

# Lösungen zu C

Blatt 4

## 1. Aufgabe

```

void wende(char String[]) {
    char *p1, *p2, tmp;           // unsere beiden Zeiger
                                   // sowie das Hilfszeichen
    p1 = p2 = String;            // alles auf den Anfang

    while(*p2++);                 // p2 an das Ende des Strings

    p2 -= 2;                       // zurück auf das Zeichen vor '\0'

    // Zeiger aufeinander zubewegen, solange sie sich noch nicht
    // erreicht haben
    for(; p1 < p2; p1++, p2--) {
        tmp = *p1;                // die Zeichen in den von p1 und
        *p1 = *p2;                // p2 referenzierten Zellen
        *p2 = tmp;                // vertauschen
    }
}

```

## 2. Aufgabe

```

int d_list_test(struct INFO *start) {

    if(start->prev != NULL)        // falls nicht erstes Element
        return 1;                 // Fehler!

    // solange Ende noch nicht erreicht und der Vorgänger des
    // Nachfolgers das aktuelle Listenelement
    while((start->next != NULL) && (start->next->prev == start))
        start = start->next;       // start eins weiter

    // falls das letzte Element erreicht wurde liefere 0 zurück,
    // sonst 1
    return(start->next != NULL);
}

```

## 3. Aufgabe / Teil a

```

#include <stdio.h>

void main(int argc, char *argv[])
{
    printf("\n\nName: %s\n", argv[0]);
    printf("Anzahl der Argumente: %d\n\n", argc - 1);
}

```

### 3. Aufgabe / Teil b

```
#include <stdio.h>

// dieses Makro bestimmt, ob es sich um einen Vokal handelt
#define Vokal(V) (((V) | 32) == 'a') || (((V) | 32) == 'e') || \
  (((V) | 32) == 'i') || (((V) | 32) == 'o') || (((V) | 32) == 'u'))

void main(int argc, char *argv[])
{
    int i, j, Vokalanzahl;

    for(i = 1; i < argc; i++) {           // durchläuft die Parameter
        Vokalanzahl = 0;                 // Vokalanzahl zurücksetzen
// durchläuft die einzelnen Buchstaben des aktuellen Parameters
        for(j = 0; argv[i][j]; j++)
            if(Vokal(argv[i][j]))       // falls es ein Vokal ist
                Vokalanzahl++;         // mitzählen
        printf("\nAnzahl der Vokale im %d. Parameter: %d", i,
            Vokalanzahl);               // Anzahl der Vokale ausgeben
    }
}
```

### 4. Aufgabe / Teil a & b

```
#include <stdio.h>

struct DEMO {
    int a;
    int b;
    double c;
};

// call by reference um die Komponente c wirklich in der
// Funktion main() zu verändern
void MW(struct DEMO *Wert)
{
    Wert->c = (Wert->a + Wert->b) / 2;
}

void main(int argc, char *argv[])
{
    struct DEMO test;

    test.a = 5;
    test.b = 7;
    MW(&test);
    printf("\n\n%5.2f\n\n", test.c);
}
```