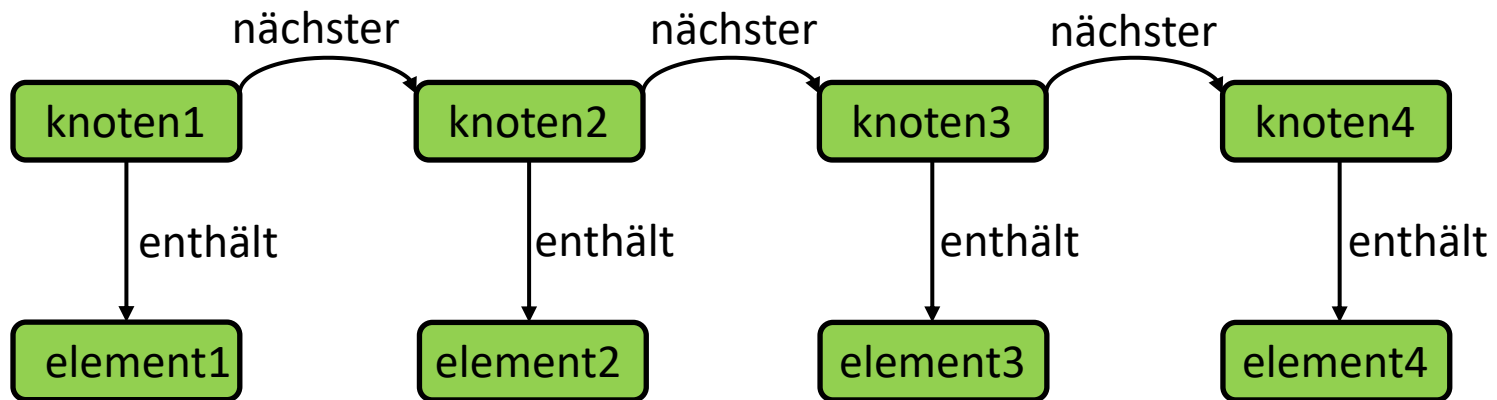


2 Einfach verkettete Listen

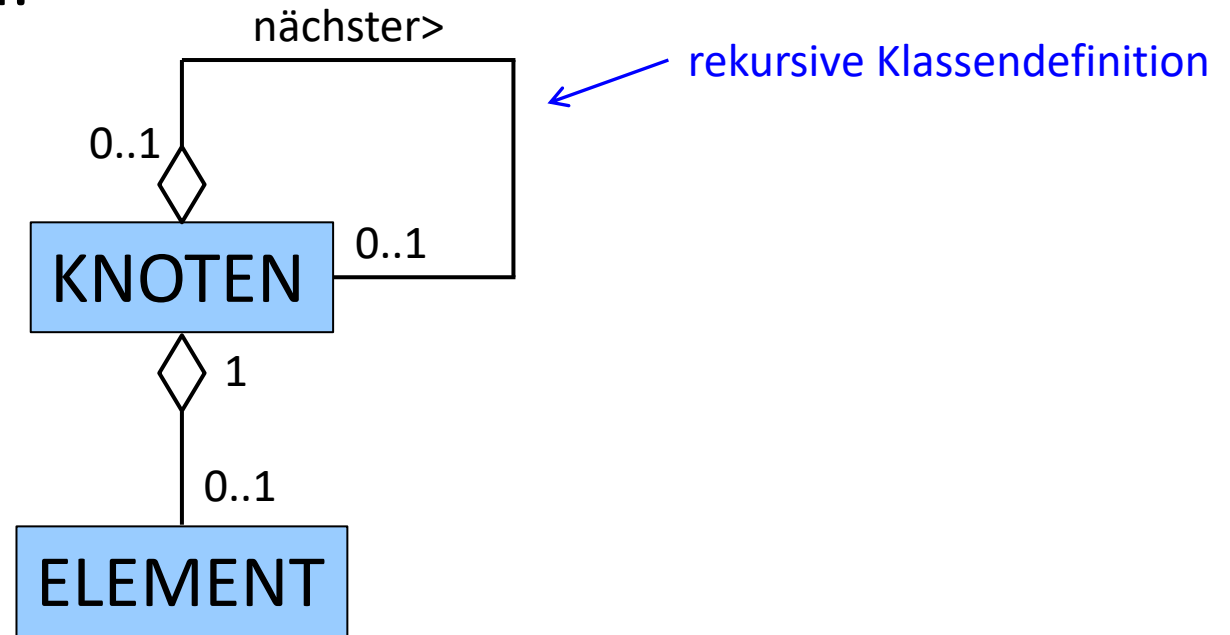
Eine Liste ist eine dynamische Datenstruktur, d.h. ihre Länge kann sich während der Laufzeit des Programms ändern.

Listenelemente heißen Knoten.

Objektdiagramm:



Klassendiagramm:



Die Klasse KNOTEN

KNOTEN
inhalt naechster
naechsterGeben() naechsterSetzen(naechsterKnoten) inhaltGeben()

inhalt ist von der Klasse Element, naechster ist von der Klasse Knoten.

Element ist die Klasse, deren Objekte in der Liste verwaltet werden sollen (Patient, Auto, ...).

Implementiere die Klassen Knoten und Element (soll nur einen Bezeichner vom Typ String verwalten.)

```
public class Knoten {
```

```
    private Element inhalt;
```

```
    private Knoten naechster;
```

```
    public Knoten(Element inh) {
```

```
        inhalt = inh;
```

```
        naechster =
```

```
    }
```

```
    public naechsterGeben
```

```
    }
```

```
    public naechsterSetzen
```

```
    }
```

```
}
```

```
public class Knoten {  
    private Element inhalt;  
    private Knoten naechster;  
  
    public Knoten(Element inh) {  
        inhalt = inh;  
        naechster = null;  
    }  
  
    public Knoten naechsterGeben() {  
        return naechster;  
    }  
  
    public void naechsterSetzen(Knoten k) {  
        naechster = k;  
    }  
}
```

```
public class Element {  
    private String bezeichner;  
    public Element(String bez){  
        bezeichner = bez;  
    }  
    public String datenGeben(){  
        return bezeichner;  
    }  
}
```

Teste das Programm, indem du "per Hand" einige Objekte der Klasse Element erzeugst, sie in Knoten füllst und miteinander verkettest.

Versuche die Reihenfolge der Objekte deines Nachbarn nachzuvollziehen! (Plätze tauschen!)