

Student Name: Chetan Ingale

PRN No.: 221111030

Course Name: C.S.E. (IoT CS BC)

Course code: CSL301

Year: S.E.

Semester: 3

Roll No.: 17

Experiment Evaluation Sheet

Experiment No.: 5

Experiment Name:

Write a program to implement the concept of
Circular Queue with Insert, Delete, Display and Exit
operations.

Sr No.	Evaluation Criteria	Marks (Out of 9)	Performance Date	Correction Date and Signature of Instructor
1	Experiment Performance			
2	Journal Performance			
3	Punctuality			
Total				

Code :

```
#include <stdio.h>
#define max 5
int front = -1, rear = -1;
int q[max];

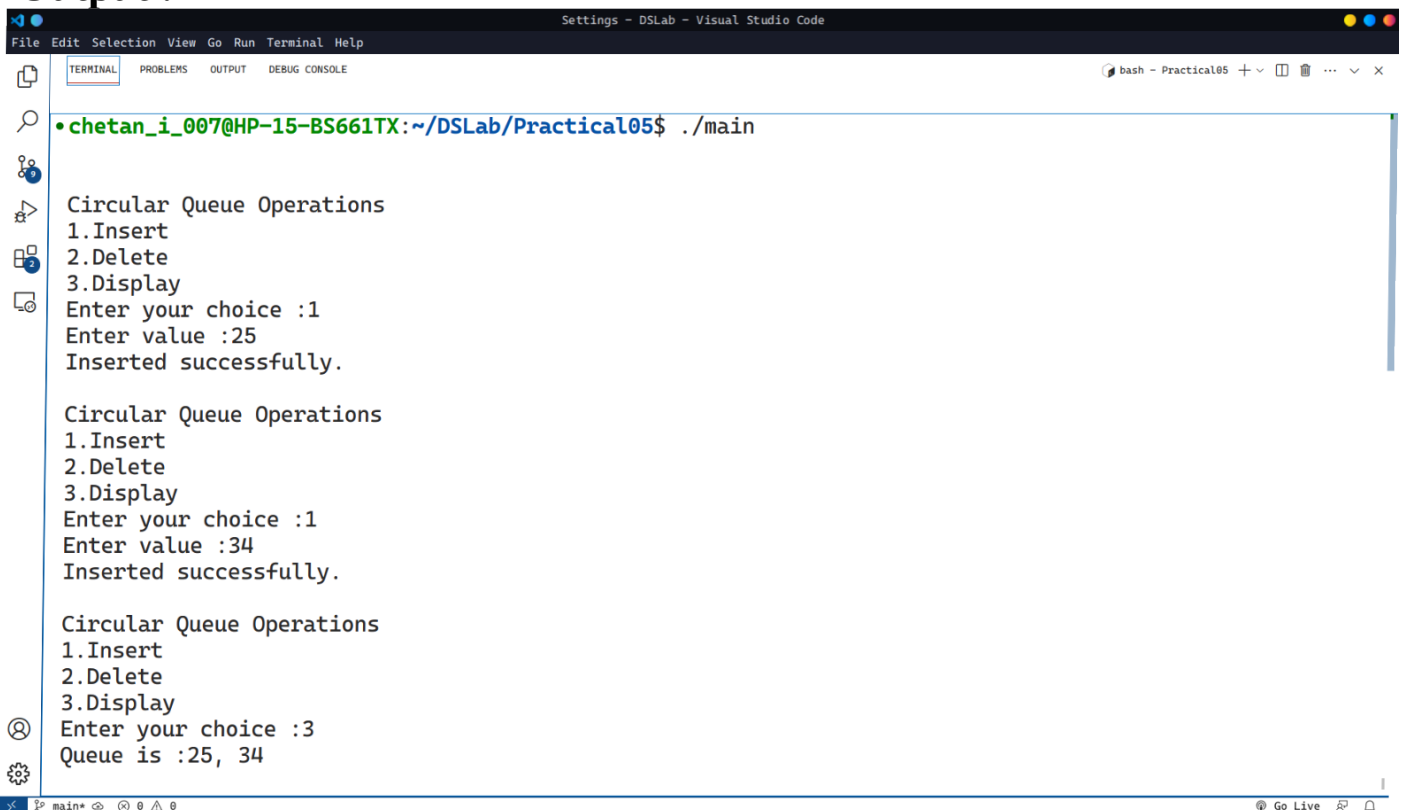
void enqueue() {
    int val;
    printf("Enter value :");
    scanf("%d", &val);
    if ((rear + 1) % max == front)
        printf("Queue is full.");
    else if (rear == -1) {
        rear = front = 0;
        q[rear] = val;
        printf("Inserted successfully.");
    }
    else {
        rear = (rear + 1) % max;
        q[rear] = val;
        printf("Inserted successfully.");
    }
}

int dequeue() {
    int val;
    if (front == -1) {
        printf("Queue is empty.");
        return -1;
    }
    else if (front == rear) {
        int val = q[front];
        front = rear = -1;
        printf("Deleted value : %d", val);
        return val;
    }
    else {
        val = q[front];
        front = (front + 1) % max;
        printf("Deleted value : %d", val);
        return val;
    }
}

void display() {
    int i;
    if (front == -1)
        printf("Queue is empty.");
    else {
        printf("Queue is :");
        for (i = front; i != rear; i = (i + 1) % max) {
            printf("%d, ", q[i]);
        }
        printf("%d", q[i]);
    }
}
```

