



Vorkurs Informatik

Fakultät für Angewandte Informatik

Lehrprofessur für Informatik

PROF. DR. LORENZ

Übungblatt Thema 3

Allgemeine Hinweise:

- Achten Sie bei allen Programmieraufgaben auf *Kompilierbarkeit* und *Einhaltung der Coding Conventions* (werden zu jedem Thema in den Vorlesungsfolien erklärt); auch dann, wenn es nicht explizit im Aufgabentext gefordert ist.

- Gehen Sie davon aus, dass alle Programme in der Programmiersprache C zu erstellen sind.

- Kompilieren Sie alle Ihre Programme mit den folgenden *Compiler-Schaltern*:

`-Wall -Wextra -std:c11 -pedantic`

Achten Sie darauf, dass trotz Verwendung dieser Schalter keine Fehler-/Warnmeldungen erzeugt werden.

In den folgenden Programmieraufgaben werden Sie sich mit

- dem Datentyp `char`,
- Vergleichsoperatoren, logischen Operatoren und Bedingungen,
- Zeichenfunktionen der Bibliothek `ctype.h`,
- und Fallunterscheidungen beschäftigen.

Zur Eigenrecherche können u.a. die folgenden Internetseiten dienen:

- http://de.wikibooks.org/wiki/C-Programmierung:_Einfache_Ein-_und_Ausgabe
- http://de.wikibooks.org/wiki/C-Programmierung:_Ausdr%C3%BCcke_und_Operatoren
- https://de.wikibooks.org/wiki/C-Programmierung:_ASCII-Tabelle
- https://de.wikibooks.org/wiki/C-Programmierung:_Kontrollstrukturen#Bedingungen
- https://de.wikibooks.org/wiki/C-Programmierung:_Standard_Header#ctype.h

In den Aufgaben 1 - 4 soll in jeder Teilaufgabe ein C-Programm mit einer eigenen `main`-Funktion erstellt werden. In Aufgabe 5 sollen Fehler in vorgegebenen Programmen gefunden und verbessert werden.

Aufgabe 1 (*ASCII-Zeichen, ca. 45 Minuten*)

a)

- Deklarieren Sie eine `char`-Variable `c`.
- Weisen Sie ihr den Wert `'9'` zu.
- Geben Sie den Wert von `c` als Zeichen aus.
- Geben Sie den Wert von `c` als ganze Zahl aus.

b)

- Deklarieren Sie eine `char`-Variable `c`.
- Weisen Sie ihr den Wert `'A'` zu.
- Geben Sie den Wert von `c` als Zeichen aus.
- Geben Sie den Wert von `c` als ganze Zahl aus.

c)

- Deklarieren Sie eine `char`-Variable `c`.
- Weisen Sie ihr den Wert `'\n'` zu.
- Geben Sie den Wert von `c` als Zeichen aus.
- Geben Sie den Wert von `c` als ganze Zahl aus.

d)

- Deklarieren Sie eine `char`-Variable `c`.
- Weisen Sie ihr den Wert `'\t'` zu.
- Geben Sie den Wert von `c` als Zeichen aus.
- Geben Sie den Wert von `c` als ganze Zahl aus.

e)

- Deklarieren Sie eine `char`-Variable `c`.
- Weisen Sie ihr den Wert `'\\'` zu.
- Geben Sie den Wert von `c` als Zeichen aus.
- Geben Sie den Wert von `c` als ganze Zahl aus.

f)

- Deklarieren Sie eine `char`-Variable `c`.
- Weisen Sie ihr den Wert `'='` zu.
- Geben Sie den Wert von `c` als Zeichen aus.
- Geben Sie den Wert von `c` als ganze Zahl aus.

g) Geben Sie das ASCII-Zeichen vor dem Zeichen 'a' in der ASCII-Tabelle aus, ohne in der ASCII-Tabelle nachzusehen.

h) Geben Sie das ASCII-Zeichen nach dem Zeichen 'z' in der ASCII-Tabelle aus, ohne in der ASCII-Tabelle nachzusehen.

i) Geben Sie das ASCII-Zeichen mit dem ASCII-Code 99 aus, ohne in der ASCII-Tabelle nachzusehen.

j) Geben Sie mit einem einzigen `printf`-Aufruf 10 Leerzeilen aus.

