

## 6.10.A Extras/Datum und Zeit in C – Arbeitsblatt

### Aufgabe 1: Sekundenzeit `time_t`

Wenn man mit Hilfe der Bibliotheksfunktion `time()` die aktuelle Uhrzeit aus dem System holt, erhält man einen Wert vom Typ `time_t`. Dieser Typ entspricht einer ganzen Zahl. Sie sollen nun in einem kleinen selbstgeschriebenen Programm mit der Funktion und dem Datentyp experimentieren und so Antworten auf die folgenden Fragen finden:

- a) Wie viele Bytes belegt eine Variable dieses Typs auf *Ihrem* System? Hinweis: Mit dem Operator `sizeof()` erhält man die Größe einer Variablen oder auch eines Datentyps in Bytes.

• \_\_\_\_\_

- b) Die Bibliotheksfunktion `time()` kann man auf zwei Arten nutzen:

```
1 x=time(NULL);  
2 time(&x);
```

Was passiert, wenn man bei der ersten Art den Parameter weglässt? Und was passiert, wenn man beim selben Aufruf beide Arten mischt, so dass `x` zweimal vorkommt?

• \_\_\_\_\_

• \_\_\_\_\_

- c) Erinnern Sie sich: Zu welchem Datum gehört der Wert null?

• \_\_\_\_\_

- d) Schlagen Sie nach: In welcher Zeitzone wird der Wert zurückgegeben?

• \_\_\_\_\_

- e) Wie hoch ist der Wert momentan ungefähr?

• \_\_\_\_\_

### Aufgabe 2: Aufgespaltene Zeit `struct tm`

In `time.h` wird ein zweiter Datentyp definiert, mit dem man einen Zeitpunkt angeben kann durch Jahreszahl, Monat, Tag, Stunde, Minute und Sekunde.

- a) Wie viele Felder hat `struct tm`, und wie heißen sie?

• \_\_\_\_\_

• \_\_\_\_\_

- b) Welche Bibliotheksfunktionen schreiben in einen solchen Datentyp?

• \_\_\_\_\_

- c) Welche Bibliotheksfunktionen lesen aus einem solchen Datentyp?

• \_\_\_\_\_

### Aufgabe 3: Datum und Uhrzeit ausgeben

Sie sollen nun kurze Programme schreiben, die das aktuelle Datum und/oder die aktuelle Uhrzeit in der örtlichen Zeitzone ausgeben.

- a) duzaus1.c,  
Format: 24.12.2018, 15:00:00 Uhr
- b) duzaus2.c,  
Format: 2018-12-24-15-00-00
- c) duzaus3.c,  
Format: Montag, der 24. Dezember 2018 um 15.00 Uhr
- d) duzaus4.c,  
Format: 12/24/18 15:00:00

### Aufgabe 4: Differenz zweier Zeitpunkte bestimmen

austausch20000.c: Bei einer Maschine im Dauerbetrieb soll ein wichtiges Ersatzteil nach 20000 Stunden ausgewechselt werden. Ihr Programm soll den Zeitpunkt der Inbetriebnahme über die Tastatur einlesen und mit dem aktuellen Zeitpunkt vergleichen. Anschließend soll es einen Vorschlag über den Austausch des Ersatzteils aussprechen:

```
Terminal
schueler@debian964:~$ austausch20000
Datum der Inbetriebnahme:
Tag.....: 6
Monat.....: 5
Jahr.....: 2020
Uhrzeit (Stunde): 10
Betriebszeit : 06.05.2020, 10 Uhr bis 07.05.2020, 11:48 Uhr
Betriebsdauer (aufgerundet): 26 Stunden
Die vorgesehene Betriebsdauer wurde noch nicht erreicht.
Ein Austausch wird noch nicht empfohlen.
```

### Aufgabe 5: Wochentag bestimmen

Ihr Programm soll zu einem Datum, das der Benutzer eingibt, den Wochentag ausgeben.

- a) wtag1.c: Das Datum wird im Dialog eingegeben.

```
Terminal
schueler@debian964:~$ wtag
Tag.....: 24
Monat....: 12
Jahr.....: 2018
Wochentag: Montag
```

- b) wtag2.c: Das Datum wird in der Befehlszeile eingegeben.

```
Terminal
schueler@debian964:~$ wtag 24 12 2018
Montag
```