

```
1  #include <iostream>
2
3  class Node {
4  public:
5      int data;
6      Node* prev;
7      Node* next;
8
9      Node(int value) : data(value), prev(nullptr), next(nullptr) {}
10 };
11
12 class DoublyLinkedList {
13 private:
14     Node* head;
15     Node* tail;
16
17 public:
18     DoublyLinkedList() : head(nullptr), tail(nullptr) {}
19
20     void insertAtBeginning(int value) {
21         Node* newNode = new Node(value);
22         if (head == nullptr) {
23             head = tail = newNode;
24         } else {
25             newNode->next = head;
26             head->prev = newNode;
27             head = newNode;
28         }
29     }
30
31     void insertAtEnd(int value) {
32         Node* newNode = new Node(value);
33         if (tail == nullptr) {
34             head = tail = newNode;
35         } else {
36             newNode->prev = tail;
37             tail->next = newNode;
38             tail = newNode;
39         }
40     }
41
42     void deleteNode(int value) {
43         Node* current = head;
44
45         while (current != nullptr) {
46             if (current->data == value) {
47                 if (current->prev != nullptr) {
48                     current->prev->next = current->next;
49                 } else {
```

```
50         head = current->next;
51     }
52
53     if (current->next != nullptr) {
54         current->next->prev = current->prev;
55     } else {
56         tail = current->prev;
57     }
58
59     delete current;
60     return;
61 }
62
63     current = current->next;
64 }
65 }
66
67 void display() {
68     Node* current = head;
69     while (current != nullptr) {
70         std::cout << current->data << " ";
71         current = current->next;
72     }
73     std::cout << std::endl;
74 }
75 };
76
77 int main() {
78     DoublyLinkedList dll;
79
80     dll.insertAtEnd(10);
81     dll.insertAtEnd(20);
82     dll.insertAtEnd(30);
83
84     std::cout << "Doubly Linked List: ";
85     dll.display();
86
87     dll.insertAtBeginning(5);
88     std::cout << "Doubly Linked List after inserting at the beginning: ";
89     dll.display();
90
91     dll.deleteNode(10);
92     std::cout << "Doubly Linked List after deleting : ";
93     dll.display();
94
95     return 0;
96 }
97
```