

ARRAYLIST - „FLEXIBLER ARRAY“

- Nachteil Array: die Anzahl Elemente sind fix
- Lösung: Datenstruktur `ArrayList<Typ>`
- Klasse von Java, die es ermöglicht eine Sammlung von Elementen flexibler zu verwalten

Datentyp der Elemente
der Sammlung

Variablenname

```
// Deklaration  
ArrayList<Pacman> pacList;  
// Erstellung  
pacList = new ArrayList();
```

```
// Elemente hinzufügen  
Pacman one = new Pacman(0,0,1);  
pacList.add(one);  
pacList.add(new Pacman(2,3,2));  
// Zugriff auf Elemente  
Pacman another = pacList.get(1);
```

```
// Erstellung  
pacList = new ArrayList();
```

neues ArrayList Objekt, mit
Funktionalität eine Sammlung
vom Typ Pacman zu speichern
aber noch leer.

pacList



```
// Elemente hinzufügen  
Pacman one = new Pacman(0,0,1);  
pacList.add(one);  
pacList.add(new Pacman(2,3,2));  
// Zugriff auf Elemente  
Pacman another = pacList.get(1);
```

pacList

Pacman Objekt one

x: 0
y: 0
size: 1

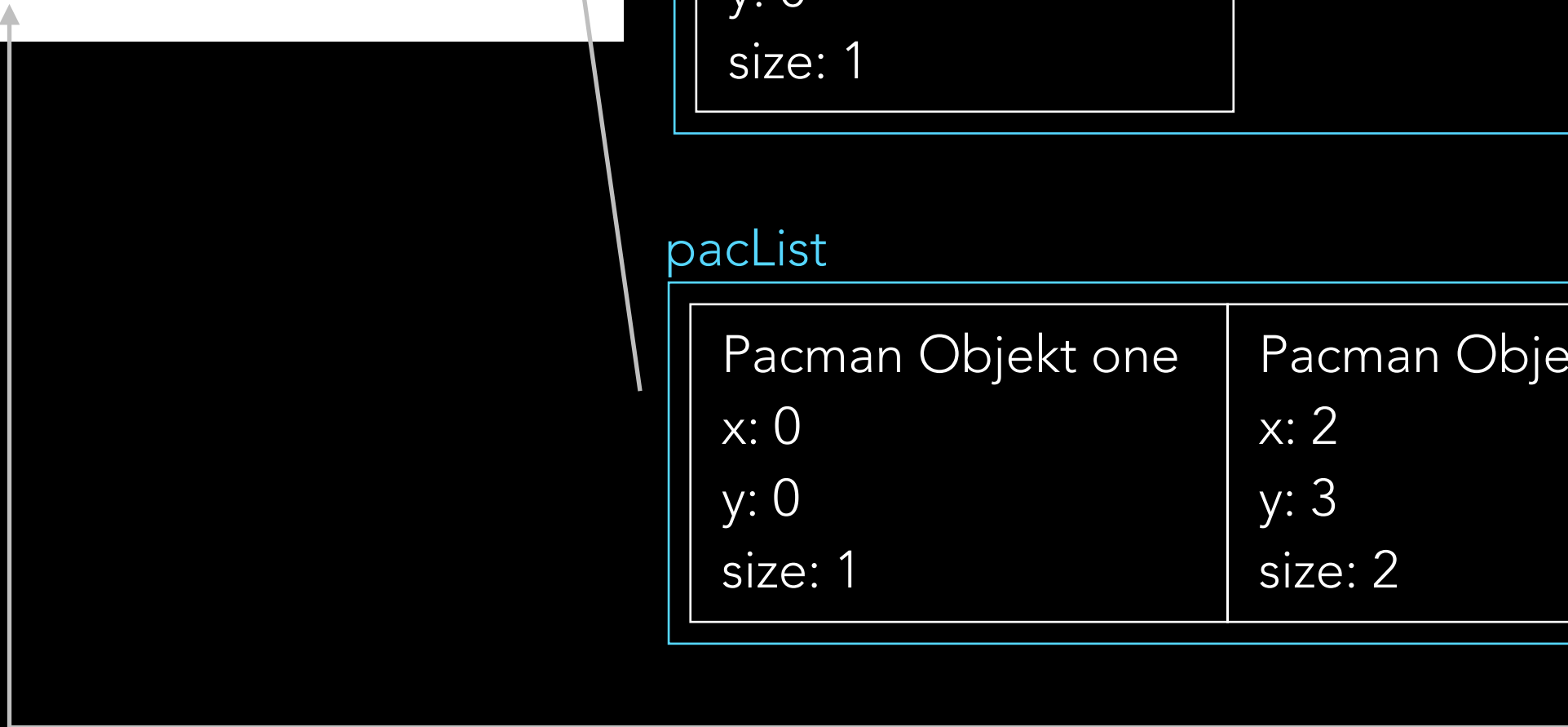
pacList

Pacman Objekt one

x: 0
y: 0
size: 1

Pacman Objekt

x: 2
y: 3
size: 2



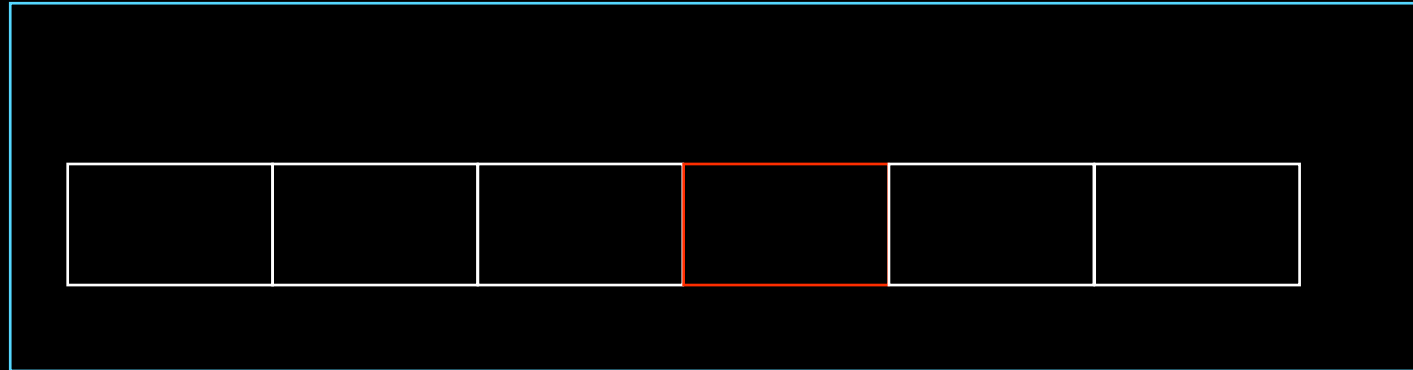
ArrayList - Element entfernen

```
for( int i = pacList.size()-1; i >= 0; i-- ) {  
    if(somethingTrue){  
        pacList.remove(i);  
    }  
}
```

- remove(int index) Methode
- Da die Grösse einer ArrayList veränderbar ist, müssen wir rückwärts durch die Liste iterieren.

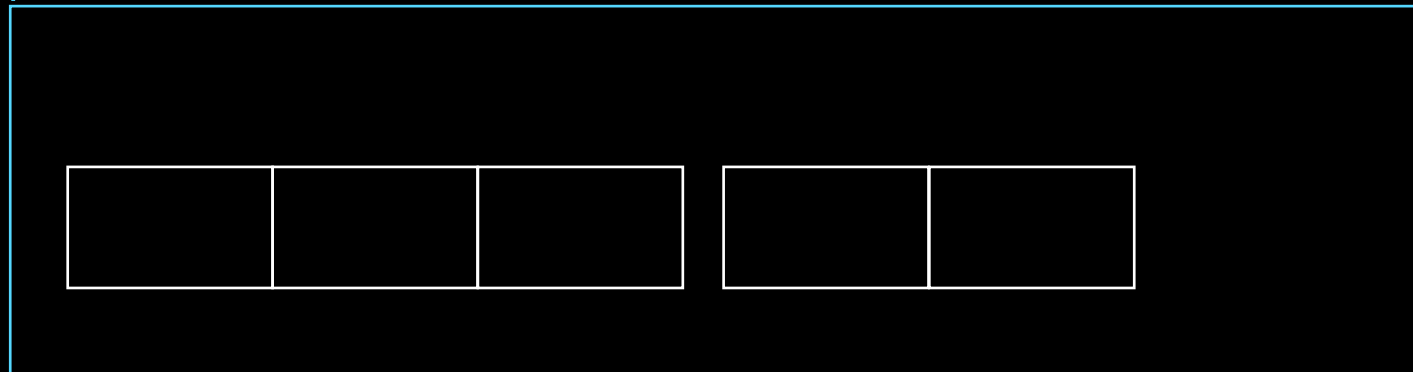
Problem wenn wir vorwärts iterieren würden:

