

## Übungsblatt 7: Programmieren in C (WS 2019/20)

Abgabe: Montag, **13.01.20**, 12:00

### Rettet das Weihnachtsfest!

Bei der Auftaktbesprechung der diesjährigen Weihnachtssaison verkündet der Weihnachtsmann: “Liebe Weihnachtswichtel, Engel und Elfen! Wir werden dieses Jahr die Planung der Geschenke und Verarbeitung der Listen mittels neuester Technologie angehen. Wir steigen um von der Elfen-verarbeitenden Datenverarbeitung auf die Elektronische Datenverarbeitung!”

Ihre Aufgabe ist es, dem Weihnachtsmann dabei zu unterstützen und Programme zu schreiben, die die Aufgaben der Wichtel, Engel und Elfen automatisieren.

#### **Aufgabe 1** *Weihnachtspapier bedrucken (4 Punkte)*

**Abgabe: muster.c in Exclaim**

“Wir brauchen mehr Weihnachtspapier zum Verpacken der Geschenke”, sagt der Oberwichtel. “Und zwar mit Tannenbaum-Muster! Dann gibt es auch keine Missverständnisse, wo die Elfen die Geschenke ablegen sollen.”

Schreiben Sie ein Programm, das als Eingabe eine ungerade Zahl  $N$  einliest und folgende Ausgaben erstellt (hier für  $N=9$ ):

```
000010000
000111000
001111100
011111110
111111111
```

$N$  ist dabei die Anzahl der Zeichen pro Zeile. Schreiben Sie Ihre Lösung in zwei Varianten: mit einer `for`- und mit einer `while`- Schleife. Die Ausgabe soll dann das Muster zweimal enthalten, mit einer Leerzeile getrennt.

[Zusatzüberlegung (keine Abgabe erforderlich): Warum ist eine Lösung mit einer `do-while` Schleife nicht gut umsetzbar?].

#### **Aufgabe 2** *Gleiche Geschenke (6 Punkte)*

**Abgabe: Flussdiagramm und gemeinsam.c in Exclaim**

Um die Einzigartigkeit der Menschen hervorzuheben, wollen die Elfen verhindern, dass zwei Personen die gleichen Geschenke bekommen. Jedes Geschenk hat eine Identifikationsnummer und für jede Person haben die Elfen eine Liste mit den Geschenknummern erstellt.

Schreiben Sie ein C-Programm, das zwei Folgen von  $N$  int-Werten einliest und die Anzahl der Zahlen ausgibt, die in beiden Folgen vorkommen. Sie können davon ausgehen, dass keine Zahl in einer der Folgen mehrmals vorkommt.

*Hinweise:*

- Definieren Sie  $N$  als Präprozessor-Konstante mit Wert 10 für die Testfälle in Exclaim.
- Erstellen Sie zunächst ein Flussdiagramm für Ihre Lösungsidee.

- Wenn Testfälle fehlschlagen, führen Sie Ihre Implementierung von `gemeinsam.c` schrittweise im Debugger aus, wie in der Vorlesung bzw. den Präsenzübungen gezeigt. Setzen Sie dazu geeignet Breakpoints an die jeweiligen Schleifen, um die Veränderung der Variablenwerte in den einzelnen Schleifendurchläufen nachzuvollziehen.

*Beispiel:* Bei Eingabe von

```
0 1 2 3 4 5 6 7 8 9
10 11 12 13 14 15 16 3 2 1
```

soll als Ausgabe 3 erfolgen.

### Aufgabe 3 Wunschzettelanalyse (5 Punkte)

#### Abgabe: `stringanalyse.c` in Exclaim

Verwaltungsenkel Bernd runzelt seine Stirn. “Wir müssen unser Qualitätsmanagement dringend verbessern und dazu brauche ich Daten, Daten, Daten.”

Schreiben Sie ein C-Programm, das einen String von der Konsole einliest und folgende Information ermittelt:

- Länge des Strings
- Ist der String ein Palindrom?

Verwenden Sie dazu **keine** Bibliotheksfunktionen aus der String-Bibliothek. Definieren die maximale Länge des Strings als Präprozessor-Konstante mit Wert 61, um genügend Speicherplatz für max. 60 Zeichen plus dem Stringterminalsymbol `\0` bereitzuhalten.

