

Kontrollfragen A

1. Vergleichen Sie Liste und Array.

Arrays

- die Elemente werden via Index angesprochen
- die Elemente liegen hintereinander im Speicher
- Arrays sind statisch, d.h. Arrays können ihre Größe nicht ändern
- der Zugriff auf ein bestimmtes Element kann direkt erfolgen
- bei einer Suche muss jedes Element des Arrays durchsucht werden, bis der Eintrag gefunden ist
- beim Einfügen oder Löschen eines Elementes müssen alle nachfolgenden Elemente im Array verschoben werden

Liste

- die Elemente werden über Zeiger miteinander verbunden
- die Elemente liegen an unterschiedlichen Stellen im Speicher
- Listen sind dynamisch, d.h. Listen können ihre Größe problemlos ändern
- der Zugriff auf ein bestimmtes Element kann nicht direkt erfolgen
- ausgehend vom ersten Element muss die Liste bis zum gewünschten Element durchgegangen werden
- bei einer Suche muss jedes Element des Arrays durchsucht werden, bis der Eintrag gefunden ist
- beim Einfügen oder Löschen eines Elementes müssen nur die Zeiger für Verkettung geändert werden.

Es gibt einfach und doppelt verkettete Liste.

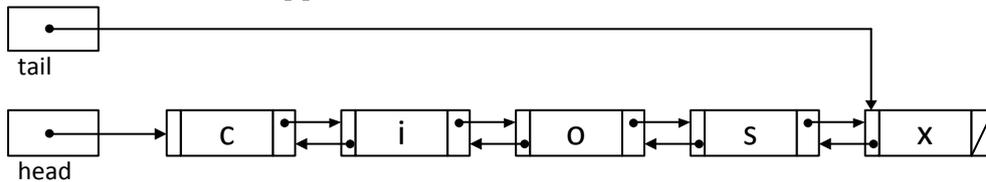
- Einfach verkettete Listen erlauben nur eine Bewegung vom ersten Element zum nächsten.
- Doppelt verkettete Listen erlauben auch eine Bewegung in die andere Richtung (prev & next).

2. Vergleichen Sie einfach und doppelt verkettete Liste.

Einfache Liste: einfacheres / schnelleres einfügen resp. löschen aus der Liste

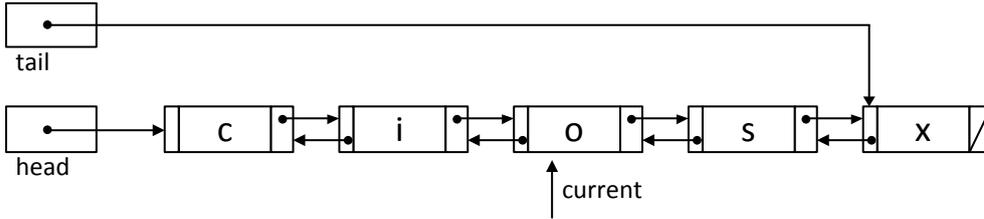
Verkettete Liste: prev / next Element ist bekannt

3. Zeichnen Sie eine doppelt verkettete Liste.

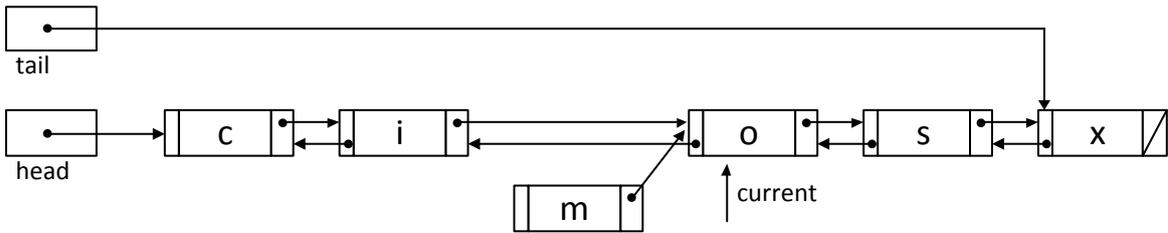


4. Führen Sie das Einfügen eines Elements schrittweise aus. Erstellen Sie eine detaillierte Skizze.

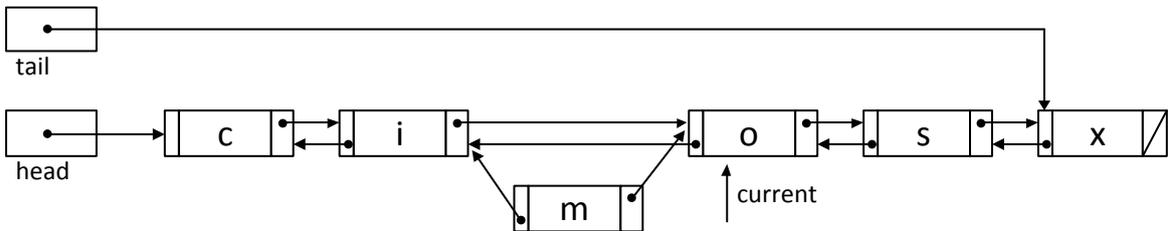
Position suchen wo das neue Element eingefügt werden muss:



