

```

/*
 * MGP23CS069
 * implement double ended queue using array
 */

#include<stdio.h>

int size = 10;
int front = -1, rear = -1;
int value;
int deque[10];

void enqueueFront(int value) {
    if (front == size) {
        printf("\nQueue Overflow");
    }
    else if (front == -1 && rear == -1) {
        front = 0;
        rear = 0;
        deque[front] = value;
    }
    else {
        front = front - 1;
        deque[front] = value;
    }
}

void enqueueRear(int value) {
    if (rear == size) {
        printf("\nQueue Overflow");
    }
    else if (rear == -1 && front == -1) {
        rear = 0;
        front = 0;
        deque[rear] = value;
    }
    else {
        rear = rear + 1;
        deque[rear] = value;
    }
}

void dequeueFront() {
    if (front == size) {
        printf("\nDequeue not possible.");
    }
    else {
        front = front + 1;
    }
}

void dequeueRear() {
    if (rear == -1) {
        printf("\nQueue Underflow");
    }
    else {
        rear = rear - 1;
    }
}

void printQueue() {
    if (front == -1 && rear == -1)
        printf("\nQueue Empty\n");
}

```

```

        else {
            printf("\n");
            for (int i = front; i <= rear; i++)
                printf("%d", deque[i]);
            printf("\n");
        }
    }

int main()
{
    int flag = 1;
    printQueue();
    while (flag == 1){
        int choice;
        printf("Enter your choice: ");
        printf("\n1.EnqueueRear\t2.EnqueueFront\t3.DequeueRear\t4.DequeueFront");
        scanf("%d", &choice);

        switch(choice) {
            case 1:
                printf("\nEnter the value to enqueue:");
                scanf("%d", &value);
                enqueueRear(value);
                break;

            case 2:
                printf("\nEnter the value to enqueue:");
                scanf("%d", &value);
                enqueueFront(value);
                break;

            case 3:
                dequeueRear();
                break;

            case 4:
                dequeueFront();
                break;

            case 5:
                flag = 0;
                break;

            default:
                printf("\ninvalid input");
                break;
        }
        printQueue();
    }
}

```