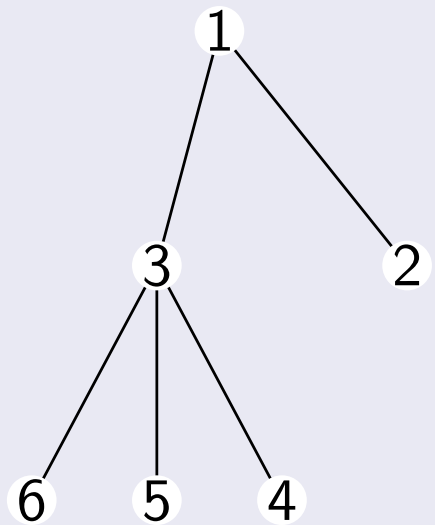
Beispiel

```
Tree t1 = new Tree(1);
Tree t2 = new Tree(2);
Tree t3 = new Tree(3);
Tree t4 = new Tree(4);
Tree t5 = new Tree(5);
Tree t6 = new Tree(6);
t1.addChild(t2);
t1.addChild(t3);
t3.addChild(t4);
t3.addChild(t5);
t3.addChild(t6);
```

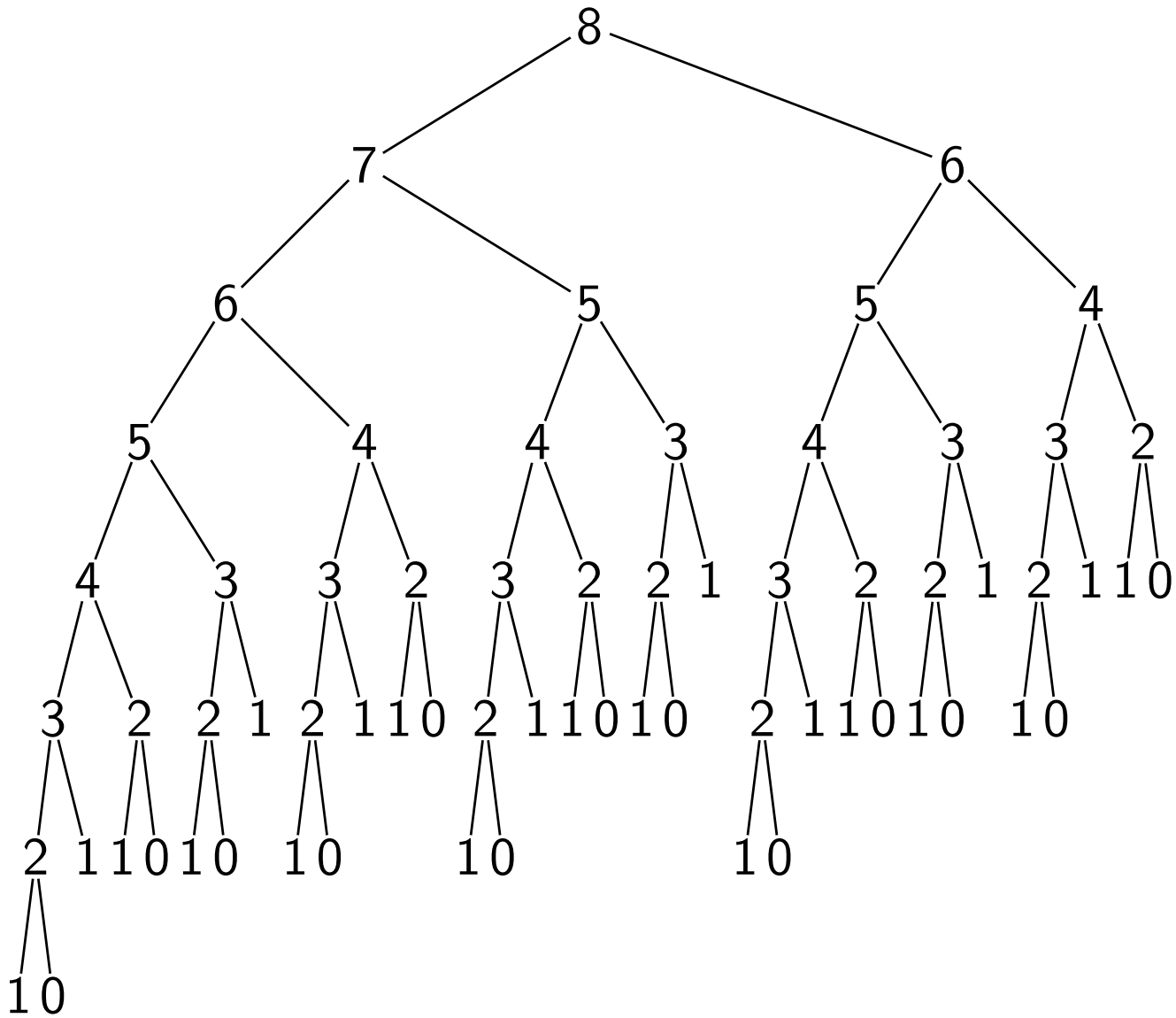
Beispiel

```
Tree t1 = new Tree(1);
Tree t2 = new Tree(2);
Tree t3 = new Tree(3);
Tree t4 = new Tree(4);
Tree t5 = new Tree(5);
Tree t6 = new Tree(6);
t1.addChild(t2);
t1.addChild(t3);
t3.addChild(t4);
t3.addChild(t5);
t3.addChild(t6);
```