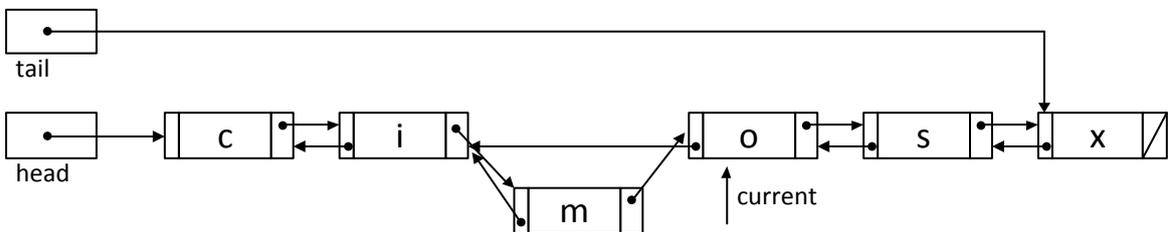
newNode.setNext(current);



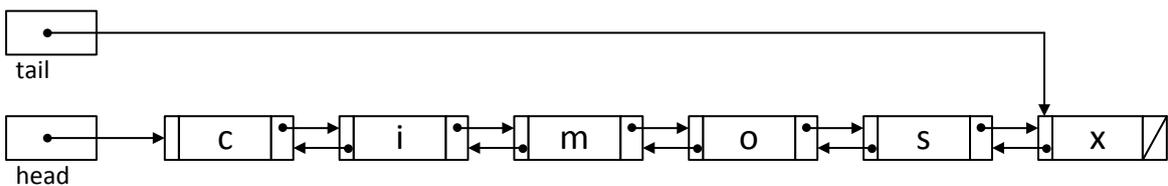
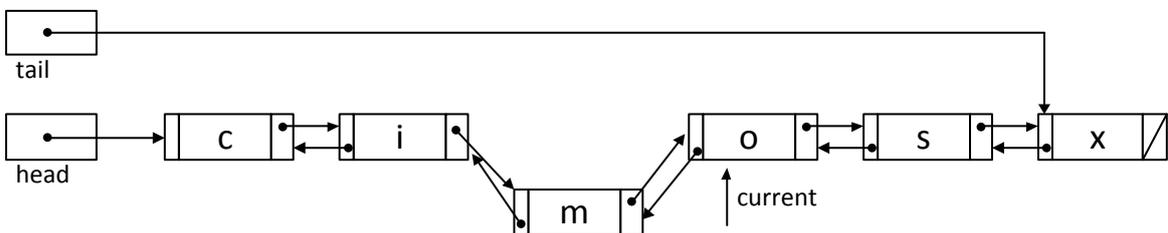
newNode.setPrev(current.getNext());



current.getPrev().setNext(newNode);

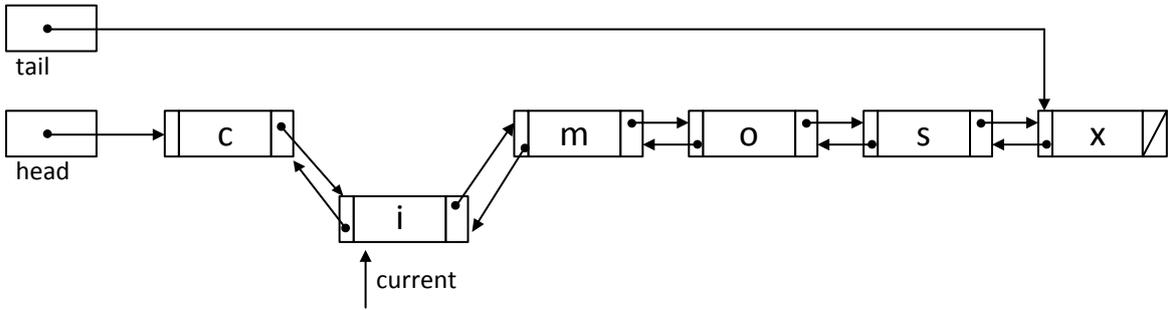
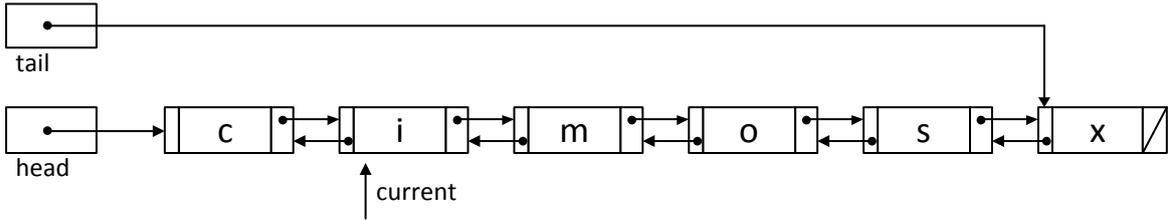


current.setPrev(newNode);

