Experiment 03\double_ended_queue.c

```c
#include <stdio.h>
#define MAX 100

int deque[MAX];
int front = -1, rear = -1;

int isFull() {
    return (front == 0 && rear == MAX - 1) || (front == rear + 1);
}

int isEmpty() {
    return front == -1;
}

void insertFront(int value) {
    if (isFull()) {
        printf("Deque Overflow\n");
        return;
    }
    if (isEmpty()) {
        front = rear = 0;
    } else if (front == 0) {
        front = MAX - 1;
    } else {
        front--;
    }
    deque[front] = value;
    printf("%d inserted at front\n", value);
}

void insertRear(int value) {
    if (isFull()) {
        printf("Deque Overflow\n");
        return;
    }
    if (isEmpty()) {
        front = rear = 0;
    } else if (rear == MAX - 1) {
        rear = 0;
    } else {
        rear++;
    }
    deque[rear] = value;
    printf("%d inserted at rear\n", value);
}

void deleteFront() {
    if (isEmpty()) {
```

```c
        printf("Deque Underflow\n");
        return;
    }
    printf("%d deleted from front\n", deque[front]);
    if (front == rear) {
        front = rear = -1;
    } else if (front == MAX - 1) {
        front = 0;
    } else {
        front++;
    }
}

void deleteRear() {
    if (isEmpty()) {
        printf("Deque Underflow\n");
        return;
    }
    printf("%d deleted from rear\n", deque[rear]);
    if (front == rear) {
        front = rear = -1;
    } else if (rear == 0) {
        rear = MAX - 1;
    } else {
        rear--;
    }
}

void display() {
    if (isEmpty()) {
        printf("Deque is empty\n");
        return;
    }
    printf("Deque: ");
    int i = front;
    while (1) {
        printf("%d ", deque[i]);
        if (i == rear)
            break;
        i = (i + 1) % MAX;
    }
    printf("\n");
}

int main() {
    int choice, value;
    while (1) {
        printf("\n1. Insert Front  2. Insert Rear  3. Delete Front  4. Delete Rear  5. Display  6. Exit\n");
        printf("Enter choice: ");
```

```c
        scanf("%d", &choice);
        switch(choice) {
            case 1:
                printf("Enter value: ");
                scanf("%d", &value);
                insertFront(value);
                break;
            case 2:
                printf("Enter value: ");
                scanf("%d", &value);
                insertRear(value);
                break;
            case 3:
                deleteFront();
                break;
            case 4:
                deleteRear();
                break;
            case 5:
                display();
                break;
            case 6:
                return 0;
            default:
                printf("Invalid choice\n");
        }
    }
}

```