

```
"""program to demonstrate doubly linked list"""
class Node:
    def __init__(self, data):
        self.data = data
        self.prev = None
        self.next = None

class DoublyLinkedList:
    def __init__(self):
        self.head = None

    def append(self, data):
        new_node = Node(data)
        if not self.head:
            self.head = new_node
        else:
            current = self.head
            while current.next:
                current = current.next
            current.next = new_node
            new_node.prev = current

    def prepend(self, data):
        new_node = Node(data)
        if not self.head:
            self.head = new_node
        else:
            new_node.next = self.head
            self.head.prev = new_node
            self.head = new_node

    def delete(self, data):
        current = self.head
        while current:
            if current.data == data:
                if current.prev:
                    current.prev.next = current.next
                else:
                    self.head = current.next

                if current.next:
                    current.next.prev = current.prev
                return
            current = current.next

    def display(self):
        current = self.head
        while current:
            print(current.data, end=" ")
            current = current.next
        print()

# Example usage:
dll = DoublyLinkedList()
```

```
dll.append(1)
dll.append(2)
dll.append(3)
dll.prepend(0)
dll.display()  # Output: 0 1 2 3
```

```
dll.delete(2)
dll.display()  # Output: 0 1 3
```

```
output
0 1 2 3
0 1 3
```