```
// Insert at begin
#include<stdio.h>
#include<stdlib.h>

struct Node{
    int data;
    struct Node * next;
}*new, *head,*tail, *temp;

 void linkedListTraversal(struct Node *temp)
 {
      while(temp!=NULL)
      {
          printf("%d",temp->data);
          temp=temp->next;
      }

  }

 // insert node at Begining
 struct Node * InsertAtBegin( struct Node * head, int data)
  {

      struct Node * new = (struct Node *) malloc(sizeof(struct Node));
      new->data=data;
       new->next=head;
       head=new;
       return head;
   }
  // create list
int main()
 {

 int value;
 char ch = 'y';
 int position;
 printf("create linked list\n");
while(ch =='y')
 {
  new = (struct Node *) malloc(sizeof(struct Node));
  printf("enter value to be inserted in linked list");
  scanf("%d",&value);
  new->data=value;
  if(head==NULL)
 {
    head=new;
    tail=new;
 }
 else
   {
        tail->next=new;
        tail=new;
   }

    fflush(stdin);
printf("Y- continue, N-exit");
scanf(" %c",&ch);
      }
  printf("enetr value to be inserted\n");
      scanf("%d",&value);
      head=InsertAtBegin(head, value);
   printf("Linked list \n");
    linkedListTraversal(head);
   return 0;

}

// insert at position in a singly linked list
#include<stdio.h>
#include<stdlib.h>

struct Node{
            int data;
            struct Node * next;
             }*new, *head,*tail, *temp;

 void linkedListTraversal(struct Node *temp)
 {
      while(temp!=NULL)
      {
          printf("%d",temp->data);
          temp=temp->next;
      }

  }
 // insert node at InsertAtPosition
 struct Node * InsertAtPosition( struct Node * head, int position, int data)
  {
       int i=1;
      struct Node * new = (struct Node *) malloc(sizeof(struct Node));
      new->data=data;
       printf("Enter position where to insert \n ");
       scanf("%d",&position);
       temp=head;
       while(i<position-1)
       {
            temp=temp->next;
            i++;
       }
       printf("%d",temp->data);
      new->next=temp->next;
      temp->next=new;
            return head;
  }
  // create list
int main()
 {

 int value;
 char ch = 'y';
 int position;
 printf("create linked list\n");
while(ch =='y')
 {
  new = (struct Node *) malloc(sizeof(struct Node));
  printf("enter value to be inserted in linked list");
  scanf("%d",&value);
  new->data=value;
  if(head==NULL)
      {
        head=new;
        tail=new;
       }
 else
        {
         tail->next=new;
         tail=new;
        }

        fflush(stdin);
printf("Y- continue, N-exit");
scanf(" %c",&ch);
}

  printf("enetr value to be inserted\n");
      scanf("%d",&value);
      head=InsertAtPosition(head, position, value);
   printf("Linked list \n");
    linkedListTraversal(head);
   return 0;

}
//insert in the end
```

```
struct Node{
    int data;
    struct Node * next;
}*new, *head,*tail, *temp;

 void linkedListTraversal(struct Node *temp)
 {
      while(temp!=NULL)
      {
          printf("%d",temp->data);
          temp=temp->next;
      }

  }

 // insert node at end
 struct Node * InsertAtEnd( struct Node * tail, int data)
  {

      struct Node * new = (struct Node *) malloc(sizeof(struct Node));
      new->data=data;
       tail->next=new;
       new->next=NULL;
       tail=new;
       return head;
   }
  // create list
int main()
 {
```

```
 int value;
 char ch = 'y';
 int position;
 printf("create linked list\n");
while(ch =='y')
 {
  new = (struct Node *) malloc(sizeof(struct Node));
  printf("enter value to be inserted in linked list");
  scanf("%d",&value);
  new->data=value;
  if(head==NULL)
 {
    head=new;
    tail=new;
 }
 else
   {
        tail->next=new;
        tail=new;
   }

    fflush(stdin);
printf("Y- continue, N-exit");
scanf(" %c",&ch);
      }

  printf("enetr value to be inserted\n");
      scanf("%d",&value);
      head=InsertAtEnd(tail, value);
   printf("Linked list \n");
    linkedListTraversal(head);
   return 0;


}
```