#include <stdio.h>

#define SIZE 5

int queue[SIZE];

int front = -1, rear = -1;

int isFull() {

return ((front == 0 && rear == SIZE - 1) || (front == rear + 1));

}

int isEmpty() {

return (front == -1);

}

void enqueue(int value) {

if (isFull()) {

printf("Queue is FULL! Cannot insert %d\n", value);

return;

}

if (front == -1)

front = 0;

rear = (rear + 1) % SIZE;

queue[rear] = value;

printf("%d inserted into queue.\n", value);

}

void dequeue() {

if (isEmpty()) {

printf("Queue is EMPTY! Cannot dequeue.\n");

return;

}

int deletedValue = queue[front];

if (front == rear) {

// Queue becomes empty

front = rear = -1;

} else {

front = (front + 1) % SIZE;

}

printf("%d deleted from queue.\n", deletedValue);

}

void display() {

if (isEmpty()) {

printf("Queue is EMPTY!\n");

return;

}

printf("Queue elements: ");

int i = front;

while (1) {

printf("%d ", queue[i]);

if (i == rear) break;

i = (i + 1) % SIZE;

}

printf("\n");

}

int main() {

int choice, value;

while (1) {

printf("\n--- Circular Queue Menu ---\n");

printf("1. Enqueue (Insert)\n");

printf("2. Dequeue (Delete)\n");

printf("3. Display\n");

printf("4. Exit\n");

printf("Enter your choice: ");

scanf("%d", &choice);

switch (choice) {

case 1:

printf("Enter value to insert: ");

scanf("%d", &value);

enqueue(value);

break;

case 2:

dequeue();

break;

case 3:

display();

break;

case 4:

printf("Exiting program.\n");

return 0;

default:

printf("Invalid choice! Please try again.\n");

}

}

}