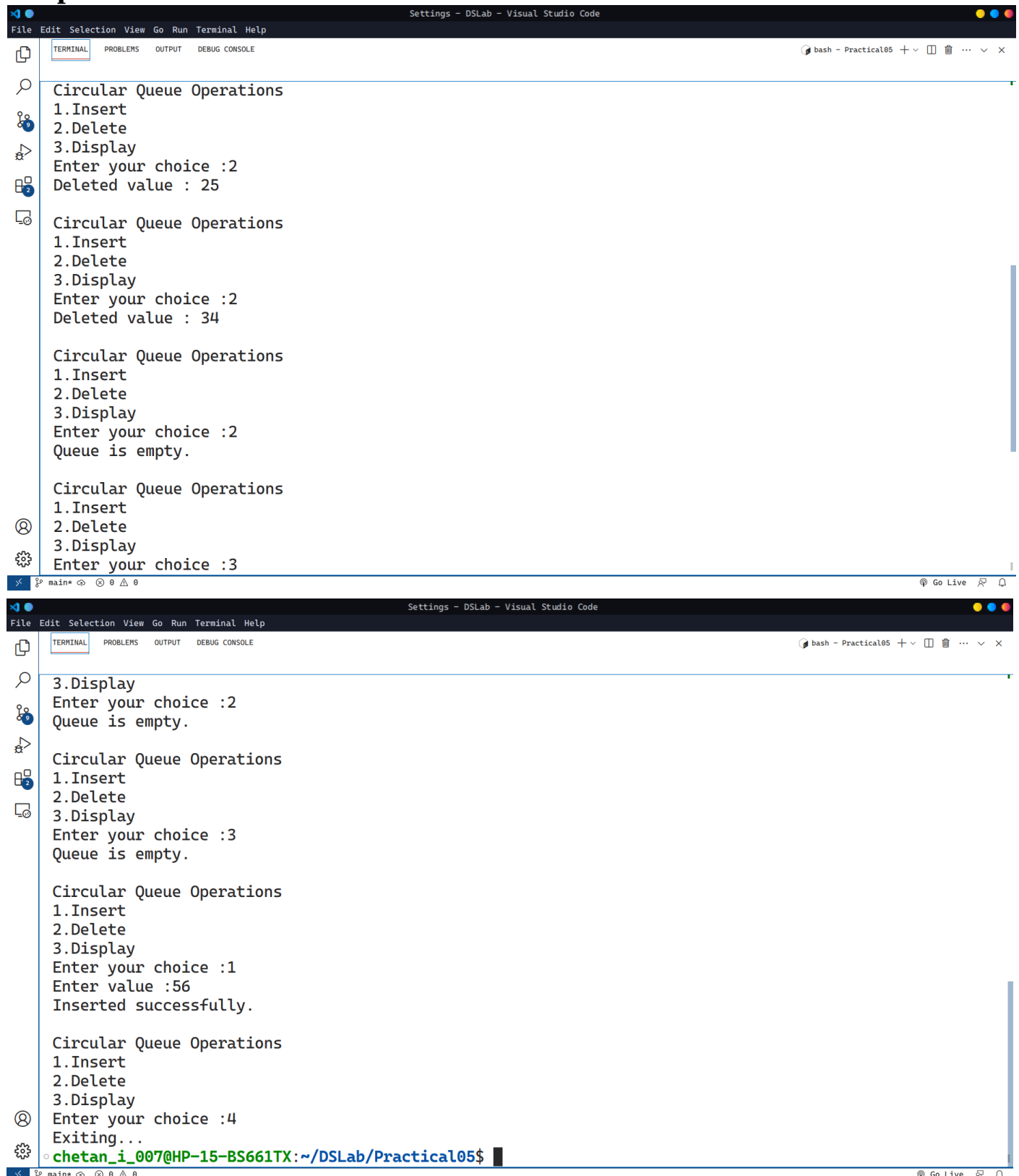
Code :

```
}  
}  
  
int main() {  
    int choice;  
    do {  
        printf("\n\nCircular Queue Operations\n");  
        printf("1.Insert\n");  
        printf("2.Delete\n");  
        printf("3.Display\n");  
        printf("Enter your choice :");  
        scanf("%d", &choice);  
        switch (choice) {  
            case 1:  
                enqueue();  
                break;  
            case 2:  
                dequeue();  
                break;  
            case 3:  
                display();  
                break;  
            case 4:  
                printf("Exiting...\n");  
                break;  
        }  
    }  
    while (choice != 4);  
    return 0;  
}
```