Um den String einzulesen, verwenden Sie bitte folgende Anweisungen:

```
char eingabe[N];
fgets(eingabe, N, stdin);
```

Dabei soll `N` den Wert 61 haben, um genügend Speicherplatz für max. 60 Zeichen plus dem Stringterminalsymbol `\0` bereitzuhalten.

*Beispiel:* Bei Eingabe von `C ist die beste Programmiersprache der Welt` soll folgende Ausgabe erfolgen:

```
Der String hat die Laenge 43 und ist kein Palindrom.
```

Bei Eingabe von `RELIEFPFEILER` soll folgende Ausgabe erfolgen:

```
Der String hat die Laenge 13 und ist ein Palindrom.
```

*Hinweis:* Um Ihre Lösung lokal zu testen, sollten Sie Ihren Eingabestring in einer Textdatei (z.B. `eingabe.txt`) abspeichern und diese bei der Ausführung auf Kommandozeile folgendermaßen als Eingabe nutzen:

```
gcc -o stringanalyse stringanalyse.c
./stringanalyse < eingabe.txt
```

*Erklärung:* Um einen String auf sichere Art einzulesen, verwenden wir die Funktion `fgets(char * str, int n, FILE * stream)`, die `n` Zeichen aus einem Eingabestrom in den Speicherbereich schreibt, der durch `str` referenziert wird. Diese Funktion nimmt einen Buffer `s` der Größe `n` und einen Eingabe-Strom. Wenn Sie `stdin` als Eingabe-Strom verwenden, wird von der Standardeingabe gelesen. Die Funktion liest bis eine neue Zeile startet, das Ende des Stroms erreicht ist oder die maximale Länge `n-1` erreicht ist. Rückgabe der Funktion ist `s`, falls das Einlesen erfolgreich war und `NULL` falls das Ende des Stroms erreicht wurde oder ein Fehler auftritt. Zeilenumbrüche am Ende der Zeile werden in `s` beibehalten.

#### Aufgabe 4 Häufigkeit von Buchstaben (5 Punkte)

**Abgabe:** `buchstaben.c` in **Exclaim**

Verwaltungsenkel Bernd ist noch nicht zufrieden. “Ich will eine buchstabengenaue Analyse der Wunschzettel! Dazu müssen wir wissen, welche Buchstaben wie häufig auftreten.”

Schreiben Sie ein C-Programm, das für einen eingegebenen String die Häufigkeit einzelner Buchstaben ermittelt und **in einem Array abgespeichert werden**. Der String verwendet ausschließlich Großbuchstaben und umfasst maximal 60 Zeichen. Verwenden Sie wie in der vorherigen Aufgabe beschrieben `fgets` zum Einlesen des Strings von `stdin`.

*Beispiel*

*Eingabe:*

*Ausgabe:*

ABCDEFGHIJKLMNOPQRSTUVWXYZ

A: 1  
B: 1  
C: 1  
D: 1  
E: 1  
F: 1  
G: 1  
H: 1  
I: 1  
J: 1  
K: 1  
L: 1  
M: 1  
N: 1  
O: 1  
P: 1  
Q: 1  
R: 1  
S: 1  
T: 1  
U: 1  
V: 1  
W: 1  
X: 1  
Y: 1  
Z: 1

#### Aufgabe 5 Wunschzettelverschlüsselung (10 Punkte)

**Abgabe:** `entschluesselung.c`

Um Daten sicher zu kommunizieren, verwendet man heutzutage asymmetrische Verschlüsselungsalgorithmen wie RSA. Die Weihnachtswichtel verwenden einen wesentlich einfacheren Algorithmus, um Wunschzettel vor fremden Blicken zu schützen. Sie nutzen dazu einen Ersetzungsalgorithmus, bei dem jeder Buchstabe im Ursprungstext durch einen anderen Buchstaben aus dem Alphabet ersetzt wird. Der Abstand zwischen den Buchstaben im Ursprungstext und dem verschlüsselten Text ist dabei konstant. Wenn man den konstanten Abstand  $d$  mit 5 definiert, wird der Buchstabe A durch F, B durch G, C durch H, ..., Y durch D, Z durch E ersetzt.

Mit einem bekannten konstanten Abstand  $d$  wäre es ziemlich einfach die Verschlüsselung zu knacken. Die Wichtel verwenden daher einen zufälligen Abstand für jeden neuen Eintrag. Um die Einträge zu entschlüsseln, müssen sie daher für jeden Eintrag den Abstand  $d$  neu ermitteln. Durch einen Glühwein-Unfall in der Wichtel-IT-Abteilung wurden nun aber die Daten zur Entschlüsselung zerstört.

