```c
#include <stdio.h>
#define MAX_SIZE 10

typedef struct {
    int data[MAX_SIZE];
    int rear;
} Queue;

void initialize(Queue* q) {
    q->rear = -1;
}

int isFull(Queue* q) {
    return q->rear == MAX_SIZE - 1;
}

int isEmpty(Queue* q) {
    return q->rear == -1;
}

void enqueue(Queue* q, int value) {
    if (isFull(q)) {
        printf("Queue is Full\n");
        return;
    }
    q->rear++;
    q->data[q->rear] = value;
}

int dequeue(Queue* q) {
    if (isEmpty(q)) {
        printf("Queue is Empty\n");
        return -1;
    }

    int removed = q->data[0];

    // Left Shift
    for (int i = 0; i < q->rear; i++) {
        q->data[i] = q->data[i + 1];
    }

    q->rear--;
    return removed;
}

void display(Queue* q) {
    if (isEmpty(q)) {
        printf("Queue is Empty\n");
        return;
    }
```

```c
    printf("Queue: ");
    for (int i = 0; i <= q->rear; i++) {
        printf("%d ", q->data[i]);
    }
    printf("\n");
}

int main() {
    Queue q;
    initialize(&q);

    int choice, val;

    while (1) {
        printf("\n1-Insert\n2-Remove\n3-Display\n4-Exit\n");
        printf("Enter your choice: ");
        scanf("%d", &choice);

        switch (choice) {
            case 1:
                printf("Enter the number: ");
                scanf("%d", &val);
                enqueue(&q, val);
                break;

            case 2:
                val = dequeue(&q);
                if (val != -1) {
                    printf("Removed value: %d\n", val);
                }
                break;

            case 3:
                display(&q);
                break;

            case 4:
                return 0;

            default:
                printf("Invalid Choice\n");
        }
    }
```

```
1-Insert
2-Remove
3-Display
4-Exit
Enter your choice: 1
Enter the number: 20

1-Insert
2-Remove
3-Display
4-Exit
Enter your choice: 1
Enter the number: 30

1-Insert
2-Remove
3-Display
4-Exit
Enter your choice: 1
Enter the number: 40

1-Insert
2-Remove
3-Display
4-Exit
Enter your choice: 3
Queue: 20 30 40
```
}
```
1-Insert
2-Remove
3-Display
4-Exit
Enter your choice: 2
Removed value: 20

1-Insert
2-Remove
3-Display
4-Exit
Enter your choice: 3
Queue: 30 40
```