

## Musterlösungen für ausgewählte Aufgaben der Übungen zu den Grundstrukturen

- Musterlösungen nachvollziehen
- Besonderes Augenmerk auf die Syntax legen (Klammersetzung...)
  - hierzu auch die Syntax unter Punkt 1 – Wiederholung Grundstrukturen im PWiki beachten
- Struktogramme als Übung erstellen
- Variablenbelegungstabellen als Übung erstellen
- Kommentarfunktion nutzen

```

x| 1 /*
  2
  3 gesamter Bereich wird als
  4
  5 Kommentar gefasst
  6
  7
  8 */
  9
 10
 11
 12 // zeilenweise
 13 // kommentieren
 14
    
```

### Exponentialfunktion a hoch x

```

x| 1 function ahochx(a,x) {
  2   var n = 0;
  3   var y = 1;
  4   do {
  5     n = n+1;
  6     y = y*a;
  7     console.log(y);
  8   } while (n < x);
  9 }
 10
x| 1 ahochx(3,4);
  2
    
```

**speichern & ausführen**

a	x	n	y	
3	4			
übergeben	übergeben	0 gesetzt (Zeile 2)	1 gesetzt (Zeile 3)	
		1	3	aus Zeile 5 und 6 Bedingung: n kleiner als x erfüllt
		2	9	...
		3	27	
		4	81	n=x → Abbruch

### Summe der ersten n Zahlen

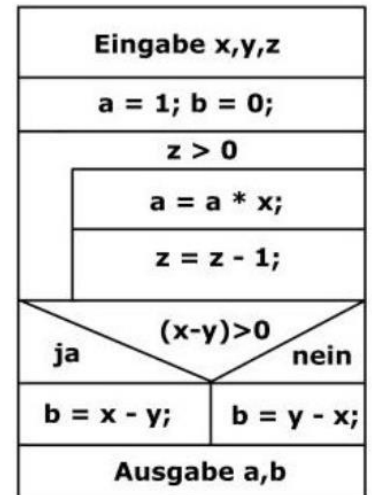
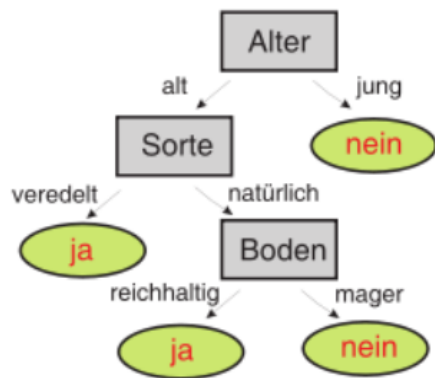
$$S = \sum_{i=0}^n i = 1 + 2 + 3 + \dots + n$$

```

x| 1 function summe_erste_n(n) {
  2   var summe = 0;
  3   for(var i=1; i<n+1; i++){
  4     summe = summe + i;
  5     //console.log(summe); andere Ausgabemöglichkeit
  6   }
  7   return summe;
  8 }
  9
x| 1 summe_erste_n(8);
  2
    
```

**speichern & ausführen**

n	i	summe	
8		0 gesetzt (Zeile 2)	
übergeben	1 (Start aus Zeile 3)		
		1	aus Zeile 4
	2	3	
	3	6	
	4	10	
	5	15	
	6	21	
	7	28	
	8	36	i=n → Abbruch



```

x| function traegt(alter, sorte, boden){
2
3   if(alter=="jung"){
4     console.log("nein");
5   }
6
7   if(alter=="alt"){
8
9     if(sorte=="veredelt"){
10      console.log("ja");
11    }
12
13    if(sorte=="natürlich"){
14
15      if(boden=="mager"){
16        console.log("nein");
17      }
18
19      if(boden=="reichhaltig"){
20        console.log("ja");
21      }
22    }
23  }
24 }
25 }
26

```

```

x| traegt("jung", "natürlich", "mager");

```

```

x| function Potenz_Betrag(x, y, z) {
2   var a = 1, b = 0;
3   // Potenz
4   while(z > 0) {
5     a = a * x;
6     z = z - 1; // oder einfach: z--;
7   }
8   // Betrag
9   if((x-y) > 0)
10    b = x - y;
11  else
12    b = y - x;
13  console.log("a = " + a);
14  console.log("b = " + b);
15 }
16

```

```

x| Potenz_Betrag(2, 50, 6);
2

```

**speichern & ausführen**

```

> a = 64
  b = 48

```

## Berechnung der Nullstellen einer quadratischen Funktion

```
x| function Nullstelle(p,q) {  
  2 var D = p*p/4-q;  
  3 if (D < 0){  
  4 console.log("keine Nullstelle")  
  5 }  
  6 else if (D == 0) {  
  7 var x1 = -p/2;  
  8 console.log("eine Nullstelle")  
  9 console.log(x1);;  
 10 }  
 11 else {  
 12 var x1 = -p/2 + Math.sqrt(D);  
 13 var x2 = -p/2 - Math.sqrt(D);  
 14 console.log("zwei Nullstellen");  
 15 console.log(x1);  
 16 console.log(x2);  
 17 }  
 18 }  
 19
```

```
x| Nullstelle(0,-50);  
  2
```

**speichern & ausführen**

```
> zwei Nullstellen  
  7.0710678118654755  
 -7.0710678118654755
```