k) Erzeugen Sie mit einem einzigen `printf`-Aufruf die folgende Ausgabe:
"Zitat".

l) Erzeugen Sie mit einem einzigen `printf`-Aufruf die folgende Ausgabe:
5%

m) Erzeugen Sie mit einem einzigen `printf`-Aufruf die folgende Ausgabe:
x x x x x

n) Erzeugen Sie mit einem einzigen `printf`-Aufruf die folgende Ausgabe:
x
x
x
x
x

o) Erzeugen Sie mit einem einzigen `printf`-Aufruf die folgende Ausgabe:
x
xx
xxx
xxxx
xxxxx

p) Erzeugen Sie mit einem einzigen `printf`-Aufruf die folgende Ausgabe:
x
 x
 x
 x
 x

q) Erzeugen Sie mit einem einzigen `printf`-Aufruf die folgende Ausgabe:

Vorname	Nachname
-----	-----
Robert	Lorenz
Johannes	Metzger

Aufgabe 2 (Zeichenfunktionen, ca. 15 Minuten)

a) Geben Sie jeweils in einer eigenen Zeile den Rückgabewert des Funktionsaufrufs `isdigit(c)` für verschiedene Zeichen `c` aus:

- Für das Zeichen 'a'

-
- Für das Zeichen 'A'
 - Für das Zeichen '.'
 - Für das Zeichen '0'
 - Für das Zeichen '\n'

b) Geben Sie jeweils in einer eigenen Zeile den Rückgabewert des Funktionsaufrufs `islower(c)` für verschiedene Zeichen `c` aus:

- Für das Zeichen 'a'
- Für das Zeichen 'A'
- Für das Zeichen '.'
- Für das Zeichen '0'
- Für das Zeichen '\n'

c) Geben Sie jeweils in einer eigenen Zeile den Rückgabewert des Funktionsaufrufs `tolower(c)` für verschiedene Zeichen `c` aus:

- Für das Zeichen 'a'
- Für das Zeichen 'A'
- Für das Zeichen '.'
- Für das Zeichen '0'
- Für das Zeichen '\n'

d) Geben Sie jeweils in einer eigenen Zeile den Rückgabewert des Funktionsaufrufs `isalpha(c)` für verschiedene Zeichen `c` aus:

- Für das Zeichen 'a'
- Für das Zeichen 'A'
- Für das Zeichen '.'
- Für das Zeichen '0'
- Für das Zeichen '\n'

e) Geben Sie jeweils in einer eigenen Zeile den Rückgabewert des Funktionsaufrufs `ispunct(c)` für verschiedene Zeichen `c` aus:

- Für das Zeichen 'a'
- Für das Zeichen 'A'
- Für das Zeichen '.'
- Für das Zeichen '0'
- Für das Zeichen '\n'

f) Geben Sie jeweils in einer eigenen Zeile den Rückgabewert des Funktionsaufrufs `isspace(c)` für verschiedene Zeichen `c` aus:

- Für das Zeichen 'a'
- Für das Zeichen 'A'

-
- Für das Zeichen '.'
 - Für das Zeichen '0'
 - Für das Zeichen '\n'

Aufgabe 3 *(Bedingungen, ca. 30 Minuten)*

a)

- Deklarieren Sie eine `int`-Variable `n` und weisen ihr eine Zufallszahl als Wert zu.
- Erstellen Sie eine Bedingung für folgende Aussage:
`n` ist größer als 1000
- Weisen Sie den Wert der Bedingung einer Variable vom Typ `bool` zu und geben Sie deren Wert aus.

b)

- Deklarieren Sie eine `int`-Variable `n` und weisen ihr eine Zufallszahl als Wert zu.
- Erstellen Sie eine Bedingung für folgende Aussage:
`n` ist negativ
- Weisen Sie den Wert der Bedingung einer Variable vom Typ `bool` zu und geben Sie deren Wert aus.

c)

- Deklarieren Sie eine `int`-Variable `n` und weisen ihr eine Zufallszahl als Wert zu.
- Erstellen Sie eine Bedingung für folgende Aussage:
`n` ist eine gerade Zahl
- Weisen Sie den Wert der Bedingung einer Variable vom Typ `bool` zu und geben Sie deren Wert aus.

d)