Beispiel Fibonacci-Bäume

```
static Tree fiboTree(int n) {
  if(n < 2) {
    return new Tree(n);
  }
  else {
    Tree t = new Tree(n);
    t.addChild(fiboTree(n − 2));
    t.addChild(fiboTree(n − 1));
    return t;
  }
}
```


8
7 6
6 5 5 4
5 4 4 3 4 3 3 2
4 3 3 2 3 2 2 1 3 2 2 1 2 1 1 0
3 2 2 1 2 1 1 0 2 1 1 0 1 0 2 1 1 0 1 0 1 0
2 1 1 0 1 0 1 0 1 0 1 0
1 0

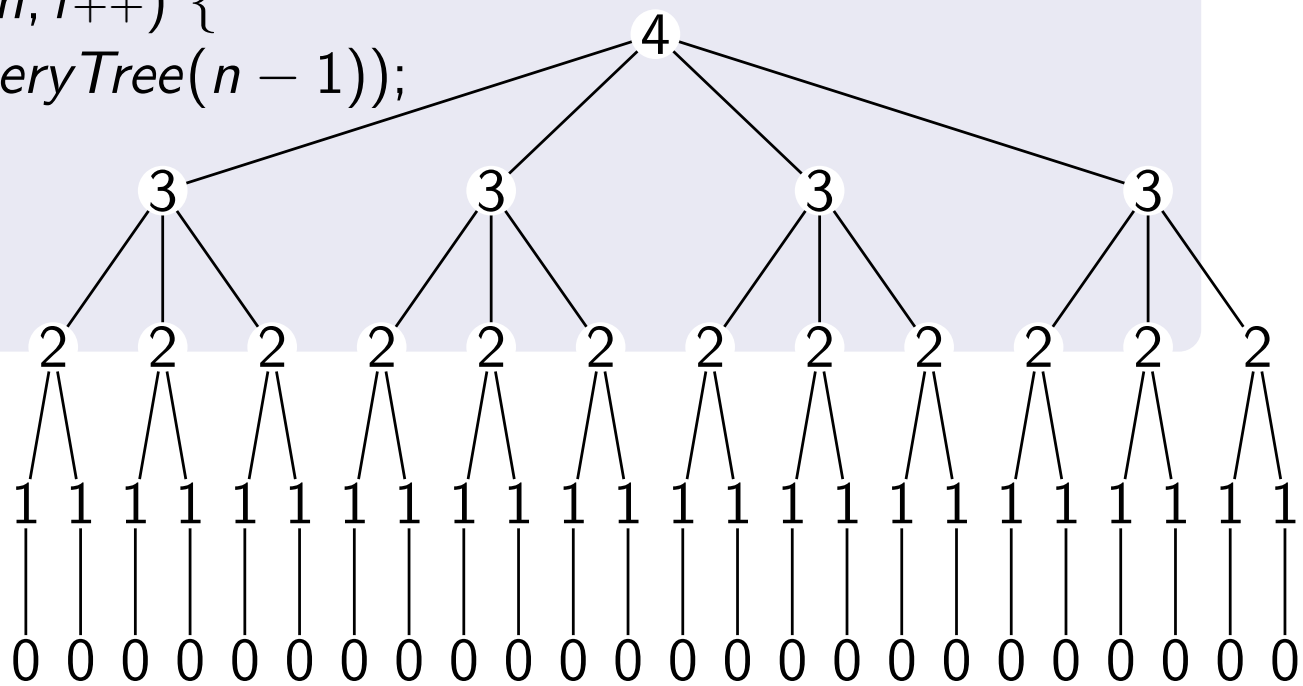
Noch ein Beispiel

```
static Tree misteryTree(int n) {
    Tree t = new Tree(n);
    for(int i = 0; i < n; i++) {
        t.addChild(misteryTree(n − 1));
    }
    return t;
}
```

Welchen Baum erzeugt *misteryTree*(4)?

Noch ein Beispiel

```
static Tree misteryTree(int n) {
    Tree t = new Tree(n);
    for(int i = 0; i < n; i++) {
        t.addChild(misteryTree(n − 1));
    }
    return t;
}
```



Wir berechnen die Summe aller Knoten eines Baumes.

```
public int totalSumRecursive() {
  int sum = getRoot();
  List<Tree> children = getChildren();
  while(!children.isempty()) {
    sum = sum + children.head().totalSumRecursive();
    children = children.tail();
  }
  return sum;
}
```

Diese Funktion ist halbwegs übersichtlich und einfach.

Später werden wir dies noch verbessern!

Jetzt eine (schlechtere) Lösung ohne Rekursion.

```
public int totalSumIterative() {
  int sum = 0;
  List<Tree> treesToSum = new List<Tree>();
  treesToSum = treesToSum.add(this);
  while(!treesToSum.isempty()) {
    Tree tree = treesToSum.head();
    treesToSum = treesToSum.tail();
    sum = sum + tree.getRoot();
    List<Tree> children = tree.getChildren();
    while(!children.isempty()) {
      treesToSum = treesToSum.add(children.head());
      children = children.tail();
    }
  }
  return sum;
}
```

Überblick

2. Rekursion
   - Rekursive Methoden
   - Backtracking
   - Memorization
   - Einfache rekursive Datenstrukturen
   - Bäume
   - Aufzählen, Untermengen, Permutationen, Bitmengen

Aufzählen

Oft müssen wir verschiedene Dinge *aufzählen*.

Zum Beispiel die Untermengen einer Menge:

Sei $M = \{3, 7, 25\}$.

Die Untermengen von M sind

$$\emptyset, \{3\}, \{7\}, \{25\}, \{3, 7\}, \{3, 25\}, \{7, 25\}, \{3, 7, 25\}.$$