



Grundlagen der Informatik und Programmierung 2

Einführung

Referenzen

Prof. Dr. Tom Vierjahn

Visual Computing (<https://vc.w-hs.de>)
Fachbereich Wirtschaft und Informationstechnik
Campus Bocholt

Sommersemester 2020



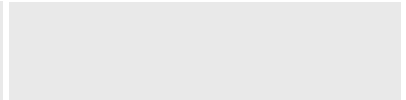
Funktionsdefinition

```
void swap_a(int a, int b) {  
    int tmp = a;  
    a = b;  
    b = tmp;  
}
```

aufrufender Code

```
int x = 17;  
int y = 42;  
  
swap_a(x, y);  
  
std::cout << "x = " << x << std::endl;  
std::cout << "y = " << y << std::endl;
```

Ausgabe



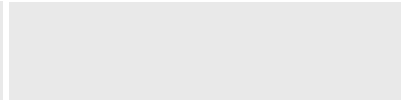
Funktionsdefinition

```
void swap_b(int* a, int* b) {  
    int tmp = *a;  
    *a = *b;  
    *b = tmp;  
}
```

aufrufender Code

```
int x = 17;  
int y = 42;  
  
swap_b(&x, &y);  
  
std::cout << "x = " << x << std::endl;  
std::cout << "y = " << y << std::endl;
```

Ausgabe



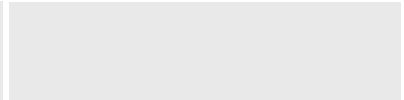
Funktionsdefinition

```
void swap_c(int& a, int& b) {  
    int tmp = a;  
    a = b;  
    b = tmp;  
}
```

aufrufender Code

```
int x = 17;  
int y = 42;  
  
swap_c(x, y);  
  
std::cout << "x = " << x << std::endl;  
std::cout << "y = " << y << std::endl;
```

Ausgabe



Definition: Referenz

Eine **Referenz** ist ein Alias-Name für ein Objekt. Eine Referenz ist selbst allerdings kein Objekt.

- ▶ Syntax für Zugriff auf referenzierte Objekte und für Zugriff auf Objekte ist identisch.
- ▶ Einmal initialisiert referenziert eine Referenz immer dasselbe Objekt.
- ▶ Eine Referenz referenziert immer ein Objekt.
- ▶ Referenzen müssen initialisiert werden.
- ▶ Operatoren wirken nicht auf die Referenz sondern auf die referenzierte Variable.
- ▶ Referenzen sind oft kleiner als das referenzierte Objekt.

veränderbare Referenz:

```
int a = 17;  
int& x = a;
```

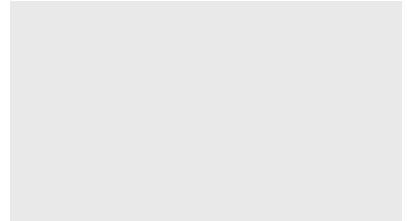
konstante Referenz:

```
int a = 17;  
const int& x = a;
```

Codebeispiel

```
int a = 17;  
int& x = a;  
std::cout << "a = " << a << std::endl;  
std::cout << "x = " << x << std::endl;  
  
++a;  
std::cout << "a = " << a << std::endl;  
std::cout << "x = " << x << std::endl;  
  
++x;  
std::cout << "a = " << a << std::endl;  
std::cout << "x = " << x << std::endl;
```

Ausgabe



Funktionsprototyp I

```
void foo_a(const Data& in,  
           Data* out);
```

Funktionsprototyp II

```
void foo_b(const Data& in,  
           Data& out);
```

Aufruf I

```
Data a;  
Data b;  
foo_a(a, &b);
```

Aufruf II

```
Data a;  
Data b;  
foo_b(a, b);
```

¹https://google.github.io/styleguide/cppguide.html#Reference_Arguments

- ▶ einführendes Beispiel
- ▶ Definition
- ▶ Referenzen vs. Zeiger als Funktionsparameter

Prof. Dr. Tom Vierjahn

► E-Mail: tom.vierjahn@w-hs.de

Visual Computing

► Web: <https://vc.w-hs.de>

► YouTube: Visual Computing WH

► Twitter: @VisComputingWH

Westfälische Hochschule

Fachbereich Wirtschaft und Informationstechnik

Campus Bocholt



Veröffentlicht unter der Creative-Commons-Lizenz

Attribution-NonCommercial-ShareAlike 4.0 International (CC BY-NC-SA 4.0)