

### 13. Queue Operations

Aim:

To write a C program for queue operations using arrays.

Algorithm:

1. Start program.
2. Define array queue[], front, and rear.
3. Implement enqueue, dequeue, and display.
4. End.

Code:

```
#include <stdio.h>
```

```
#define SIZE 5
```

```
int queue[SIZE], front = -1, rear = -1;
```

```
void enqueue(int value) {  
    if (rear == SIZE - 1)  
        printf("Queue Overflow\n");  
    else {  
        if (front == -1) front = 0;  
        queue[++rear] = value;  
    }  
}
```

```
void dequeue() {  
    if (front == -1 || front > rear)  
        printf("Queue Underflow\n");  
    else  
        printf("Dequeued %d\n", queue[front++]);  
}
```

```
void display() {  
    if (front == -1 || front > rear)  
        printf("Queue is empty\n");  
    else {  
        printf("Queue: ");  
        for (int i = front; i <= rear; i++)  
            printf("%d ", queue[i]);  
        printf("\n");  
    }  
}
```

```
int main() {  
    enqueue(10);  
    enqueue(20);  
    enqueue(30);  
    display();  
    dequeue();  
    display();  
    return 0;  
}
```

Sample Output:

```
Queue: 10 20 30  
Dequeued 10  
Queue: 20 30  
  
=== Code Execution Successful ===
```

Result:

Queue operations were successfully implemented.