EX NO: 3 (c)	LINKED LIST IMPLEMENTATION OF QUEUE ADTSOPERATIONS

Aim:
To implement queue using linked list, we need to set the following things before implementing actual operations.

Algorithm:
Step 1: Include all theheader fileswhich are used in the program. And declareall the user defined functions.
Step 2: Define a ‘Node’ structure with two members data and next.
Step 3: Define two **Node** pointers ‘**front**’ and ‘**rear**’ and set both to **NULL**.
Step 4: Implement the **main** method by displaying Menu of list of operationsand make suitable function calls in the **main** method to perform user selected operation.

enQueue(value) - Inserting an element into the Queue

We can use the following steps to insert a new node into the queue...

- **Step 1:** Create a **newNode** with given value and set ‘**newNode ! next**’ to **NULL**.
- **Step 2:** Check whether queue is **Empty** (**rear** == **NULL**)
- **Step 3:** If it is **Empty** then, set **front** = **newNode** and **rear** = **newNode**.
- **Step 4:** If it is **Not Empty** then, set **rear ! next** = **newNode** and **rear** = **newNode**.

deQueue() - Deleting an Element from Queue

We can use the following steps to delete a node from the queue...

- **Step 1:** Check whether **queue** is **Empty** (**front == NULL**).
- **Step 2:** If it is **Empty**, then display **”Queue is Empty!!! Deletion is not possible!!!”** and terminate from the function
- **Step 3:** If it is **Not Empty** then, define a Node pointer ‘**temp**’ and set it to ‘**front**’.
- **Step 4:** Then set ‘**front** = **front ! next**’ and delete ‘**temp**’ (**free(temp)**).

display() - Displaying the elements of Queue

We can use the following steps to display the elements (nodes) of a queue...

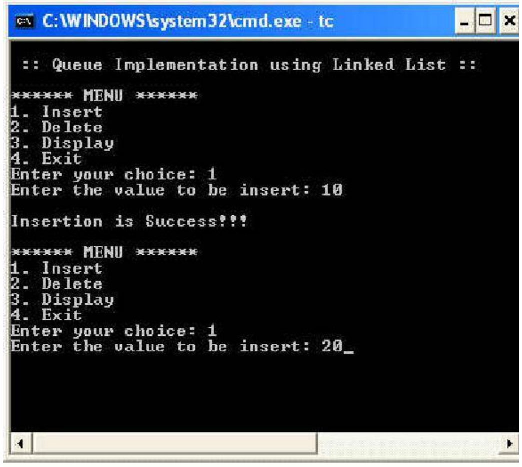
- **Step 1:** Check whether queue is **Empty** (**front** == **NULL**).
- **Step 2:** If it is **Empty** then, display **’Queue is Empty!!!’** and terminate thefunction.
- **Step 3:** If it is **Not Empty** then, define a Node pointer **’temp’** and initializewith **front**.
- **Step 4:** Display ‘**temp ! data** —>’ and move it to the next node. Repeat the sameuntil ‘**temp**’ reaches to ‘**rear**’ (**temp ! next** != **NULL**).
- **Step 4:** Finally! Display ‘**temp ! data** —> **NULL**’.

Program :

```
#include<stdio.h>
#include<conio.h>
struct Node
{
  int data;
  struct Node *next;
}*front = NULL,*rear = NULL;
void insert(int);
void delete();
void display();
void main()
{
  int choice, value;
  clrscr();
  printf(“\n:: Queue Implementation using Linked List ::\n”);
  while(1){
    printf(“\n****** MENU ******\n”);
    printf(“1. Insert\n2. Delete\n3. Display\n4. Exit\n”);
    printf(“Enter your choice: “);
    scanf(“%d”,&choice);
    switch(choice){
     case 1: printf(“Enter the value to be insert: “);
                 scanf(“%d”, &value);
                 insert(value);
                 break;
     case 2: delete(); break;
     case 3: display(); break;
     case 4: exit(0);
     default: printf(“\nWrong selection!!! Please try again!!!\n”);
    }}}
void insert(int value)
{
  struct Node *newNode;
  newNode = (struct Node*)malloc(sizeof(struct Node));
  newNode->data = value;
  newNode -> next = NULL;
  if(front == NULL)
    front = rear = newNode;
  else{
    rear -> next = newNode;
    rear = newNode;
  }printf(“\nInsertion is Success!!!\n”);}
void delete()
{
if(front == NULL)
    printf(“\nQueue is Empty!!!\n”);
  else{
    struct Node *temp = front;
    front = front -> next;
    printf(“\nDeleted element: %d\n”, temp->data);
    free(temp);
  }}
void display()
{
  if(front == NULL)
    printf(“\nQueue is Empty!!!\n”);
  else{
    strct Node *temp = front;
```

```
        while(temp->next != NULL){
         printf(“%d—>”,temp->data);
         temp = temp -> next;}
        printf(“%d—>NULL\n”,temp->data);}
    }
```

Output

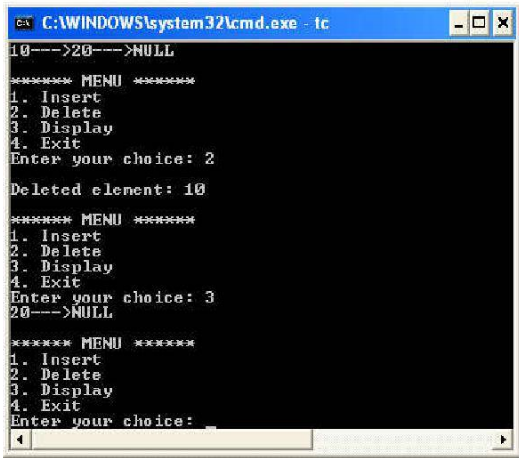


```
 :: Queue Implementation using Linked List ::

****** MENU ******
1. Insert
2. Delete
3. Display
4. Exit
Enter your choice: 1
Enter the value to be insert: 10

Insertion is Success!!!

****** MENU ******
1. Insert
2. Delete
3. Display
4. Exit
Enter your choice: 1
Enter the value to be insert: 20_
```



```
10--->20--->NULL

****** MENU ******
1. Insert
2. Delete
3. Display
4. Exit
Enter your choice: 2

Deleted element: 10

****** MENU ******
1. Insert
2. Delete
3. Display
4. Exit
Enter your choice: 3
20--->NULL

****** MENU ******
1. Insert
2. Delete
3. Display
4. Exit
Enter your choice: _
```

Result:

Thus the program for implementing queue using linked list is executed sucessfully & verified.