

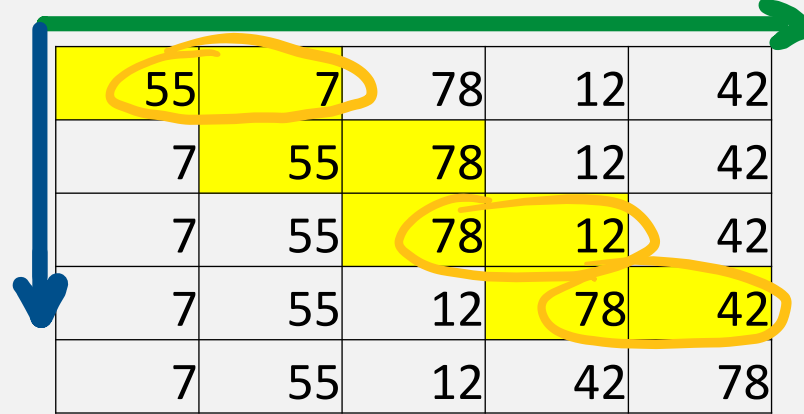
BUBBLESORT

Sortieren durch Austauschen

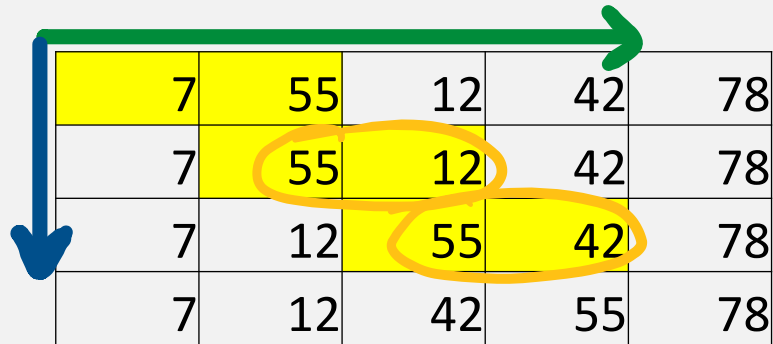
INSPIRATION



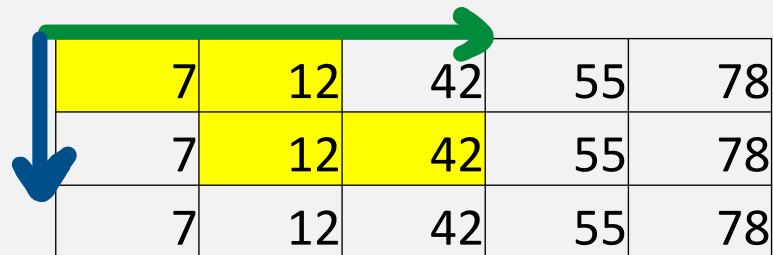
BEISPIEL ABLAUF



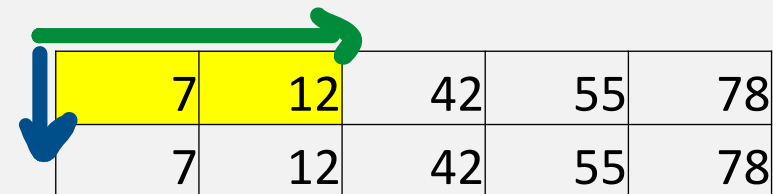
55	7	78	12	42	
7	55	78	12	42	
7	55	78	12	42	
7	55	12	78	42	
7	55	12	42	78	