- Deklarieren Sie eine `int`-Variable `n` und weisen ihr eine Zufallszahl als Wert zu.
- Erstellen Sie eine Bedingung für folgende Aussage:
`n` hat genau 5 Stellen
- Weisen Sie den Wert der Bedingung einer Variable vom Typ `bool` zu und geben Sie deren Wert aus.

e)

- Deklarieren Sie eine `int`-Variable `n` und weisen ihr eine Zufallszahl als Wert zu.
- Erstellen Sie eine Bedingung für folgende Aussage:
`n` ist kleiner als oder gleich 999
- Weisen Sie den Wert der Bedingung einer Variable vom Typ `bool` zu und geben Sie deren Wert aus.

f)

- Deklarieren Sie eine `int`-Variable `n` und weisen ihr eine Zufallszahl als Wert zu.

-
- Erstellen Sie eine Bedingung für folgende Aussage:
n ist ungleich 0
 - Weisen Sie den Wert der Bedingung einer Variable vom Typ `bool` zu und geben Sie deren Wert aus.

g)

- Deklarieren Sie eine `int`-Variable `n` und weisen ihr eine Zufallszahl als Wert zu.
- Erstellen Sie eine Bedingung für folgende Aussage:
n ist (ohne Rest) durch 10 teilbar und hat mindestens 3 Stellen
- Weisen Sie den Wert der Bedingung einer Variable vom Typ `bool` zu und geben Sie deren Wert aus.

h)

- Deklarieren Sie eine `int`-Variable `n` und weisen ihr eine Zufallszahl als Wert zu.
- Erstellen Sie eine Bedingung für folgende Aussage:
n ist nicht (ohne Rest) durch 10 teilbar
- Weisen Sie den Wert der Bedingung einer Variable vom Typ `bool` zu und geben Sie deren Wert aus.

i)

- Deklarieren Sie eine `int`-Variable `n` und weisen ihr eine Zufallszahl als Wert zu.
- Erstellen Sie eine Bedingung für folgende Aussage:
n ist (ohne Rest) durch 2 oder durch 3 teilbar
- Weisen Sie den Wert der Bedingung einer Variable vom Typ `bool` zu und geben Sie deren Wert aus.

j)

- Deklarieren Sie eine `char`-Variable `c`, generieren Sie eine ganze Zufallszahl und weisen Sie `c` als Wert den Rest bei ganzzahliger Division der Zufallszahl durch 128 zu.
- Erstellen Sie eine Bedingung für folgende Aussage:
Der ASCII-Code von `c` ist kleiner 64
- Weisen Sie den Wert der Bedingung einer Variable vom Typ `bool` zu und geben Sie deren Wert aus.

k)

- Deklarieren Sie eine `char`-Variable `c`, generieren Sie eine ganze Zufallszahl und weisen Sie `c` als Wert den Rest bei ganzzahliger Division der Zufallszahl durch 128 zu.
- Erstellen Sie eine Bedingung für folgende Aussage:
Der ASCII-Code von `c` ist größer als oder gleich der ASCII-Code von 'a'
- Weisen Sie den Wert der Bedingung einer Variable vom Typ `bool` zu und geben Sie deren Wert aus.

l)

- Deklarieren Sie eine `char`-Variable `c`, generieren Sie eine ganze Zufallszahl und weisen Sie `c` als Wert den Rest bei ganzzahliger Division der Zufallszahl durch 128 zu.

-
- Erstellen Sie eine Bedingung für folgende Aussage, ohne eine Bibliotheksfunktion zu benutzen:
c ist eine Ziffer
 - Weisen Sie den Wert der Bedingung einer Variable vom Typ `bool` zu und geben Sie deren Wert aus.

m)

- Deklarieren Sie eine `char`-Variable `c`, generieren Sie eine ganze Zufallszahl und weisen Sie `c` als Wert den Rest bei ganzzahliger Division der Zufallszahl durch 128 zu.
- Erstellen Sie eine Bedingung für folgende Aussage, ohne eine Bibliotheksfunktion zu benutzen:
c ist kein lateinischer Großbuchstabe
- Weisen Sie den Wert der Bedingung einer Variable vom Typ `bool` zu und geben Sie deren Wert aus.

n)

- Deklarieren Sie eine `char`-Variable `c`, generieren Sie eine ganze Zufallszahl und weisen Sie `c` als Wert den Rest bei ganzzahliger Division der Zufallszahl durch 128 zu.
- Erstellen Sie eine Bedingung für folgende Aussage:
c ist das Zeichen `' ; '`
- Weisen Sie den Wert der Bedingung einer Variable vom Typ `bool` zu und geben Sie deren Wert aus.