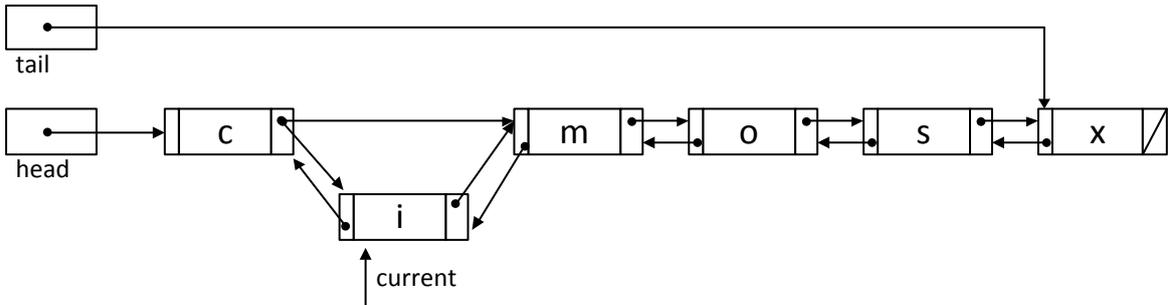


5. Führen Sie das Entfernen eines Elements schrittweise aus.
Erstellen Sie eine detaillierte Skizze.

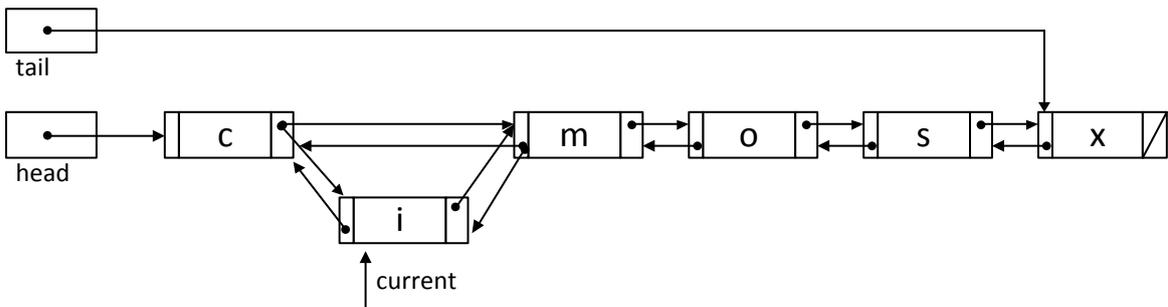
Position des zu löschenden Element suchen:



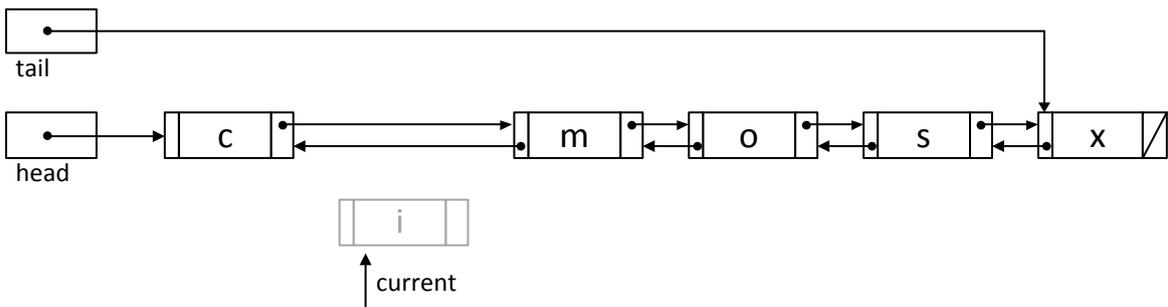
current.getPrev().setNext(current.getNext());



current.getNext().setPrev(current.getPrev());



current = null;



Kontrollfragen B

1. Nennen Sie Alltagsbeispiele für das LIFO-Verhalten.
LIFO: Last In First Out
Kleiderstapel, Papierkorb
2. Was sind die Vor- und Nachteile der Stack Implementation mit Array?
Vorteile: bei einem Array ist das Einfügen und Löschen schnell, braucht wenig Speicher
Nachteil: Grösse (Anz. Elemente) ist begrenzt / statisch
3. Nennen Sie typische Anwendungen für Stacks.
 - *Parameterübergabe an Methoden*
 - *Programmausführungen (Programmzähler bei Methode Aufruf, Interrupt)*
 - *Test auf korrekte Klammersetzung*
 - *Auswertung von Postfix-Ausdrücken / UPN-Methode bei HP Taschenrechner*

Kontrollfragen C

1. Was bedeutet FIFO?
First On First Out
2. Was bedeutet LIFO?
Last In First Out
3. Was ist der Unterschied zwischen Stack und Queue?
Stack: LIFO
Queue: FIFO
4. Welche Implementierungsvarianten kennen Sie für den Stack und die Queue?
Stack: Array / ArrayList
Queue: LinkedList, Ringpuffer