

Lösungsvorschlag Aufgabe 1 (15 Punkte):

a) Ausgabe:

i	a
1	[7, 5, 2, 4, 1, 3]
2	[7, 5, 2, 4, 1, 3]
3	[7, 5, 4, 2, 1, 3]
4	[7, 5, 4, 2, 1, 3]
5	[7, 5, 4, 3, 2, 1]

b) Man muss lediglich $a[j - 1] < t$ durch $a[j - 1] > t$ ersetzen.

Lösungsvorschlag Aufgabe 2 (15 Punkte):

```
public static double preisDornloseRosen(Blume[] blumen) {  
    double res = 0.0;  
    for (int i = 0; i < blumen.length; i++) {  
        if (blumen[i] instanceof Rose &&!((Rose)blumen[i]).hatDornen()) {  
            res += blumen[i].getPreis();  
        }  
    }  
    return res;  
}
```

Lösungsvorschlag Aufgabe 3 (8 + 4 + 18 = 30 Punkte):

a) **Effiziente Lösung**

```
Element contains(int v) {  
    if (v == this.val) return true;  
    else if (v < this.val) {  
        if (this.left == null) return false;  
        else return this.left.contains(v);  
    } else {  
        if (this.right == null) return false;  
        else return this.right.contains(v);  
    }  
}
```

Brute-Force Lösung

```
Element contains(int v) {  
    boolean res = v == this.val;  
    res |= (this.left != null && this.left.contains(v));  
    res |= (this.right != null && this.right.contains(v));  
    return res;  
}
```

b) **public** *Integer* next() **throws** *NoSuchElementException* {
 if (this.n == null) **throw new** *NoSuchElementException*();
 int res = this.n.val;
 forward()
 return res;
}

c) **void** forward() {
 int oldVal = this.n.val;
 if (this.n.right != null) this.n = this.n.right.min();
 else {
 while (this.n != null && this.n.val <= oldVal) {
 this.n = this.n.parent;
 }
 }
}

Lösungsvorschlag Aufgabe 4 (15 + 5 = 20 Punkte):

a)

$$\langle n \geq 0 \rangle$$

$$\langle n \geq 0 \wedge 0 = 0 \wedge 0 = 0 \rangle$$

`i = 0;`

$$\langle n \geq 0 \wedge i = 0 \wedge 0 = 0 \rangle$$

`res = 0;`

$$\langle n \geq 0 \wedge i = 0 \wedge res = 0 \rangle$$

$$\langle res = \frac{i*(i+1)*(2*i+1)}{6} \wedge i \leq n \rangle$$

while (`i < n`) {

$$\langle res = \frac{i*(i+1)*(2*i+1)}{6} \wedge i \leq n \wedge i < n \rangle$$

$$\langle res + (i + 1) * (i + 1) = \frac{(i+1)*(i+1+1)*(2*(i+1)+1)}{6} \wedge i + 1 \leq n \rangle$$

`i = i + 1;`

$$\langle res + i * i = \frac{i*(i+1)*(2*i+1)}{6} \wedge i \leq n \rangle$$

`res = res + i * i;`

$$\langle res = \frac{i*(i+1)*(2*i+1)}{6} \wedge i \leq n \rangle$$

}

$$\langle res = \frac{i*(i+1)*(2*i+1)}{6} \wedge i \leq n \wedge i \not< n \rangle$$

$$\langle res = \frac{n*(n+1)*(2*n+1)}{6} \rangle$$

b) Eine gültige Variante für die Terminierung ist $V = n - i$, denn die Schleifenbedingung $B = i < n$ impliziert $n - i \geq 0$ und es gilt:

$$\langle n - i = m \wedge i < n \rangle$$

$$\langle n - (i + 1) < m \rangle$$

`i = i + 1;`

$$\langle n - i < m \rangle$$

`res = res + i * i;`

$$\langle n - i < m \rangle$$

Damit ist die Terminierung der einzigen Schleife in P gezeigt.

Lösungsvorschlag Aufgabe 5 (20 Punkte):

a) $v == val$

b) $v < val \ \&\& \ left == null$

c) $prefixsum \ xs = prefixsum' \ xs \ 0$

$prefixsum' \ [] \ res = [res]$

$prefixsum' \ (x:xs) \ res = (x+res) : (prefixsum' \ xs \ (x + res))$

Lösungsvorschlag Aufgabe 6 (4 + 6 + 10 = 20 Punkte):

a) i) $f(X, g(Y, Y), a), f(g(Z, Z), Z, Y)$: mgu $X/g(a, a), Y/a, Z/g(a, a)$

ii) $f(g(X, Y), Z, X), f(Z, g(a, b), Y)$: clash failure a/b

iii) $f(X, Y, Z), f(Y, Z, g(X, Y))$: occur failure X in $g(X, X)$

iv) $f(g(X, Y), X, Z), f(g(Z, Z), Y, a)$: mgu $X/a, Y/a, Z/a$

b) A / [1, 2]

A / [2, 1]

c) `sort([], []).`

`sort([X|XS], YS) :- sort(XS, ZS), add(X, ZS, YS).`