



Universität Augsburg  
Institut für Informatik

Übung zur Vorlesung Informatik 1

WS 2017/18

Fakultät für Angewandte Informatik

Lehrprofessur für Informatik

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

8.11.2017

---

## Übungsblatt 3

---

**Abgabe: 15.11.2017, 12:00 Uhr** (Postkasten der Veranstaltung und E-Mail an Tutor)

- Dieses Übungsblatt muss im Team abgegeben werden (Einzelabgaben sind nicht erlaubt).
- Bitte zur Angabe von Namen, Übungsgruppe und Teamnummer das **Deckblatt** verwenden!
- Die **Zeitangaben** geben zur Orientierung an, wie viel Zeit für eine Aufgabe später in der Klausur vorgesehen wäre; gehen Sie davon aus, dass Sie zum jetzigen Zeitpunkt wesentlich länger brauchen und die angegebene Zeit erst nach ausreichender Übung erreichen.

\* leichte Aufgabe / \*\* mittelschwere Aufgabe / \*\*\* schwere Aufgabe

---

### Aufgabe 9 \* (*Wissensfragen, jede Frage 1 Minute*)

Beantworten Sie möglichst knapp und genau in einem Satz die folgenden Fragen.

a) (Gültigkeit von Ausdrücken)

1. Ist die Zeichenfolge `!1` ein gültiger Ausdruck in C? Wenn nein, warum?
2. Ist die Zeichenfolge `!!0` ein gültiger Ausdruck in C? Wenn nein, warum?
3. Ist die Zeichenfolge `== 2` ein gültiger Ausdruck in C? Wenn nein, warum?
4. Ist die Zeichenfolge `0 ! 0` ein gültiger Ausdruck in C? Wenn nein, warum?
5. Ist die Zeichenfolge `x < 15 < y` ein gültiger Ausdruck in C? Wenn nein, warum?
6. Ist die Zeichenfolge `!x > y && z` ein gültiger Ausdruck in C? Wenn nein, warum?

b) (Auswertung von Ausdrücken)

1. Welchen Wahrheitswert hat der Ausdruck `1`?
2. Welchen Wahrheitswert hat der Ausdruck `1.05`?
3. Welchen Wahrheitswert hat der Ausdruck `0.02`?
4. Welchen Wahrheitswert hat der Ausdruck `!1.05`?
5. Welchen Wahrheitswert hat der Ausdruck `1 < 0`?
6. Welchen Wahrheitswert hat der Ausdruck `!(1 < 0)`?
7. Welchen Wahrheitswert hat der Ausdruck `1 && 0`?
8. Welchen Wahrheitswert hat der Ausdruck `!1 && 0`?
9. Welchen Wahrheitswert hat der Ausdruck `!(1 && 0)`?
10. Welchen Wahrheitswert hat der Ausdruck `'A' != 66`?
11. Welchen Wahrheitswert hat der Ausdruck `'B' == 'b'`?
12. Welchen Wert hat der Ausdruck `0 ? 0 : 1`?
13. Welchen Wert hat der Ausdruck `1 ? (1 ? 0 : 1) : 0`?
14. Welchen Wert hat der Ausdruck `isdigit('0')`?
15. Welchen Wert hat der Ausdruck `!toupper('b')`?

---

**Aufgabe 10** \*\* (*Kommunikation zwischen Steuerwerk und Speicherwerk, 15 Minuten*)

a) (Lesen aus dem Speicherwerk \*\*, 7 Minuten)

Beschreiben Sie detailliert die Kommunikation zwischen Steuerwerk und Hauptspeicher über Register und Steuersignale beim *Lesen* des in der Speicherzelle mit der Adresse 100 gespeicherten Werts.

b) (Schreiben in das Speicherwerk \*, 8 Minuten)

Beschreiben Sie detailliert die Kommunikation zwischen Steuerwerk und Hauptspeicher über Register und Steuersignale beim *Schreiben* des Werts 5 in die Speicherzelle mit der Adresse 200.

