Lab -5

```
#include<stdio.h>
# include <stdlib.h>
struct node{
int info;
struct node* link;
};
struct node* first;
void insert_rear(){
struct node* temp;
temp = (struct node* )malloc(sizeof(struct node));
printf("Enter the integer\n");
scanf("%d", &temp -> info);
temp -> link = NULL};
 if (first == NULL){
     first = temp;
}
 else{
    struct node* p;
    p= first;
    while (p-> link != NULL) {
      p= p -> link
    }
   p-> link = temp;
 }
}
```

```
void insert-front(){
 struct node* temp;
 temp = (struct node*)malloc(sizeof (struct node));
 printf ("Enter the integer\n");
 scanf ("%d", & temp->info);
 temp -> link = NULL;
  if (first == NULL){
    first = temp;
  }
  else {
     temp -> link = first;
      first = temp;
 }
}
void insert_loc(){
struct node* temp;
struct node* r;
int loc, count=1;
temp = (struct node*) malloc (sizeof(struct node));
printf (" Enter location");
scanf ("%d", & loc);
printf ("Enter the integer\n");
scanf ("%d", & temp->info);
temp -> link = NULL;
 r = first;
```

```
while (loc> count ] {
   p= p ->link;
   count += 1;
 }
 temp->link = p -> link;
 p -> link = temp;
}
void display() {
struct node* temp;
if (first == NULL) {
     printf (" list is empty" );
}
else {
  temp = first;
   while (temp != NULL) {
    printf(" %d" , temp -> info );
    temp = temp -> link;
   }
 }
}
void main() {
int choice = 15;
while (choke != 0) {
printf("Enter your choice \n 0 Exit \n 1 Insert at the
end \n 2. Insert ine the front \n 3 insert at a
  Location \n  4. display the nodes \n");
scanf ("%d", &choice);
 case 0: {
  break;
 }
 case 1: {
  insert_rear();
  break;
 }
 case 2: {
  insert_front();
   break;
  }
 case 3: {
 insert_loc();
   break;
 }
 case 4: {
  display();
 break;
}
default: {
printf("wrong choice!\n");
 break;
}
}
}
```