" );
    printf(" 3)Insert at last  \n" );
    printf(" 4)Insert at position \n" );
    printf(" 5)Display \n" );
    printf(" 6)Exit \n ");

    printf("Enter your choice \n");
    scanf("%d",&choice);
    switch(choice){

        case 1:
        head=createlk();
        break;
```

```
case 2:
printf("enter value to insert: \n");
scanf("%d",&val);
head=insertfirst(head,val);
break;

case 3:
printf("enter value to insert: \n");
scanf("%d",&val);
head=insertlast(head,val);
break;

case 4:
printf("enter value to insert: \n");
scanf("%d",&val);
int pos;
printf("enter position to enter \n");
scanf("%d",&pos);
head=insertatposition(head,val,pos);
break;

case 5:
displaylk(head);
break;

case 6:
printf("Exiting program \n");
return 0;
```

```
        default:
        printf("Invalid choice \n");
    }

}
return 0;
}
```

Output:



```
 4)Insert at position
 5)Display
 6)Exit
 Enter your choice
4
enter value to insert:
7
enter position to enter
3
 Linked list operations
 1)Create linked list
 2)Insert at first
 3)Insert at last
 4)Insert at position
 5)Display
 6)Exit
 Enter your choice
5
elements are
1
2
7
3
4
5
 Linked list operations
 1)Create linked list
 2)Insert at first
 3)Insert at last
 4)Insert at position
 5)Display
 6)Exit
 Enter your choice
6
Exiting program
PS C:\Users\n6787\OneDrive\Desktop\c\big.c>
```