---

### Aufgabe 11 \* (*Wahrheitstafeln, 21 Minuten*)

In jeder Teilaufgabe sollen Sie für eine oder mehrere Bedingungen Wahrheitstafeln erstellen, die jeweils alle Kombinationen von Wahrheitswerten der benutzten Operanden enthalten (s. Folien Kapitel 4). Die Operanden werden dabei mit **A**, **B**, **C** bezeichnet. Kommt ein Operand mehrmals in einer Bedingung vor, so ist jeweils für jedes Vorkommen derselbe Wahrheitswert einzusetzen.

a) (\*, 2 Minuten)

Erstellen Sie eine Wahrheitstafel für die Bedingung  $\neg(\neg(\neg A))$ .

b) (\*, 4 Minuten)

Erstellen Sie eine Wahrheitstafel für die Bedingung  $A \vee (A \wedge B)$ .

c) (\*, 2 Minuten)

Erstellen Sie eine Wahrheitstafel für die Bedingung  $A \wedge A$ .

d) (\*, 5 Minuten)

Erstellen Sie Wahrheitstafeln für die Bedingungen  $\neg(A \vee B)$  und  $(\neg A) \wedge (\neg B)$ . Man sagt, die beiden Bedingungen  $\neg(A \vee B)$  und  $(\neg A) \wedge (\neg B)$  sind **äquivalent**. Was heißt das? (Eigene Recherche)

*Anmerkung:* Auch die Bedingungen  $\neg(A \wedge B)$  und  $(\neg A) \vee (\neg B)$  sind **äquivalent**.

e) (\*, 8 Minuten)

Erstellen Sie Wahrheitstafeln für die Bedingungen  $A \wedge (B \wedge C)$  und  $(A \wedge B) \wedge C$ . Auch diese beiden Bedingungen sind **äquivalent**. Aufgrund dieser Eigenschaft nennt man den  $\wedge$ -Operator **assoziativ** (siehe auch Folien Kapitel 4).

*Anmerkung:* Auch der  $\vee$ -Operator ist **assoziativ**.

---

## Aufgabe 12 \*\* (*Gültigkeit von Benutzereingaben, 20 Minuten*)

In jeder Teilaufgabe sollen Sie jeweils ein Hauptprogramm implementieren, das eine bestimmte Benutzereingabe erwartet. Das Programm soll jeweils

- eine Benutzereingabe einlesen,
- diese auf Gültigkeit überprüfen und
- in geeigneten Variablen speichern, und
- schließlich auf Kommandozeile ausgeben, ob die Eingabe gültig war oder nicht.

Die Verwendung folgender Bibliotheksfunktionen ist erlaubt: `printf`, `scanf`, `getchar`

a) (\*, 4 Minuten)

Schreiben Sie ein Hauptprogramm, das als Benutzereingabe einen `integer`-Wert erwartet.

b) (\*\*, 4 Minuten)

Schreiben Sie ein Hauptprogramm, das als Benutzereingabe zwei nichtnegative `double`-Werte, getrennt durch ein Leerzeichen, erwartet.

c) (\*\*, 4 Minuten)

Schreiben Sie ein Hauptprogramm, das eine Benutzereingabe erwartet, die aus genau zwei ASCII-Zeichen besteht.

d) (\*\*, 4 Minuten)

Schreiben Sie ein Hauptprogramm, das als Benutzereingabe eine ganze Zahl zwischen  $-10$  und  $10$  (jeweils einschließlich) erwartet.

e) (\*\*\*, 4 Minuten)

Schreiben Sie ein Hauptprogramm, das als Benutzereingabe eine positive ganze Zahl unmittelbar gefolgt von höchstens einem Kleinbuchstaben erwartet.

