```cpp
#include <iostream>

class Node {
public:
    int data;
    Node* next;

    Node(int value) : data(value), next(nullptr) {}
};

class SinglyLinkedList {
private:
    Node* head;

public:
    SinglyLinkedList() : head(nullptr) {}

    void insertAtBeginning(int value) {
        Node* newNode = new Node(value);
        newNode->next = head;
        head = newNode;
    }

    void insertAtEnd(int value) {
        Node* newNode = new Node(value);
        if (head == nullptr) {
            head = newNode;
        } else {
            Node* current = head;
            while (current->next != nullptr) {
                current = current->next;
            }
            current->next = newNode;
        }
    }

    void deleteNode(int value) {
        Node* current = head;
        Node* prev = nullptr;

        while (current != nullptr && current->data != value) {
            prev = current;
            current = current->next;
        }

        if (current != nullptr) {
            if (prev != nullptr) {
                prev->next = current->next;
            } else {
```

```
50                  head = current->next;
51              }
52              delete current;
53          }
54      }
55
56      void display() {
57          Node* current = head;
58          while (current != nullptr) {
59              std::cout << current->data << " ";
60              current = current->next;
61          }
62          std::cout << std::endl;
63      }
64  };
65
66  int main() {
67      SinglyLinkedList sll;
68
69      sll.insertAtEnd(10);
70      sll.insertAtEnd(20);
71      sll.insertAtEnd(30);
72
73      std::cout << "Singly Linked List: ";
74      sll.display();
75
76      sll.insertAtBeginning(20);
77      std::cout << "Singly Linked List after inserting at the beginning: ";
78      sll.display();
79
80      sll.deleteNode(10);
81      std::cout << "Singly Linked List after deleting : ";
82      sll.display();
83
84      return 0;
85  }
86
```