

Übung 8

Doppelt verkettete Listen unter Java

1. Programmieren Sie die klassische **public Node**-Struktur mit einem **String data** und einem **Node next**- sowie einem **Node prev**-Element als Java-Klasse (wie in der Vorlesung).
Hinweis: **next** zeigt immer auf das jeweils nächste, **prev** auf das jeweilige Vorgänger-Element, wobei diese Zeiger immer auf **null** gesetzt werden müssen, wenn sie nicht existieren.
2. Schreiben Sie einen Konstruktor für **Node**, der **prev** und **next** auf den Wert **null** setzt, und den **String data** auf den Leerstring "" setzt.
3. Fügen Sie zur Klasse **Node** eine Methode **public void remove()** ohne Parameter hinzu, die das aktuelle Element/Objekt entfernt.
4. Fügen Sie zur Klasse **Node** eine Methode **public void add(Node newnode)** hinzu, die das neue Element **newnode** hinter das aktuelle Element in die Liste einbaut.
5. Fügen Sie der Klasse **Node** eine Methode **public Node find(String searchtext)** hinzu, die in einer vorhandenen Liste von **Node**-Elementen sämtliche Elemente nach dem angegebenen Text **searchtext** durchsucht, bis entweder der Text gefunden, oder das Ende der Liste erreicht wurde. Die gefundene **Node** soll zurückgegeben werden, oder **null**, falls nichts gefunden wurde.

