

```
def add(x,y):
    while x > 0:
        y += 1
        x -= 1
    return y
```

Speicher

x = 3
y = 4

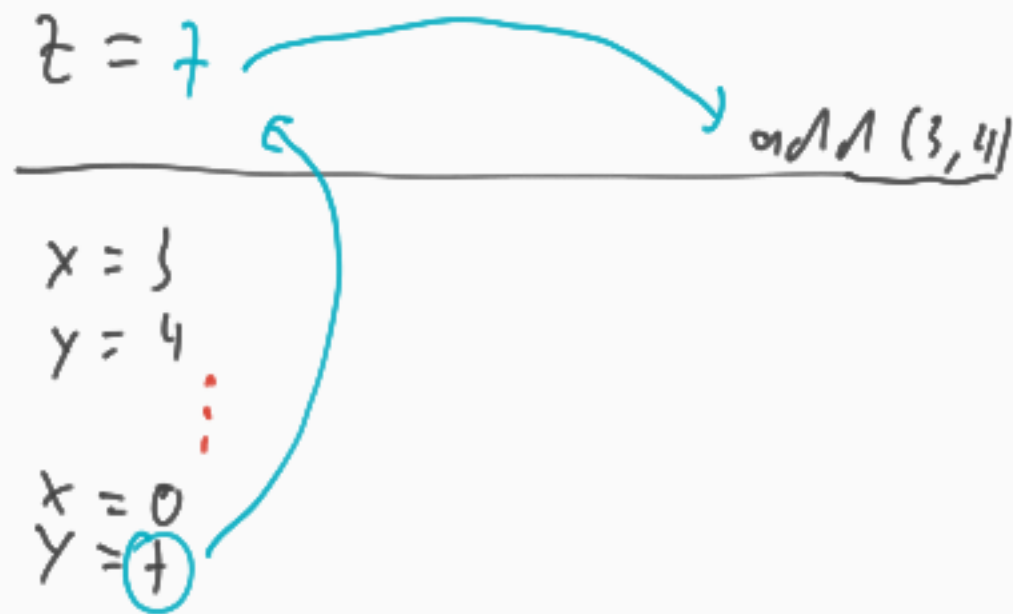
Beginn
while → nach 1.
Dunkelant → 2. → 3.
x = 3
y = 4
x = 2
y = 5
x = 1
y = 6
x = 0
y = 7

Programm

```
def add(x,y):  
    while x > 0:  
        y += 1  
        x -= 1  
  
    return y
```

1. `z = add(3,4)`
`t = add(z,2)`

Speicher



Programm

```
def add(x,y):  
    while x > 0:  
        y += 1  
        x -= 1  
  
    return y
```

1. z = add(3,4)
2. t = add(z,2)

Speicher

z = 7
t = 9

add(z,2)

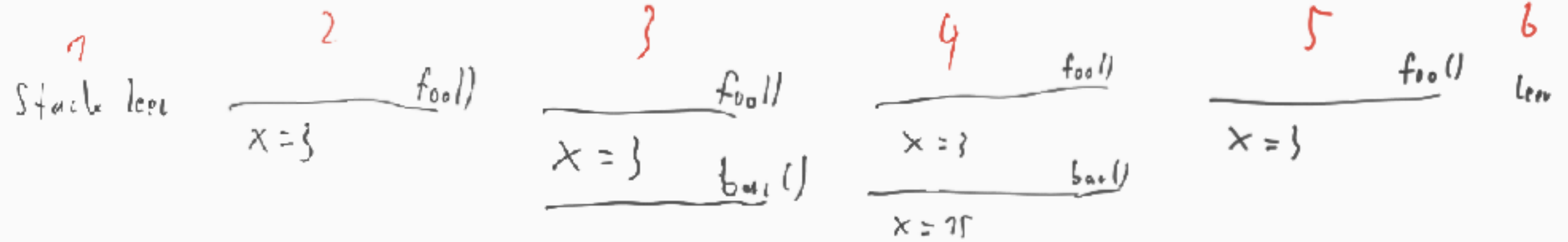
x = 7
y = 2
↓
x = 6
y = 3
↓
x = 5
y = 4
↓
⋮
↓
x = 0
y = 9

The diagram illustrates the state of memory during the execution of the 'add' function. The function is called with arguments z=7 and 2. The local variables x and y are initialized to 7 and 2 respectively. The function then enters a while loop that decrements x and increments y until x reaches 0. The final value of y is 9, which is then assigned to the global variable t. A blue arrow points from the final value of y (9) to the assignment t = 9.

Programm

```
def foo():  
    x = 3  
    print(x)  
    bar()  
    print(x)  
  
def bar():  
    x = 15  
    print(x)  
  
foo()  
  
6 →
```

Speicher



Programm

```
def factorial(n):  
    if n == 0:  
        return 1  
    e = n * factorial(n-1)  
    return e
```

factorial(3)

$$5! = 5 \cdot 4 \cdot \underbrace{3 \cdot 2 \cdot 1}_{4!}$$

$$n! = \begin{cases} 1 & \text{falls } n=0 \\ n \cdot (n-1)! & \text{sonst} \end{cases}$$

Alternative Version des Programms

```
def factorial(n):  
    return 1 if n == 0 else n * factorial(n-1)
```

Speicher $6 = \text{factorial}(3)$

$n = 3$

$e = 3 \cdot 2$

$= \text{factorial}(2)$

$n = 2$

$e = 2 \cdot 1$

$= \text{factorial}(1)$

$n = 1$

$e = 1 \cdot 1$

$= \text{factorial}(0)$

$n = 0$

$e = 1$

$$\begin{aligned} 5! &= 5 \cdot 4! \\ &= 5 \cdot \underbrace{4 \cdot 3!}_{4 \cdot 2!} \\ &= 5 \cdot 4 \cdot \underbrace{3 \cdot 2!}_{3 \cdot 1!} \\ &= 5 \cdot 4 \cdot 3 \cdot \underbrace{2 \cdot 1!}_{2 \cdot 0!} \\ &= 5 \cdot 4 \cdot 3 \cdot 2 \cdot \underbrace{1 \cdot 0!}_{1 \cdot 1} \\ &= 5 \cdot 4 \cdot 3 \cdot 2 \cdot 1 \cdot 1 \end{aligned}$$