

Sofern nichts anderes steht, sind die Ausgaben der Programmfragmente anzugeben.

Aufgabe 3.1

```
1 j = 3
2 while j < 15:
3     print(j)
4     j = j + 2
```

Aufgabe 3.2

```
1 j = 3
2 while j < 15:
3     j = j + 3
4     print(j)
```

Aufgabe 3.3

```
1 i = 14
2 while i > 1:
3     print(i)
4     i = i - 4
```

Aufgabe 3.4

```
1 c = 2
2 while c < 8:
3     print(c)
4     c = c + 1
```

Aufgabe 3.5

```
1 s = 2
2 while s < 18:
3     s = s + 2
4 print(s)
```

Aufgabe 3.6

```
1 # Läuft das Programm endlos?
2 k = 0
3 while k < 10:
4     k = k + 1
```

Aufgabe 3.7

```
1 # Läuft das Programm endlos?  
2 k = 0  
3 while k > 10:  
4     k = k - 1
```

Aufgabe 3.8

```
1 # Läuft das Programm endlos?  
2 k = 0  
3 while k != 5:  
4     k = k + 2
```

Aufgabe 3.9

```
1 print('BrYqGm'.lower())
```

Aufgabe 3.10

```
1 print('hgPQeN'.upper())
```

Aufgabe 3.11

```
1 print('AtnYbH'.capitalize())
```

Aufgabe 3.12

```
1 print(not False or True)
```

Aufgabe 3.13

```
1 print(False or True and False and True)
```

Aufgabe 3.14

```
1 print(False or not True and True)
```

Aufgabe 3.15

```
1 print(19 in [9, 17, 19, 3, 10, 13, 8, 16])
```

Aufgabe 3.16

```
1 print(10 not in [17, 18, 9, 14, 2, 19, 12, 6])
```

Aufgabe 3.17

```
1 print(690 * 348 == 348 * 690)
```

Aufgabe 3.18

```
1 print(398 % 10 != 289 % 10)
```

Aufgabe 3.19

```
1 print(942 + 292 < 292 + 942)
```

Aufgabe 3.20

```
1 print(134 < 728 < 773)
```

Aufgabe 3.21

```
1 print('python' == 'python ')
```

Aufgabe 3.22

```
1 print('Unterricht' < 'Unterwelt')
```

Aufgabe 3.23

```
1 print('Zimmer' < 'aufräumen')
```

Aufgabe 3.24

Zeige anhand eines Beispiels im Zehnersystem, dass der Vergleich von zwei Gleitkommazahlen zu einem falschen Ergebnis führen kann, wenn beispielsweise nur 4 Stellen der Zahlen gespeichert werden können.

Aufgabe 3.25

```
1 b = 2
2 if b > 8:
3     b = b + 1
4 b = b + 2
5 print(b)
```

Aufgabe 3.26

```
1 q = 3
2 if q <= 7:
3     q = q + 1
4 else:
5     q = q + 2
6 q = q + 4
7 print(q)
```

Aufgabe 3.27

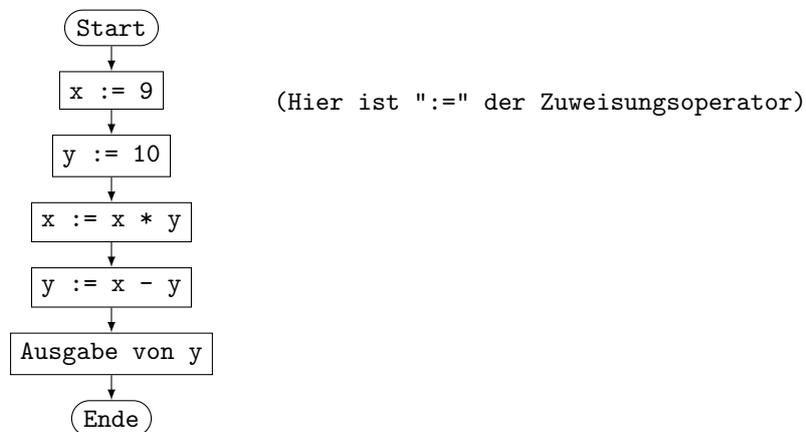
```
1 p = 7
2 if p <= 5:
3     if p <= 1:
4         p = p + 1
5     else:
6         p = p + 2
7 else:
8     if p <= 8:
9         p = p + 3
10    else:
11        p = p + 4
12 p = p + 10
13 print(p)
```

Aufgabe 3.28

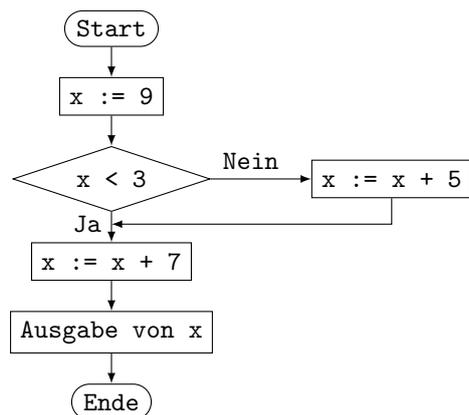
```
1 c = 4
2 if c < 1:
3     c = c + 1
4 elif c <= 5:
5     c = c + 2
6 elif c < 8:
7     c = c + 3
8 else:
9     c = c + 4
10 c = c + 10
11 print(c)
```

Die folgenden Flussdiagramme beschreiben jeweils ein Programm. Welche Ausgabe(n) macht es? Beachte, dass hier „Pseudocode“ verwendet wird und Zuweisungen mit dem Operator „:=“ dargestellt werden. Das erlaubt es, den Vergleichsoperator mit „>“ statt wie in Python mit „>“ zu schreiben.