Um den Wichteln zu helfen und die Zettel zu entschlüsseln, nutzen wir folgende Taktik: Der

Buchstabe E ist in deutschen Texten der häufigste Buchstabe. Basierend auf dieser Information können wir den Abstand für einen Wunschzettel ermitteln!

Schreiben Sie ein C-Programm, das für einen eingegebenen Text den Abstand  $d$  ermittelt und den Text entschlüsselt. Der Text verwendet ausschließlich Großbuchstaben und Leerzeichen und umfasst maximal 60 Zeichen. Leerzeichen werden nicht verschlüsselt. Falls eine Entschlüsselung nicht möglich ist, da es mehrere potentielle Distanzen gibt, die verwendet werden könnten, geben Sie aus: `Entschluesselung nicht moeglich` Bei Eingabe von `PHLQH OLVVH` soll der Text `MEINE LISTE` ausgegeben werden.

*Hinweise:*

1. Verwenden Sie Ihre Lösung zur Ermittlung der Anzahl von Buchstaben!
2. Bei der Eingabe von Strings auf der Kommandozeile wird bei Verwendung von `fgets` der Zeilenumbruch als Zeichen dem Eingabestring hinzugefügt. Dieses zusätzliche Zeichen im umgewandelten Text können Sie beim Testen ignorieren.<sup>1</sup>

## Aufgabe 6 *Die fleißigsten Studierenden (freiwillig)*

Die Engel haben das Jahr Aufzeichnungen angefertigt, wer brav war und eine Liste aller Studierenden ermittelt, die regelmäßig an den Übungen teilgenommen haben. Um die Auszeichnung "Fleißigster Studi 2018" zu vergeben, benötigt der Weihnachtsmann für jeden Studiengang die Person mit der höchsten Punktzahl auf den gesamten Übungsblättern.

Ihre Aufgabe ist es ein Programm zu schreiben, das aus der Liste der Engel den fleißigsten Studierenden ermittelt und ausgibt.

1. Laden Sie von der Vorlesungshomepage die Vorlage `studis.c` herunter und ergänzen Sie diese wie in den nächsten Schritten beschrieben.
2. Modellieren Sie Studierende als einen `struct Studi` mit folgenden Strukturvariablen:
  - Name mit max. 20 Zeichen als String (`name`)
  - Punktezahl als Integer (`punkte`)
  - Studiengang als Wert eines `enum Fach {EIT, INF, MATH}` (`studiengang`)
3. In der Vorlage finden Sie eine Funktion `void init(struct Studi *s, char *eingabe)`, die die Informationen aus dem String `eingabe` nimmt und in die Struktur `s` einträgt. Beschreiben Sie (als Kommentar) die Funktionsweise von `init`; d.h. die einzelnen Schritte sowie die Verwendung von `strtok`. Erläutern Sie anhand von `"Michael Meier,34,INF"`, wie sich der Eingabestring `eingabe` durch Aufruf der Funktion `init` verändert.
4. Schreiben Sie eine Funktion `void busyStudi(Studi* studis, int n, Fach f)` die aus einem Array von `n` Studierenden-Einträgen den Namen des Studierenden mit der höchsten Punktzahl aus einer Fachrichtung `f` ermittelt und ausgibt.
5. Schreiben Sie eine `main`-Funktion, die wie in Aufgabe 1 oben beschrieben, die zunächst die Daten der Studierenden zeilenweise einliest und ein Array von Strukturen des `Studi` füllt. Die Eingabe soll dabei max. 100 Einträge enthalten. Die Ausgabe des Programms soll den Namen des fleißigsten Studis des Fachbereichs EIT beinhalten.

*Beispiel:*

*Eingabe:*

*Ausgabe:*

```
Michael Maier,12,INF    Fleissigster Studi der EIT: Marja Martin
Marja Martin,56,EIT
Berta Bubble,23,INF
Karl Klein,34,EIT
Ulli Schmitt,87,MATH
Yan Ying,34,EIT
```

<sup>1</sup>Alternativ können Sie Ihren Eingabestring in einer Textdatei abspeichern und diese bei der Ausführung auf Kommandozeile folgendermaßen als Eingabe nutzen:

```
gcc -o entschluesselung entschluesselung.c
./entschluesselung < eingabe.txt
```