pacList



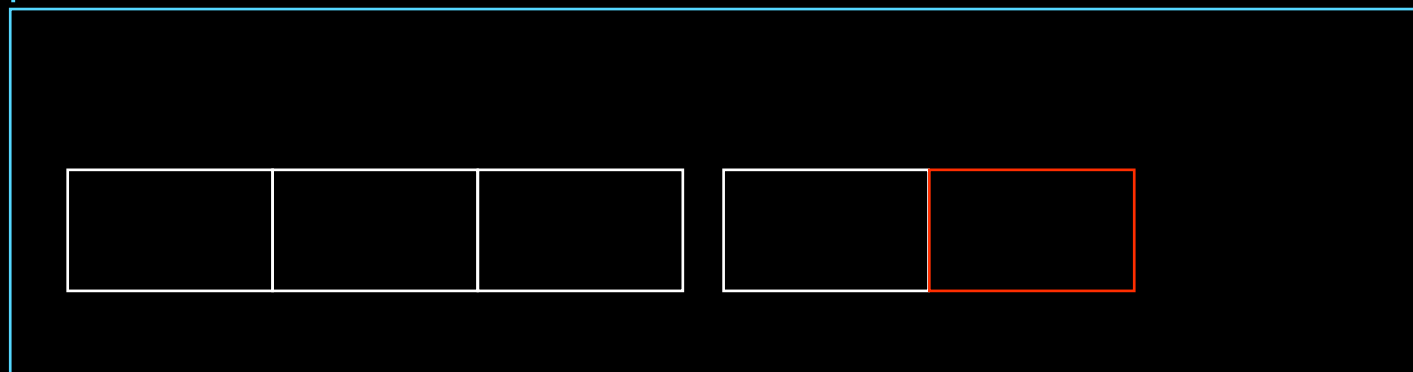
$i = 3$
 $size = 6$

pacList



$remove(i)$

pacList

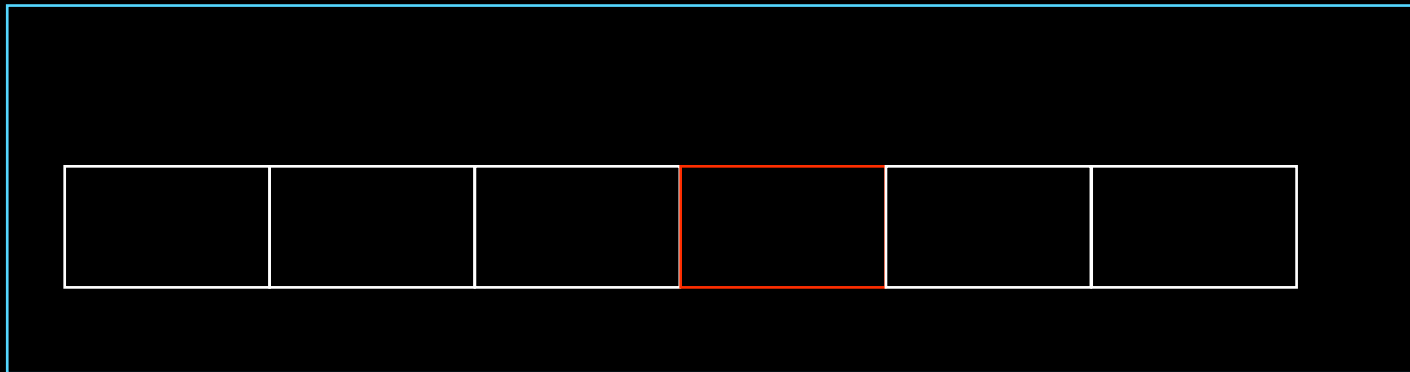


$i++$ ist 4

noch schlimmer: im nächsten Schritt wird