

Übungen zu C

Blatt 4

1. Aufgabe

Schreiben Sie eine komplette C-Funktion, mit deren Hilfe man einen String umkehren kann. Die Signatur der Funktion soll

```
void wende(char String[])
```

sein.

2. Aufgabe

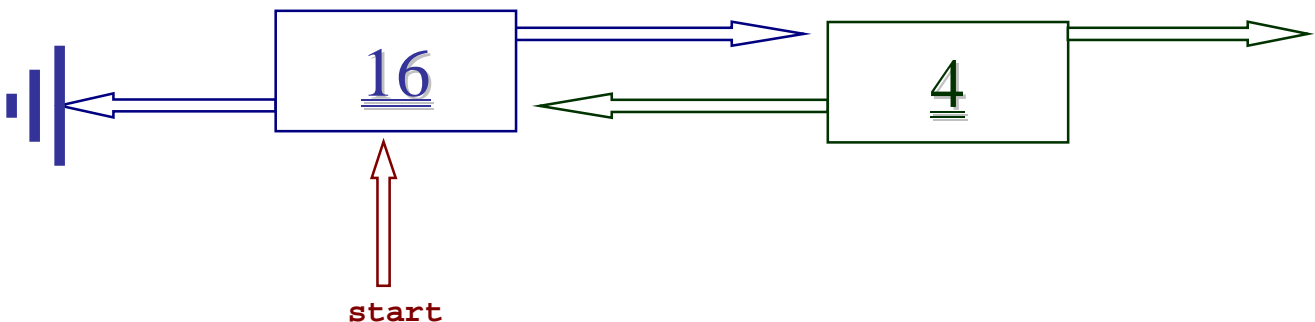
Gegeben ist folgende Struktur:

```
struct INFO {  
    int Datum;  
    struct INFO *next;  
    struct INFO *prev;  
};
```

Schreiben Sie eine Funktion mit folgender Signatur:

```
int d_list_test(struct INFO *start)
```

Die Funktion bekommt als Parameter das Startelement einer doppelt verketteten Liste (wie im Unterricht besprochen, siehe Beispiel) übergeben. Die Funktion soll überprüfen, ob die Verkettung in Ordnung ist, d.h. wenn der Nachfolger des Listenelements mit dem Datum 16 ein Listenelement mit dem Datum 4 ist, dann sollte der Vorgänger des Listenelements mit dem Datum 4 ein Listenelement mit dem Datum 16 sein.



Die Funktion liefert bei einer falsch verketteten Liste **1** zurück, sonst **0**.

Es sollen nicht [nur] die Inhalte verglichen werden, besser sind die Adressen der Listenelemente.

2. Aufgabe – Zusatzaufgabe

Wer das geschafft hat kann die Funktion so umstellen, dass man ihr ein beliebiges Element der Liste übergeben kann.

Außerdem kann man prüfen, ob es sich um eine Ringliste handelt, d.h. ob der Vorgänger des ersten Elements das letzte und der Nachfolger des letzten Elements das erste ist.

3. Aufgabe / Teil a

Schreiben Sie ein komplettes C-Programm, das seinen Namen und die Anzahl der ihm übergebenen Argumente ausgibt.

Beispiel (die Ausgabe des Programms ist **fett** gedruckt):

```
C:\>testprg para1 /23 *.ba?
```

```
Name: C:\TESTPRG.EXE
```

```
Anzahl der Argumente: 3
```

3. Aufgabe / Teil b

Ermitteln Sie die Anzahl der Vokale für jeden Parameter des Programms.

4. Aufgabe / Teil a

Gegeben ist folgende Struktur:

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

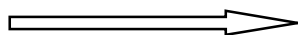
Schreiben Sie eine Funktion, die eine Variable des Typs `struct DEMO` übergeben bekommt und die Komponente `c` mit dem Mittelwert von `a` und `b` füllt. Sie soll nichts zurückliefern.

call by value oder **call by reference**?

4. Aufgabe / Teil b

Schreiben sie ein Testprogramm, das eine Variable des Typs `struct DEMO` anlegt, die Komponenten `a` und `b` füllt, die Funktion mit der Variablen aufruft und danach den Wert der Komponente `c` ausgibt. Wenn also vor dem Aufruf in `a 5` und in `b 7` stand, soll nach dem Aufruf in der Komponente `c 6` stehen.

DEMO	
int a	5
int b	7
double c	?



DEMO	
int a	5
int b	7
double c	6.0