ADITHI GIRIMAJI
1BM19CS005

Lab - program-4:

```
#include<stdio.h>
#include<stdlib.h>
int front=-1;
int rear=-1;
int queue[10];
int MAX;
void Enque(int);
void Deque();
void display();
int main()
{ int choice, item;
  printf("Enter the maximum number of elements
  in the queue \n");
  scanf("%d",&MAX);
  do {
  printf("1. Insert into the queue \n");
  printf("2. Delete from the queue \n");
  printf("3. Display the contents of the
     queue \n");
  printf("4. Exit \n");
  printf("Enter your choice: \n");
  scanf("%d",&choice);
  switch(choice)
  { case 1: if((front==0 && rear==MAX-1)) || (front
                                    == rear+1))
      { printf("Queue is full.\n");
        } else
    { printf("Enter the element \n");
      scanf("%d", &item);
      Enque(item); }
```

Scanned with CamScanner

```
        break;
    case 2: Deque(); break;
    case 3: display();
            break;
    case 4: exit(0);
    }
} while (choice != 4);
    return 0;
}

void Enque (int de)
{
    rear = (rear+1) % MAX;
    queue[rear] = de;
    if (front == -1)
        front = 0;
    return;
}

void Deque()
{ int item;
    if ((front == -1) && (rear == -1))
    { printf("Queue is empty \n"); return;
    }
    else
    { item = queue[front];
        if (front == rear)
        { front = -1;
          rear = -1;
        } else
        { front = (front+1) % MAX;
        }
    } printf("Removed element from the queue is
            %d \n", item); return; }
```

Scanned with CamScanner

Scanned with CamScanner

```
void display()
{   int i;
    if((( front == -1) && rear == -1))
    {  printf (" Queue is empty \n");
       return;
    }
    else
    {  printf(" Queue contents: \n");
       for (i = front; i != rear; i = (i+1) % MAX)
       {  printf (" %d \t", queue[i]);
       }
       printf(" %d ", queue[rear]);
       printf ("\n");
       return;
    }
}
```

Scanned with CamScanner
Scanned with CamScanner