Output :

```
Settings - DSLab - Visual Studio Code  
File Edit Selection View Go Run Terminal Help  
TERMINAL PROBLEMS OUTPUT DEBUG CONSOLE  
bash - Practical05 + - □ ... v X  
• chetan_i_007@HP-15-BS661TX:~/DSLab/Practical05$ ./main  
  
Circular Queue Operations  
1.Insert  
2.Delete  
3.Display  
Enter your choice :1  
Enter value :25  
Inserted successfully.  
  
Circular Queue Operations  
1.Insert  
2.Delete  
3.Display  
Enter your choice :1  
Enter value :34  
Inserted successfully.  
  
Circular Queue Operations  
1.Insert  
2.Delete  
3.Display  
Enter your choice :3  
Queue is :25, 34
```

Output :



```
Settings - DSLab - Visual Studio Code
File Edit Selection View Go Run Terminal Help
TERMINAL PROBLEMS OUTPUT DEBUG CONSOLE
bash - Practical05 + - [ ] ... v x

Circular Queue Operations
1.Insert
2.Delete
3.Display
Enter your choice :2
Deleted value : 25

Circular Queue Operations
1.Insert
2.Delete
3.Display
Enter your choice :2
Deleted value : 34

Circular Queue Operations
1.Insert
2.Delete
3.Display
Enter your choice :2
Queue is empty.

Circular Queue Operations
1.Insert
2.Delete
3.Display
Enter your choice :3

3.Display
Enter your choice :2
Queue is empty.

Circular Queue Operations
1.Insert
2.Delete
3.Display
Enter your choice :3
Queue is empty.

Circular Queue Operations
1.Insert
2.Delete
3.Display
Enter your choice :1
Enter value :56
Inserted successfully.

Circular Queue Operations
1.Insert
2.Delete
3.Display
Enter your choice :4
Exiting...
chetan_i_007@HP-15-BS661TX:~/DSLab/Practical05$
```

Conclusion :

Through this experiment we have learnt about how to implement a Circular Queue using the C language. Various operations like enqueue, dequeue, isfull, and isempty are applied on the circular queue. This experiment helps us in using circular queue as a data structure for further reference.