



Vorlesung „Grundlagen der Informatik und Programmierung 1“

MODULARISIERUNG

Parameter und Variablen

Prof. Dr. Tom Vierjahn

Visual Computing (<https://vc.w-hs.de>)

Fachbereich Wirtschaft und Informationstechnik – Campus Bocholt



Wintersemester 2020/21



Veröffentlicht unter der Creative-Commons-Lizenz

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


- ▶ Die aufrufende Funktion kann nicht in die aufgerufene Funktion sehen.
- ▶ Die aufgerufene Funktion kann nicht in die aufrufende Funktion sehen.
- ▶ Von innen und außen sind nur Parameter und Rückgabe zu sehen.
- ▶ Daten werden kopiert.

Beispiel:

```
void main() {  
    int area = rectangle_area(3, 5);  
    printf("Rechteckfläche: %d\n", area);  
}
```

```
int rectangle_area(int side_a, int side_b) {  
    int result = side_a * side_b;  
    return result;  
}
```

```
int main() {  
    int data[] =  
        {3, 1, 4, 1, 5, 9};  
  
    output(6, data);  
    revert(6, data);  
    output(6, data);  
    return 0;  
}
```




```
void output(int count, int array[]) {  
    for (int i = 0; i < count; ++i) {  
        printf("%d ", array[i]);  
    }  
    printf("\n");  
}
```

```
int main() {  
    int data[] =  
        {3, 1, 4, 1, 5, 9};  
  
    output(6, data);  
    revert(6, data);  
    output(6, data);  
    return 0;  
}
```


```
void revert(int count, int array[]) {  
    for (int a = 0, z = count - 1;  
         a < z; ++a, --z) {  
        int tmp = array[z];  
        array[z] = array[a];  
        array[a] = tmp;  
    }  
}
```

```
void main() {  
    char text[] = "GIP 1";  
    printf("Länge: %d \n",  
          length(text));  
  
    char word[] = "GIP 2";  
  
    if (equal(text,  
             word)) {  
        printf("== \n");  
    } else {  
        printf("!= \n");  
    }  
}
```



```
int length(char string[]) {  
    int i = 0;  
    for (i = 0; string[i] != '\0'; ++i) {  
    }  
    return i;  
}
```

```
void main() {  
    char text[] = "GIP 1";  
    printf("Länge: %d \n",  
          length(text));  
  
    char word[] = "GIP 2";  
  
    if (equal(text,  
             word)) {  
        printf("== \n");  
    } else {  
        printf("!= \n");  
    }  
}
```



```
int equal(char a[], char b[]) {  
    int i;  
    for(i = 0; (a[i] != '\0') &&  
           (a[i] == b[i]); ++i) {  
    }  
    return a[i] == b[i];  
}
```

```
void main() {  
    char text[20];  
    text[0] = '\0';  
    char b;  
    for (b = 'a'; b <= 'k'; ++b) {  
        append(text, b);  
        printf("%s \n", text);  
    }  
}
```

```
void append(char str[], char c) {  
    int i;  
    for (i = 0;  
         str[i] != '\0'; ++i);  
    str[i] = c;  
    str[i + 1] = '\0';  
}
```

Zeichenketten verändern

Längen-Test verhindert Fehler

```
void main() {  
    char text[20];  
    text[0] = '\\0';  
    char b;  
    for (b = 'a'; b <= 'k'; ++b) {  
        bool could_append =  
            append(20, text, b);  
        if (could_append) {  
            printf("%s \\n", text);  
        }  
    }  
}
```

```
bool append(int size,  
            char str[], char c) {  
    int i;  
    for (i = 0;  
         str[i] != '\\0'; ++i);  
    if (i >= size - 1) {  
        return false;  
    }  
    str[i] = c;  
    str[i + 1] = '\\0';  
    return true;  
}
```


Gültigkeitsbereich von Variablen

```
int a = 5;
printf("%d\n", a);
{
    ++a;
    printf("%d\n", a);
}
printf("%d\n", a);
{
    int a = 19;
    printf("%d\n", a);
}
printf("%d\n", a);
a = 3;
{
    a--;
    printf("%d\n", a);
}
```


```
int counter = 0;

void main() {
    int i;
    for(i = 1; i <= 5; i++)
    {
        int x = offset(i);
        printf("%d-ter Aufruf: %d\n",
              counter, x);
    }
}
```

```
int offset(int x)
{
    int shift = 123;
    counter++;
    return x + shift;
}
```

- ▶ Blackbox-Prinzip
- ▶ Arrays und Zeichenketten als Parameter
- ▶ Gültigkeitsbereiche
- ▶ globale und lokale Variablen

Prof. Dr. Tom Vierjahn

▶  tom.vierjahn@w-hs.de

Visual Computing

▶  <https://vc.w-hs.de>

▶  VisualComputingWH

▶  Visual Computing WH

▶  @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)