i zu 5 $size$ ist aber auch 5

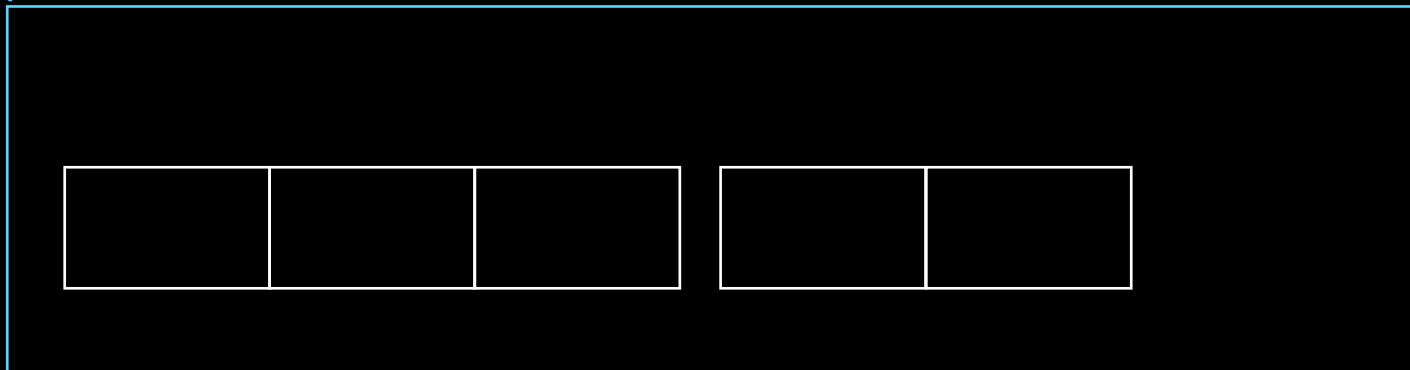
rückwärts:

pacList



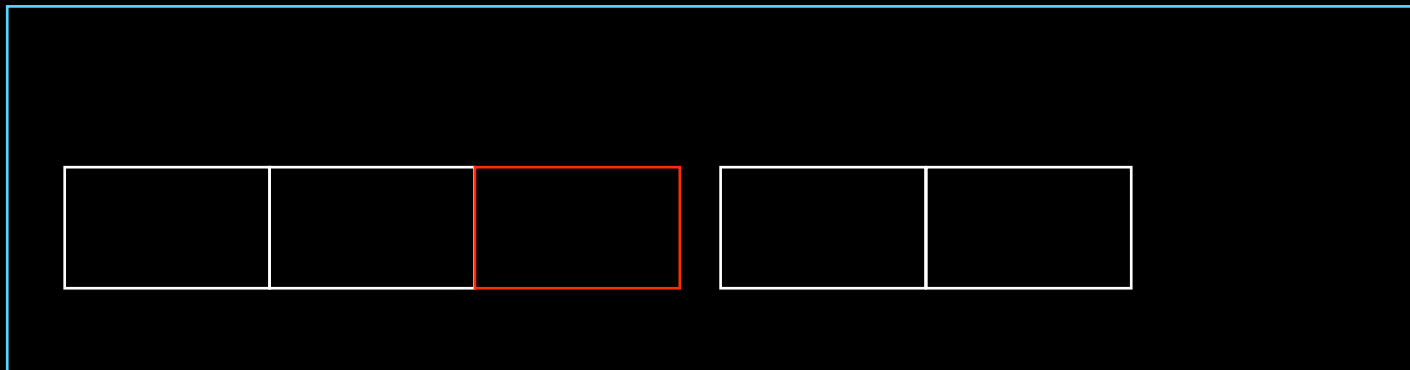
$i = 3$
size = 6

pacList



remove(i)

pacList



$i - 1$ ist 2