

Deckblatt: Übung zur Vorlesung Informatik 1

Fakultät für Angewandte Informatik

Lehrprofessur für Informatik

PROF. DR. LORENZ, MARIUS BRENDLE, JOHANNES METZGER, LEV SOROKIN

WS 2017/18

Hinweis: Es sind alle Felder auszufüllen! Abgabe der Übungsblätter immer **mittwochs** (Ausnahme wenn Feiertag: donnerstags) bis **spätestens 12:00 Uhr** in die entsprechend gekennzeichneten Briefkästen der Veranstaltung im Erdgeschoss des Instituts für Informatik (Gebäude N). Zuwiderhandlung wird mit Strafe geahndet! (Punktabzug)

Übungsblatt	
-------------	--

(hier die Nummer des bearbeiteten **Übungsblatts** eintragen)

	Übung 01 (1055 N) Montag 08:15 - 09:45 Uhr (Lennart Eing)
	Übung 02 (1056 N) Montag 08:15 - 09:45 Uhr (Alexander Fuchs)
	Übung 03 (1057 N) Montag 08:15 - 09:45 Uhr (Michelle Lienhart)
	Übung 04 (1055 N) Montag 12:15 - 13:45 Uhr (Henning Cui)
	Übung 05 (1056 N) Montag 12:15 - 13:45 Uhr (Christian Schavitz)
	Übung 06 (1055 N) Montag 14:00 - 15:30 Uhr (Maximilian Demmler)
	Übung 08 (1056 N) Montag 17:30 - 19:00 Uhr (Moritz Feldmann)
	Übung 09 (1057 N) Montag 17:30 - 19:00 Uhr (Dat Le Thanh)
	Übung 10 (1057 N) Dienstag 12:15 - 13:45 Uhr (Alexander Szöke)
	Übung 11 (1057 N) Dienstag 14:00 - 15:30 Uhr (Denise Böhm)
	Übung 12 (1056 N) Dienstag 17:30 - 19:00 Uhr (Marvin Drexelius)
	Übung 13 (1057 N) Dienstag 17:30 - 19:00 Uhr (Tom Wolfskämpf)
	Übung 14 (1055 N) Mittwoch 08:15 - 09:45 Uhr (Jonas Junge)
	Übung 15 (1055 N) Mittwoch 10:00 - 11:30 Uhr (Elisabeth Korndörfer)
	Übung 16 (1054 N) Donnerstag 14:00 - 15:30 Uhr (Florian Magg)
	Übung 17 (1057 N) Donnerstag 14:00 - 15:30 Uhr (Lukas Lodes)
	Übung 18 (1054 N)* Donnerstag 17:30 - 19:00 Uhr (Patrick Eckert)
	Übung 19 (1058 N) Freitag 08:15 - 09:45 Uhr (Lena Tikovsky)
	Übung 20 (1054 N) Freitag 10:00 - 11:30 Uhr (Felix Fischer)
	Übung 21 (1055 N)* Freitag 14:00 - 15:30 Uhr (Isabell Rücker)
	Übung 23 (1057 N) Freitag 15:45 - 17:15 Uhr (André Schweiger)

(hier die eingeteilte **Übungsgruppe** ankreuzen)

*(1056 N bis 03.11.17)

Teamnummer	
------------	--

(hier die Nummer des eingeteilten **Teams** eintragen)

(hier die **Vor- und Nachnamen** aller Teammitglieder eintragen)

Aufgabe		
Aufgabe		
Aufgabe		
Aufgabe		
Gesamt		

(vom Tutor auszufüllen)

Übungsblatt 1

1)

a)

1)

```
int i, j;
```

2)

```
a = '<';
```

3)

3,14

4)

```
int
```

5)

Ja, double

6)

Nein, Hochkomma impliziert char, '--1' ist jedoch länger als ein zeichen.

7)

Ja, char.

8)

```
1 + 2 * 1 - 1 / 1 = 2
```

9)

```
x * x
```

b)

1)

Füllt, falls notwendig, vor der (ganzen) Zahl leerzeichen hinzu, um eine mindestbreite von 3 zu garantieren.

2)

Stellt eine Zahl mit genau 4 nachkommastellen dar.

3)

Ja. Zeige +/- explizit, fülle nullen auf bis die Ausgabe 7 zeichen lang ist.

4)

Ja, `p` gibt die Adresse eines Pointers aus.

