

## Aufgabe 24

Beweisen Sie folgende Aussagen durch formales Nachprüfen der Definition der Landauschen Symbole aus der Vorlesung oder widerlegen Sie die Aussage durch Angabe eines Gegenbeispiels (mit Begründung, warum dies ein Gegenbeispiel ist):

- a)  $u(n) \in O(n^2)$  und  $v(n) \in O(n) \Rightarrow u(n)+v(n)*n \in O(n^2)$
- b)  $u(n) \in \Omega(n^3) \Rightarrow v(n) = u(n)*n \in \Omega(n^4)$

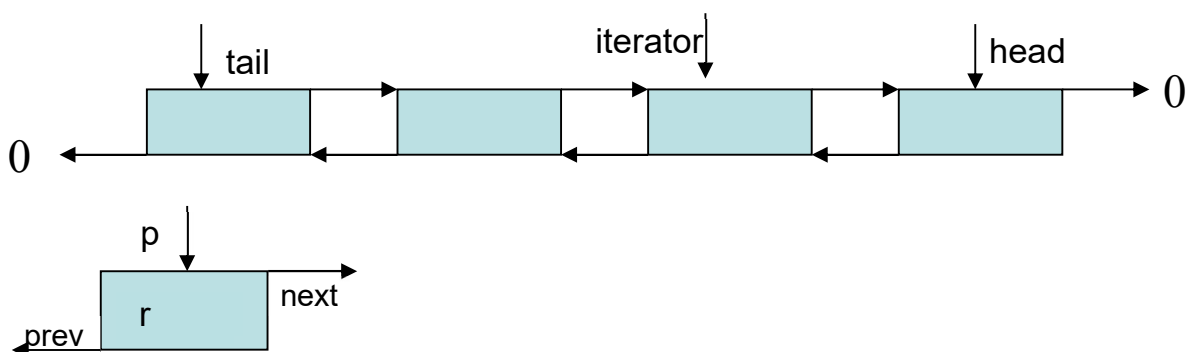
## Aufgabe 25

In der Vorlesung wurde eine Implementierung des ADT Tabelle mit Hilfe eines arrays des Datentyps <TR> vorgestellt.

Implementieren Sie folgende öffentlich sichtbaren Schnittstellenfunktionen mit Hilfe einer doppelt verketteten Liste:

```
public:
    Tabelle(); // kreiert leere Tabelle
    ~Tabelle(); // gibt von der Tabelle belegten Speicherbereich frei
    bool insert(item_type r); // fügt neuen Record vor dem aktuellen Record ein
    bool append(item_type r); // fügt neuen Record am Ende der Liste ein
    bool first(); // erster Record wird zum aktuellen Record
    bool last(); // letzter Record wird zum aktuellen Record
    bool next(); // der dem aktuellen Record folgende Record
    // wird zum neuen aktuellen Record
    bool previous(); // der dem aktuellen Record vorausgehende Record
    // wird zum neuen aktuellen Record
    bool delete_node(); // löscht aktuellen Record
    bool get_node(item_type &r); // liefert aktuellen Record in der Recordvariablen r
    bool set_node(item_type &r); // ersetzt aktuellen Record durch Recordvariable r
};
```

Tragen Sie die Resultate der Zeigeroperationen in die unten stehende Skizze ein. Achten Sie dabei auf Sonderfälle (leere Tabelle, iterator=0, iterator=head/tail,...)



## **Aufgabe 26**

Welche Aufwände entstehen bei den Methoden delete, insert in Aufgabe 22. Von welcher Komplexitätsklasse sind diese Aufwände?  
Vergleichen Sie die Aufwände im Falle einer Implementierung mit Hilfe eines arrays und mit Hilfe der doppelt verketteten Liste.