Aufgabe 4 (*Fallunterscheidungen, ca. 15 Minuten*)

a)

- Deklarieren Sie eine `int`-Variable `n` und weisen ihr eine Zufallszahl als Wert zu.
- Falls `n` größer ist als 1000, geben Sie 1 aus.
- Geben Sie **Ende** aus.

b)

- Deklarieren Sie eine `int`-Variable `n` und weisen ihr eine Zufallszahl als Wert zu.
- Falls `n` größer ist als 1000, geben Sie 1 aus, sonst geben Sie 0 aus.
- Geben Sie **Ende** aus.

c)

- Deklarieren Sie eine `int`-Variable `n` und weisen ihr eine Zufallszahl als Wert zu.
- Falls `n` kleiner ist als 1000, geben Sie 2 aus, falls `n` zwischen 1000 und 10000 liegt, geben Sie 1 aus, sonst geben Sie 0 aus.
- Geben Sie **Ende** aus.

d)

- Deklarieren Sie eine `int`-Variable `n` und weisen ihr eine Zufallszahl als Wert zu.
- Falls `n` kleiner ist als 1000, geben Sie 2 aus, sonst machen Sie folgendes:

-
- Falls `n` durch 2 teilbar ist, geben Sie 1 aus, sonst geben Sie 0 aus.
 - Geben Sie **Ende** aus.
- e)
- Deklarieren Sie eine `int`-Variable `n` und weisen ihr eine Zufallszahl als Wert zu.
 - Falls `n` kleiner ist als 1000, machen Sie folgendes:
 - Falls `n` durch 2 teilbar ist, geben Sie 1 aus, sonst geben Sie 0 aus.Sonst geben Sie 2 aus.
 - Geben Sie **Ende** aus.

Aufgabe 5 (*Syntaxfehler, ca. 15 Minuten*)

Die folgenden C-Programme sollten direkt nach dem Kompilieren auf der Kommandozeile ausführbar sein. Leider haben sich einige Fehler eingeschlichen! Untersuchen Sie die zugehörige(n) Fehlermeldung(en) des Compilers, schreiben Sie jeweils eine fehlerbereinigte Version des Programms und fügen Sie einen Kommentar mit einer Beschreibung des Fehlers ein.

a)

```
1 #include <stdio.h>
2
3 int main(void)
4 {
5     If (1 < 2) {
6         printf("Hallo");
7     }
8     return 0;
9 }
```

b)

```
1 #include <stdio.h>
2
3 int main(void)
4 {
5     int n;
6     8 = n;
7     return 0;
8 }
```

c)

```
1 #include <stdio.h>
2
3 int main(void)
4 {
5     int n, char c;
6     return 0;
7 }
```

d)

```
1 #include <stdio.h>
2
3 int main(void)
4 {
5     if 1 < 2 {
6         printf("Hallo");
7     }
```

```
7     }
8     return 0;
9 }
```

e)

```
1 #include <stdio.h>
2
3 int main(void)
4 {
5     char c = 'a';
6     if (1 < 2) {
7         printf("%C", c);
8     }
9     return 0;
10 }
```

f)

```
1 #include <stdio.h>
2
3 int main(void)
4 {
5     char c = 'a';
6     if (1 < 2) (
7         printf("%c", c);
8     )
9     return 0;
10 }
```

g)

```
1 #include <stdio.h>
2
3 int main(void)
4 {
5     char c = 'a';
6     if (1 < 2) {
7         printf("%c", c);
8     }
9     return 0;
10 }
```

h)

```
1 #include <stdio.h>
2
3 int main(void)
4 {
5     char c = 'a';
6     if (1 < 2) {
7         printf("%c", c);
8     };
9     return 0;
10 }
```

i)

```
1 #include <stdio.h>
2
3 int main(void)
4 {
5     bool b = true;
6     printf("%i", b);
7     return 0;
8 }
```