5)

```
1.0
```

6)

```
001.1
```

c)

1)

```
a -x -test
```

2)

```
gcc a.c -o Programm
```

2)

a)

- 1: `Main` wurde groß geschrieben
- 1: Runde Klammern anstelle von geschweiften
- 1: Rückgabetyt sollte `int` sein
- 2: Geschweifte Klammern anstelle von runden
- 3: `return` muss klein geschrieben werden
- 3: Keine Klammern um `0`
- 3: Semikolon vergessen
- 4: Geschweifte Klammern anstelle von runden

```
int main(void)
{
    return 0;
}
```

b)

- 1: einfache Raute (nicht doppelt)
- 3: `int` anstelle von `char` als Rückgabetyt
- 5: Sollte `printf("Hallo");` heißen (Klammern, Anführungszeichen und Semikolon)
- nach 5: `return` Statement vergessen (`return 0;)`

```
#include <stdio.h>

int main(void)
{
    printf("Hallo\n");
    return 0;
}

/* Ausgabe: Hallo */
```

c)

- vor 1: `#include <stdio.h>` vergessen
- 1: Parameter der `main` Methode vergessen (`void`)
- 4: `%e` stellt double in Exponentenschreibweise dar, `%f` sollte hier verwendet werden
- 4: Nur zwei Nachkommastellen sind erwünscht, deshalb sollte der `printf` ausdrück `%.2f` heißen

```
#include <stdio.h>

int main(void)
{
    double d = 5.50;

    printf("%.2f\n", d);
    return 0;
}

/* Ausgabe: 5.50 */
```

d)

- 1: `<stdio.h>` anstelle von `"stdio.h"`
- 3: Parameter der `main` Methode vergessen (`void`)
- 6: Da ein einzelnes Zeichen ausgegeben werden soll, wird `%c` benötigt (als erstes Argument)
- 6: Das statement sollte `printf("%c\n", d);` heißen
- nach 7: Funktion muss wieder geschlossen werden. (`}` vergessen)

```
#include <stdio.h>

int main(void)
{
    char d = 'a';

    printf("%c\n", d);
    return 0;
}

/* Ausgabe: a */
```

3)

a)

```
#include <stdio.h>

int main(void)
{
    printf("line\n\n");

    return 0;
}
```

b)

```
#include <stdio.h>

int main(void)
{
    printf("%c\\n\n");

    return 0;
}
```

c)

```
#include <stdio.h>

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

    return 0;
}
```

d)

```
#include <stdio.h>

int main(int argc, char * argv[])
{
    printf("Vierfaches der anzahl der Parameter:%d\n", (argc - 1) * 4);

    return 0;
}
```

e)

```
#include <stdio.h>
#include <math.h>

int main(int argc, char * argv[])
{
    double result = sqrt((double)(argc - 1) * 2);

    printf("Quadratwurzel der verdoppelten Anzahl der Parameter:%.2f\n", result);

    return 0;
}
```

f)

```
#include <stdio.h>

int main(int argc, char * argv[])
{
    printf("Programmname: %s\nArgument 1:  %s\nArgument 2:  %s\n",
        argv[0],
        argv[1],
        argv[2]
    );

    return 0;
}
```

g)

```
#include <stdio.h>

int main(void)
{
    int input;

    printf("Bitte geben Sie eine Zahl ein: ");

    scanf("%d", &input);

    printf("Sie haben %d eingegeben\n", input);

    return 0;
}
```

4)

a)

```
double calc_arithmetic_mean(int a, int b)
```

b)

```
#include <stdio.h>
#include <math.h>
/* wird math.h wirklich benötigt? */

double calc_circumference_square(double length, double width)
{
    return 2 * (length + width);
}

int main(void)
{
    double length, width;

    scanf("%lf %lf", &length, &width);
    printf("Der Umfang eines Rechtecks mit Laenge %f und Breite %f ist %f\n",
        length,
        width,
        calc_circumference_square(length, width));
    return 0;
}
```

c)

```
#include <stdio.h>

int main(void)
{
    char input;

    printf("Please enter any character: ");
    scanf("%c", &input);
    printf("Previous character: %c\n",
        input - 1);

    return 0;
}
```

d)

```
#include <stdio.h>
#include <stdlib.h>

int main(void)
{
    char input;

    printf("Please enter any character: ");

    scanf("%c", &input);

    printf("Abstand zu 'A': %d\n", abs('A' - input));

    return 0;
}
```

e)

```
#include <stdio.h>
#include <math.h>

double milliseconds_to_hours(double ms)
{
    return ms / 1000 / 60 / 60;
}

int main(void)
{
    int ms;

    printf("Please enter a duration in milliseconds: ");
    scanf("%d", &ms);
    printf("%d milliseconds ammount to %.2f hours.\n",
        ms,
        milliseconds_to_hours(ms));

    return 0;
}
```