---

Es gibt folgende Arten von **Zusatzaufgaben**:

- Aufgaben zu **Schulstoff**, der Voraussetzung zum Verständnis der Vorlesung ist. Es wird davon ausgegangen, dass Sie diesen (Schul-)Stoff bereits beherrschen.
- **Alte Klausuraufgaben** zu Themen des Übungsblatts.

Besprechungen / Lösungen zu Zusatzaufgaben:

- Alle Zusatzaufgaben sind **unbewertet** und dementsprechend nicht abzugeben.
- Alle Zusatzaufgaben werden in der **Globalübung** besprochen.

### **Zusatzaufgabe C \*\*\*** (*Klausuraufgabe aus dem Wintersemester 2016/2017, 6 Minuten*)

Implementieren Sie eine Funktion `int read_semester(void)`, die vom Benutzer eine ganze Zahl zwischen 1 und 9 (jeweils einschließlich) einliest und im Erfolgsfall zurückgibt.

- Bei *ungültigen Eingaben* soll die Funktion den Eingabepuffer leeren und 0 zurückgeben.
- Bei allgemeinen Lesefehlern (auch *Pufferfehlern* genannt) soll sie **EOF** zurückgeben.
- Zum *Leeren des Eingabepuffers* steht die folgende Funktion zur Verfügung:  
(diese muss nicht von Ihnen implementiert werden!)  
`int flush_buff(void)`: Gibt 0 zurück, falls ein Pufferfehler auftritt, sonst 1.

### **Zusatzaufgabe D \*** (*Wiederholung von Schulstoff: Exponential- und Logarithmus-Funktion*)

Lösen Sie diese Aufgaben ausnahmslos ohne Hilfsmittel wie Taschenrechner oder ähnliches (stehen Ihnen auch in der Klausur nicht zur Verfügung). Mit genug Übung sollten Sie für jede dieser Aufgaben jeweils nicht mehr als ca. 10 Sekunden brauchen.

- Schreiben Sie die Zahl 1 in der Form  $2^n$  für eine passende ganze Zahl  $n \in \mathbb{Z}$ .
- Schreiben Sie die Zahl 2 in der Form  $2^n$  für eine passende ganze Zahl  $n \in \mathbb{Z}$ .
- Schreiben Sie die Zahl 1024 in der Form  $2^n$  für eine passende ganze Zahl  $n \in \mathbb{Z}$ .
- Schreiben Sie die Zahl 0.5 in der Form  $2^n$  für eine passende ganze Zahl  $n \in \mathbb{Z}$ .
- Schreiben Sie die Zahl 0.125 in der Form  $2^n$  für eine passende ganze Zahl  $n \in \mathbb{Z}$ .
- Schreiben Sie die Zahl 0 in der Form  $\log_2(n)$  für eine passende ganze Zahl  $n \in \mathbb{N}$ .
- Schreiben Sie die Zahl 5 in der Form  $\log_2(n)$  für eine passende ganze Zahl  $n \in \mathbb{N}$ .
- Schreiben Sie die Zahl 10 in der Form  $\log_2(n)$  für eine passende ganze Zahl  $n \in \mathbb{N}$ .
- Schreiben Sie die Zahl  $-2$  in der Form  $\log_2(n)$  für eine passende ganze Zahl  $n \in \mathbb{R}$ .
- Berechnen Sie  $2^3 \cdot 2^7$ .
- Berechnen Sie  $2^{-3} \cdot 2^7$ .
- Berechnen Sie  $\frac{2^3}{2^7}$ .
- Berechnen Sie  $\log_2(8)$ .
- Berechnen Sie  $\log_2(4) + \log_2(8)$ .
- Berechnen Sie  $\log_2(4) + \log_2(0.125)$ .
- Berechnen Sie  $\log_2(4) - \log_2(0.125)$ .
- Berechnen Sie  $\log_2(8^{10})$ .
- Berechnen Sie  $2^{\log_2(64)}$ .
- Berechnen Sie  $\log_2(2^{64})$ .