1

Lab 8 – Queue using Linked list

IBM19CS006

```
#include <stdio.h>
#include <stdlib.h>
void insert ();
void display ();
void delete();
struct node
{  int data;
   struct node *next;
};
struct node *rear=NULL, *front=NULL;

int main() {
  int choice;
  do {
```

```
printf("\n1. Insert \n2. Display \n3. Delete \n4. Exit");
printf("Enter choice : ");
scanf("%d", &choice);
switch(choice)
{
case 1: insert();
    break;
case 2: display();
break;
case 3: delete();
 break;
 case 4: exit(0);
}
} while(choice !=4);
return 0;
}
void insert()
{
 struct node * newnode;
 newnode = (struct node *)malloc(sizeof(struct node));
 printf("Enter element : \n");
 scanf("%d", &newnode->data);
 newnode->next = NULL;
if(rear == NULL)
{ rear = newnode;
    front = newnode;
  }
else{
   rear->next = newnode;
```

```
rear = newnode;
    }
  }

void display()
{   struct node *temp;
  if(front == NULL)
{   printf("Queue is empty \n");
  return;
}
  temp=front;
  while(temp != NULL)
  {
    printf("%d\t", temp->data);
     temp=temp->next;
  }
}
void delete() {
if(front == NULL){
    printf("Queue is empty \n");
    return;
  }
  else{
  printf("Deleted element : %d", front->data);
  if(front == rear ){
      printf("Queue is empty \n");
      front = NULL; rear = NULL;
  }
  else front=front->next;
    }
  }
```