```
#include <iostream>
using namespace std;
// Node Structure
struct Node
{
    int data;
    Node *next;
};
// Queue class
class Queue
{
private:
    Node *front, *rear;

public:
    // Constructor
    Queue()
    {
        front = NULL;
        rear = NULL;
    }

    // Enqueue function
    void enqueue(int value)
    {
        Node *newNode = new Node();

        if (!newNode)
        {
            cout << "Heap Overflow! Cannot allocate memory." << endl;
            return;
        }
        newNode->data = value;
        newNode->next = NULL;
        if (front == NULL)
        {
            front = rear = newNode;
        }
        else
        {
            rear->next = newNode;
            rear = newNode;
        }
```

```
    cout << value << " enqueued to queue." << endl;
}

// Dequeue function
void dequeue()
{
    if (front == NULL)
    {
        cout << "Error: Queue Underflow (Empty)." << endl;
        return;
    }
    Node *temp = front;
    cout << front->data << " dequeued from queue." << endl;
    front = front->next;
    if (front == NULL)
    {
        rear = NULL;
    }
    delete temp;
}

// Peek function
void peek()
{
    if (front == NULL)
    {
        cout << "Queue is Empty." << endl;
        return;
    }
    cout << "Front element is: " << front->data << endl;
}

// Display function
void display()
{
    if (front == NULL)
    {
        cout << "Queue is Empty." << endl;
        return;
    }

    Node *temp = front;
```

```
        cout << "Queue: ";
        while (temp != NULL)
        {
            cout << temp->data << " -> ";
            temp = temp->next;
        }
        cout << "NULL" << endl;
    }
};
// Main function
int main()
{
    Queue q;
    int choice, val;

    do
    {
        cout << "\n--- LINKED LIST QUEUE MENU ---" << endl;
        cout << "1. Enqueue" << endl;
        cout << "2. Dequeue" << endl;
        cout << "3. Peek" << endl;
        cout << "4. Display" << endl;
        cout << "5. Exit" << endl;
        cout << "Enter choice: ";
        cin >> choice;

        switch (choice)
        {
        case 1:
            cout << "Enter value: ";
            cin >> val;
            q.enqueue(val);
            break;
        case 2:
            q.dequeue();
            break;
        case 3:
            q.peek();
            break;
        case 4:
            q.display();
            break;
        case 5:
```

```
            cout << "Exiting..." << endl;
            break;
        default:
            cout << "Invalid choice." << endl;
        }
    } while (choice != 5);

    return 0;
}
```