

## Übungen zur Vorlesung Einführung in das Programmieren für TM

### Serie 2

**Aufgabe 2.1.** Schreiben Sie eine Funktion `dabs`, die für  $x \in \mathbb{R}$  den Absolutbetrag  $|x|$  zurückliefert. Schreiben Sie ferner ein aufrufendes Hauptprogramm, in dem  $x$  über die Tastatur eingelesen und  $|x|$  ausgegeben werden. Die Mathematikbibliothek `libm.so` soll nicht verwendet werden! Speichern Sie den Source-Code unter `dabs.c` in das Verzeichnis `serie02`.

**Aufgabe 2.2.** Schreiben Sie eine Funktion `bogenmass`, die einen im Gradmaß gegebenen Winkel  $\theta \in \mathbb{R}^+$  ins Bogenmaß umrechnet. Dabei soll der Rückgabewert  $\psi$  in reduzierter Form als  $\psi \in [0, 2\pi)$  zurückgegeben werden. Speichern Sie den Source-Code unter `bogenmass.c` in das Verzeichnis `serie02`.

**Aufgabe 2.3.** Schreiben Sie eine Funktion `folgenglied`, die für gegebenes  $n \in \mathbb{N}$  das Folgenglied  $a_n := (-1)^n/n$  zurückgibt. Schreiben Sie ferner ein aufrufendes Hauptprogramm, in dem  $n$  eingelesen und  $a_n$  ausgegeben werden. Speichern Sie den Source-Code unter `folgenglied.c` in das Verzeichnis `serie02`.

**Aufgabe 2.4.** Schreiben Sie eine Funktion `rundung`, die für eine gegebene Zahl  $x \in \mathbb{R}$  die ganze Zahl  $n \in \mathbb{Z}$  zurückliefert, die  $x$  am nächsten liegt. Falls  $x$  genau in der Mitte zwischen zwei ganzen Zahlen liegt, werde die größere zurückgeliefert. Schreiben Sie ferner ein aufrufendes Hauptprogramm, das die Zahl  $x$  einliest und gerundet ausgibt. Speichern Sie den Source-Code unter `rundung.c` in das Verzeichnis `serie02`.

**Aufgabe 2.5.** Schreiben Sie eine `void`-Funktion `datum`, die zu einer gegebenen Zahl  $z \in \mathbb{N}$  das zugehörige Datum geeignet ausgibt. Dabei errechnet sich das gesuchte Datum aus den Ziffern von  $z$  unter Berücksichtigung der Formatierung `TTMMJJJJ`, z.B. entspricht  $z = 10102014$  dem zehnten Oktober 2014. Man beachte, dass führende Nullen nicht gespeichert werden, z.B. wird der erste Oktober 2014 als  $z = 1102014$  repräsentiert (5 Stellen). Schreiben Sie ferner ein aufrufendes Hauptprogramm, in dem  $z$  eingelesen und die Funktion aufgerufen werden. Speichern Sie den Source-Code unter `datum.c` in das Verzeichnis `serie02`.

**Aufgabe 2.6.** Gegeben sei ein Kreis in Form seines Mittelpunkts  $(x, y) \in \mathbb{R}^2$  und seines Radius  $r > 0$ . Gegeben sei ferner ein Punkt  $(u, v) \in \mathbb{R}^2$ . Schreiben Sie eine Funktion `locate`, die zurückgibt, ob der Punkt  $(u, v)$  im Kreis (Rückgabe -1), auf der Kreislinie (Rückgabe 0) oder außerhalb des Kreises (Rückgabe 1) liegt. Schreiben Sie ferner ein aufrufendes Hauptprogramm, das die Zahlen  $x, y, r, u, v \in \mathbb{R}$  einliest, die Funktion `locate` aufruft und danach in der Shell ausgibt, wie der Punkt  $(u, v)$  im Verhältnis zum Kreis  $(x, y, r)$  liegt. Speichern Sie den Source-Code unter `locate.c` in das Verzeichnis `serie02`.

**Aufgabe 2.7.** Gegeben seien drei Punkte  $(x, y)$ ,  $(u, v)$  und  $(a, b)$  in  $\mathbb{R}^2$ . Schreiben Sie eine Funktion `punkte`, die überprüft, ob die 3 Punkte auf einer Geraden liegen. Schreiben Sie ferner ein aufrufendes Hauptprogramm, in dem die 6 Parameter eingelesen werden und das Resultat ausgegeben wird. Speichern Sie den Source-Code unter `punkte.c` in das Verzeichnis `serie02`.

**Aufgabe 2.8.** Was versteht man unter *Type-Casting*? Welche zwei Typen von *Type-Casting* gibt es? Was gibt folgender Code am Bildschirm aus? Erklären Sie auch warum!

```
#include <stdio.h>

main() {
    int x = 1;
    int y = 5;

    double erg1 = x / y;
```

```
double erg2 = (double) x / y;
double erg3 = 1. / 5;
int erg4 = (double) x / y;

printf("erg1 = %f\n",erg1);
printf("erg2 = %f\n",erg2);
printf("erg3 = %f\n",erg3);
printf("erg4 = %d\n",erg4);
}
```