

# Hardwarenahe Softwareentwicklung Tutorium

---

Call By Referenz and Value  
SS 16

Tobias Trabelsi

9. Juni 2016

Hochschule Bochum  
Bochum University  
of Applied Sciences



# Inhaltsverzeichnis

1	Call By Value	3
2	Call By Referenz	5
3	Übung	7
3.1	Array Befüllen . . . . .	7

# 1 Call By Value

Mit folgendem Quelltext werden in der Funktion:

*tausche\_value(int a,int b);*

zwei integer per Value übergeben. Der Gültigkeitsbereich bleibt jedoch auf die jeweiligen Funktionen beschränkt. Das bedeutet, dass die beiden Integer in der main Methode nicht verändert werden.

```
1 #include <stdio.h>
2
3 void tauschen_value(int , int);
4
5
6 int main()
7 {
8
9     int a,b;
10    a = 5;
11    b = 10;
12
13    printf("Variablen vor dem Tauschen in der main : a = %d \t b= %d \n",
14           a,b);
15
16    tauschen_value(a,b);
17
18    printf("Variablen nach dem Tauschen in der main : a = %d \t b= %d \n"
19           ,a,b);
20
21    return 0;
22 }
```

```
21
22 void tauschen_value(int a, int b){
23     printf("Variablen vor dem Tauschen in der Funktion : a = %d \t b= %d\n",a,b);
24     int tmp = a;
25     a = b;
26     b = tmp;
27     printf("Variablen nach dem Tauschen in der Funktion : a = %d \t b= %d\n",a,b);
28 }
```

## 2 Call By Referenz

Mit folgendem Quelltext werden in der Funktion:

*tausche\_referenz(int \* a, int \* b);*

zwei integer per Referenz übergeben. Also die Pointer auf den Adressbereich in dem der Wert der Variablen steht. Die Vertauschung innerhalb der Funktion hat so nun auch Auswirkungen auf die Variablen in der Main Funktion.

```
1 #include <stdio.h>
2
3 void tauschen_value(int , int);
4
5
6 int main()
7 {
8
9     int a,b;
10    a = 5;
11    b = 10;
12
13    printf("Variablen vor dem Tauschen in der main : a = %d \t b= %d \n",
14           a,b);
15
16    tauschen_value(a,b);
17
18    printf("Variablen nach dem Tauschen in der main : a = %d \t b= %d \n"
19           ,a,b);
20
21    return 0;
22 }
```

```
21
22 void tauschen_value(int a, int b){
23     printf("Variablen vor dem Tauschen in der Funktion : a = %d \t b= %d\n",a,b);
24     int tmp = a;
25     a = b;
26     b = tmp;
27     printf("Variablen nach dem Tauschen in der Funktion : a = %d \t b= %d\n",a,b);
28 }
```

### 3 Array Befüllen

Befüllen Sie ein Array mit Integern.

Lassen Sie sich danach das Array in einer anderen Funktion ausgeben. Übergeben Sie dazu einen Pointer auf das 0. Element des Arrays, so wie die Länge des Arrays. In einer kopfgesteuerten Schleife können Sie nun den Pointer hoch zählen und den entsprechenden Wert ausgeben.