Zayd Ahmed
1BM21CS254

Queue Implementation using Linked list

Input:-

```
// PROGRAM TO IMPLEMENT QUEUE USING LL

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

struct NODE
{
    int data;
    struct NODE *link;
};

typedef struct NODE Node;
Node *front, *rear,*new;

void insert(){
    if(rear==NULL){
        front=(Node *)malloc(sizeof(Node));
        rear=front;
        printf("enter the element\n");
        scanf("%d",&front->data);
        return;
    }
    else{
        new=(Node *)malloc(sizeof(Node));
        rear->link=new;
        rear=new;
        printf("enter the element\n");
        scanf("%d",&new->data);
        new->link=NULL;
        return;
    }
}

void delete(){
    if(front==NULL){
        printf("empty QUEUE\n");
```

```
        return;
    }
    else{
        printf("deleted: %d\n",front->data);
        Node *temp;
        temp=front;
        front=front->link;
        free(temp);
    }
}

void display(){
    Node *temp;
    temp=front;
    do{
        printf("%d\n",temp->data);
        temp=temp->link;
    }while(temp!=NULL);
}

int main(){
    int choice;
    do{
        printf("1.insert\n2.delete\n3.display\n4.exit\n");
        scanf("%d",&choice);
        switch (choice)
        {
        case 1:
            insert();
            break;
        case 2:
            delete();
            break;
        case 3:
            display();
            break;
        case 4:
            exit(0);
        default:
            printf("enter a valid choice\n");
            break;
        }
    }while(choice!=4);
    return 0;
}
```

Output:-

```
1.insert
2.delete
3.display
4.exit
1
enter the element
10
1.insert
2.delete
3.display
4.exit
1
enter the element
20
1.insert
2.delete
3.display
4.exit
1
enter the element
30
1.insert
2.delete
3.display
4.exit
3
10
20
30
1.insert
2.delete
3.display
4.exit
2
deleted: 10
1.insert
2.delete
3.display
4.exit
3
20
30
1.insert
2.delete
3.display
4.exit
4


...Program finished with exit code 0
Press ENTER to exit console.
```