7	55	12	42	78	
7	55	12	42	78	
7	12	55	42	78	
7	12	42	55	78	



7	12	42	55	78	
7	12	42	55	78	
7	12	42	55	78	

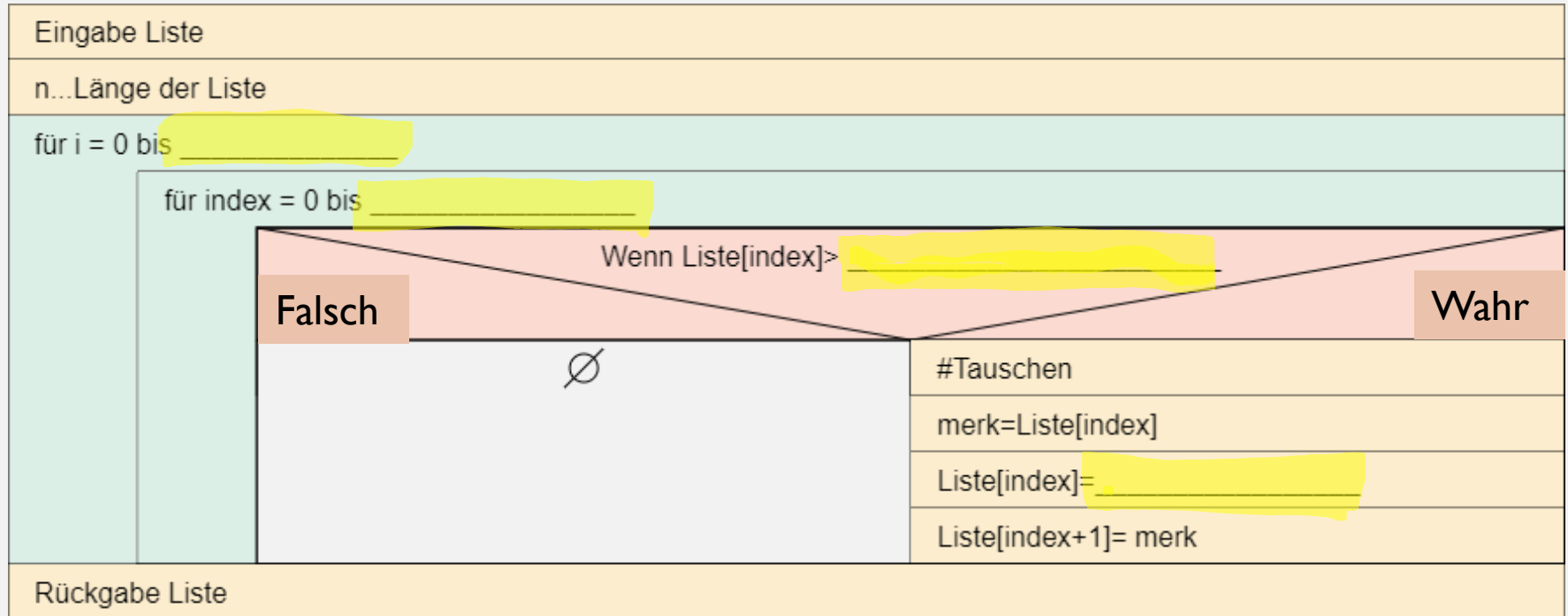


7	12	42	55	78	
7	12	42	55	78	

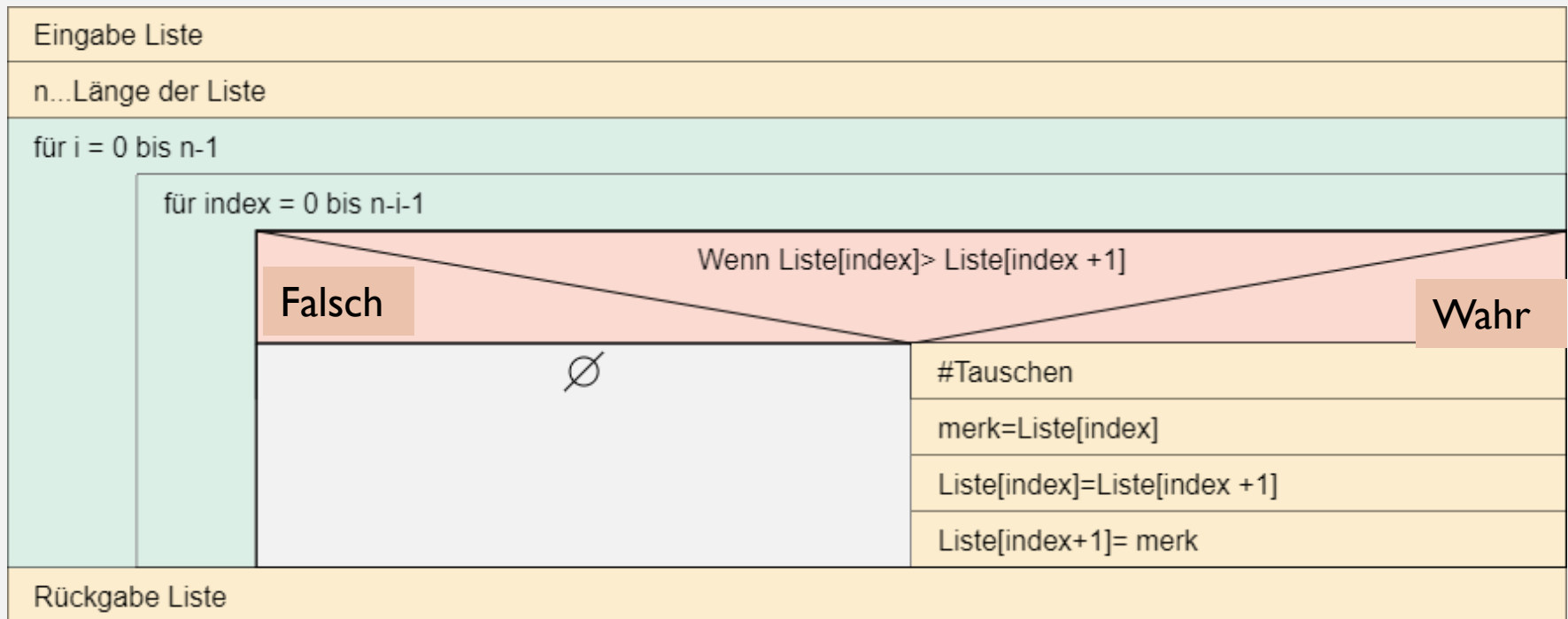
BESCHREIBUNG

In einem ersten Schritt wird das erste mit dem Zweiten Element verglichen. Ist das erste Element größer als das zweite, dann tausche die beiden. Verahre genauso mit allen restlichen Listenelementen bis zum Ende der Liste. Nach diesem Durchlauf steht am Ende des Feldes das größte Element. Verahre nun genauso mit der Liste, ohne das letzte Element. Dieses muss nicht mehr einsortiert werden, denn es steht schon an der richtigen Stelle. Nach Ende dieses Durchlaufes steht an der vorletzten Position das zweitgrößte Element. Verahre so, bis an der ersten Stelle das kleinste Element steht.

STRUKTOGRAMM



LÖSUNG



ZEIT MESSEN

- Stoppt die Zeit, die zum Sortieren benötigt wird
- Idee:

```
from time import*

start=clock()
#Liste sortieren
ende=clock()
#Zeit ausgeben
```

- Testet die Laufzeit mit verschiedenen Listenlängen
- Formuliert einen Zusammenhang

VERFAHREN - QUELLTEXT

```
def tausch(Liste, index_1, index_2):  
    tmp=Liste[index_1]  
    Liste[index_1]=Liste[index_2]  
    Liste[index_2]=tmp  
    return Liste
```

```
def bubblesort(Liste):  
    for i in range(0, len(Liste)-1):  
        for index in range(0, len(Liste)-i-1):  
            if (Liste[index]>Liste[index+1]):  
                tausch(Liste, index, index+1)  
    return Liste
```