PROGRAM

```
#include <stdio.h>
#define MAX 10
int rear = -1,front= -1,top = -1;
int queue[MAX],stack[MAX];
void push(int data){
   if(top < MAX-1)
      stack[++top] = data;
   else
      printf("stack overflow\n");
}
int pop(){
  if(top > -1)
      return stack[top--];
  printf("stack is empty\n");
  return 0;

}
void enqueue(int data){
   if((rear+1)%MAX==front)
      printf("Queue is full\n");
   else{
      if(front == -1)
         front = 0;
      rear = (rear+1)%MAX;
      queue[rear] = data;
    }
}
int dequeue(){
   int temp = 0;
   if(rear == -1)
      printf("Queue is empty\n");
   else if(front == rear){
      temp =queue[front];
      front = -1; rear = -1;
   }
   else{
      temp = queue[front];
      front = (front+1)%MAX;
   }
   return temp;
}

void display(){
   if(rear == -1)
      printf("Queue is empty\n");
   else{
      int current = front;
      printf("queue:");
      while(rear!=current){
         printf("%d, ",queue[current]);
         current = (current+1)%MAX;}
      printf("%d\n",queue[current]);
   }
}
void reversequeue(){
   if(rear == -1)
      printf("\n");
   else{
      while(rear != -1)
         push(dequeue());
      while(top != -1)
         enqueue(pop());
   }
}
int main(){
   int choice,flag=1,data;
   while(flag){

printf("Enter\n1.enqueue\n2.dequeue\n3.reverse
queue\n4.exit\n");
      scanf("%d",&choice);
      switch(choice){
         case 1: printf("Enter data : ");
scanf("%d",&data);
                 enqueue(data); display();
                 break;
         case 2: printf("Data removed:
%d\n",dequeue());
                 display();
                 break;
         case 3: reversequeue();display();
                 break;
         case 4: flag=0; break;
         default:printf("Invalid Input");
      }
   }
}
```

OUTPUT

```
Enter
1.enqueue
2.dequeue
3.reverse queue
4.exit
1
Enter data : 5
queue:5
Enter
1.enqueue
2.dequeue
3.reverse queue
4.exit
1
Enter data : 2
queue:5, 2
Enter
1.enqueue
2.dequeue
3.reverse queue
4.exit
1
Enter data : 6
queue:5, 2, 6
Enter
1.enqueue
2.dequeue
3.reverse queue
4.exit
2
Data removed: 5
queue:2, 6
Enter
1.enqueue
2.dequeue
3.reverse queue
4.exit
1
Enter data : 9
queue:2, 6, 9
Enter
1.enqueue
2.dequeue
3.reverse queue
4.exit
3
queue:9, 6, 2
Enter
1.enqueue
2.dequeue
3.reverse queue
4.exit
1
Enter data : 1
queue:9, 6, 2, 1
Enter
1.enqueue
2.dequeue
3.reverse queue
4.exit
3
queue:1, 2, 6, 9
Enter
1.enqueue
2.dequeue
3.reverse queue
4.exit
4
```