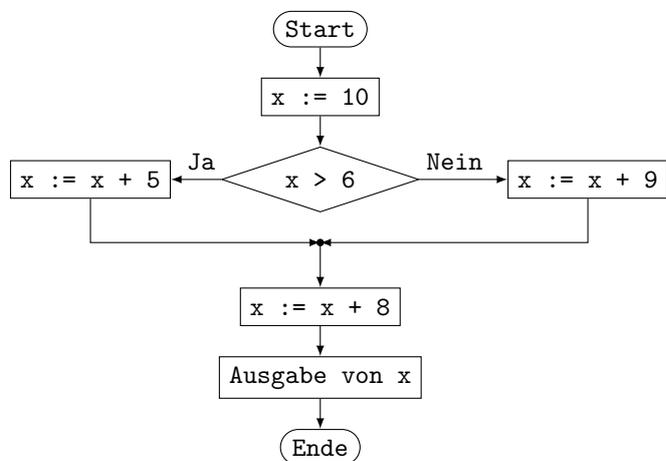
Aufgabe 3.29



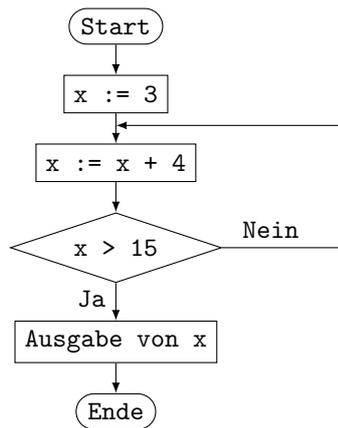
Aufgabe 3.30



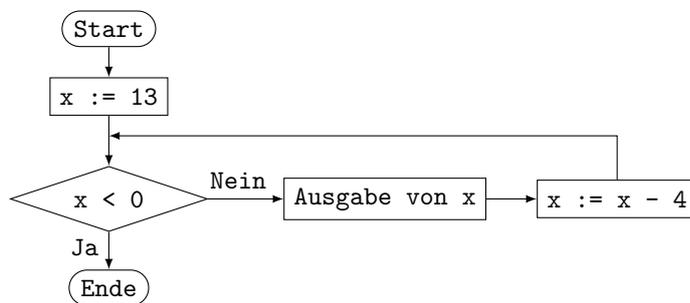
Aufgabe 3.31



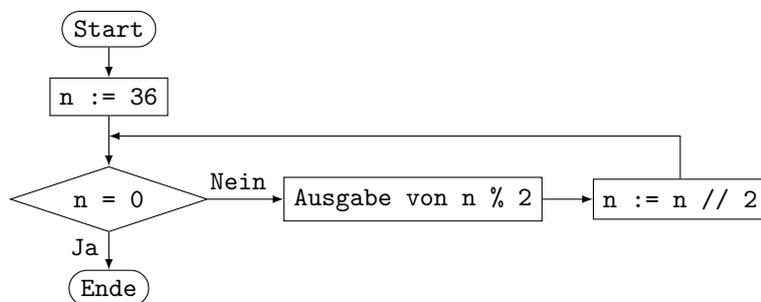
Aufgabe 3.32



Aufgabe 3.33



Aufgabe 3.34



Markiere und beschreibe die Syntaxfehler in den Programmfragmenten.

Aufgabe 3.35

```
1 x = 5
2 if x <= 5:
3     x = x + 1
4 else:
5     x = x + 2
6 print(x)
```

Aufgabe 3.36

```
1 liste = [1, 2, 3, 4, 5, 6, 7, 8 9, 10, 11, 12]
2 x = 13
3 if x in liste:
4     print('Element kommt nicht in der Liste vor.')
```

Aufgabe 3.37

```
1 x = 52
2 if x <= 50
3     print('ok')
4 else:
5     print('not okay')
```

Aufgabe 3.38

```
1 note = 4.5
2 if note == 6:
3     print('sehr gut')
4 else if note == 5:
5     print('gut')
6 elif note == 4:
7     print('genügend')
8 elif note < 4
9     print('ungenügend')
10 else:
11     print('keine gültige Note')
```

Aufgabe 3.39

```
1 i = x
2 while x < 4:
3     x = x + 1
4     print(x)
```

Aufgabe 3.40

```
1 n = 4
2 s = 0
3 i = 1
4 while i <= n
5     s = s + i
6     i = i + 1
7 print s
```

Aufgabe 3.41

```
1 import random
2
3 buchstaben = [a, b, c]
4
5 wahl_computer = random.choice(buchstaben)
6 wahl_user = None
7
8 while wahl_user not in buchstaben:
9     wahl_user = input('a, b, c ?')
10    wahl_user = wahl_user.lower()
11
12 print(wahl_computer, wahl_user)
```