```
#include <stdio.h>
#define SIZE 5

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

int isFull() {
   return (rear == SIZE - 1);
}

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

void enqueue(int value) {
   if (isFull()) {
```

```
        printf("Queue is full! Cannot enqueue %d.\n", value);
    } else {
        if (front == -1) {
            front = 0;
        }
        rear++;
        queue[rear] = value;
        printf("%d enqueued to queue.\n", value);
    }
}

int dequeue() {
    if (isEmpty()) {
        printf("Queue is empty! Cannot dequeue.\n");
        return -1;
    } else {
        int value = queue[front];
        front++;
        if (front > rear) {
            front = rear = -1;
        }
        printf("%d dequeued from queue.\n", value);
        return value;
    }
}
```

```
void traverse() {
    if (isEmpty()) {
        printf("Queue is empty!\n");
    } else {
        printf("Queue elements: ");
        for (int i = front; i <= rear; i++) {
            printf("%d ", queue[i]);
        }
        printf("\n");
    }
}

int main() {
    int choice, value;
    while (1) {
        printf("\nQueue Operations:\n");
        printf("1. Enqueue\n");
        printf("2. Dequeue\n");
        printf("3. Traverse\n");
        printf("4. Exit\n");
        printf("Enter your choice: ");
        scanf("%d", &choice);
```

```
        switch (choice) {
            case 1:
                printf("Enter the value to enqueue: ");
                scanf("%d", &value);
                enqueue(value);
                break;
            case 2:
                dequeue();
                break;
            case 3:
                traverse();
                break;
            case 4:
                printf("Exiting program.\n");
                return 0;
            default:
                printf("Invalid choice! Please try again.\n");
        }
    }
    return 0;
}
```