

Lösungshinweise zum Übungsblatt 2: Programmieren in C (WS 2018/19)

1. Zur Beantwortung von Fragen und Hilfe bei Problemen stehen wir Ihnen immer in der Praktische Übung, Mittwoch 15:30, Raum 32-410-PC zur Verfügung.
2. Sie erreichen uns per Email unter pinc-support@cs.uni-kl.de!
3. Sie können sich im Exclaim System unter <https://softtech.cs.uni-kl.de/exclaim> mit Ihrem persönlichen STATS-Account einloggen und Dateien zu den einzelnen Übungen hochladen.
4. Die Aufgaben auf diesem Übungsblatt sind alle freiwillig zu bearbeiten.

Aufgabe 1: Compiler-Ausgaben verstehen

Abgabe: Exclaim als `zinsen.c`

1. Finden Sie die Fehler im folgenden Programm und notieren Sie diese, indem Sie diese z.B. auf diesem Blatt markieren (ohne Hilfe eines Compilers!).

```
#include <stdio.h>

/ Programm zur Berechnung von Kapitalertrag

void start (void) {
    integer laufzeit = 10;
    double grundkapital = 1000.00;
    double kapital = grundkapital
    int zins = 5.0;
    int jahr = 1;

    printf ("Tabelle fuer Grundkapital %7.2f EUR\n", grundkapital);
    printf ("Kapitalstand zum Jahresende:\n");
    while jahr <= laufzeit {
        printf ("\nJahr: %2d\t");
        kapital = kapital * (1. + zins/100.);
        printf (Kapital: %7.2f EUR, kapital);
        jahre <- jahre + 1;
    }
    /* Ausgabe
    printf ("\n\nAus %7.2f EUR Grundkapital\n", grundkapital);
    printf ("wurden in %d Jahren %7.2f EUR\n"; laufzeit, kapital);
    return 1;
}
```

2. Laden Sie sich nun das Programm von der Vorlesungshomepage herunter und korrigieren Sie die Fehler.

```
#include <stdio.h> // Spitze Klammern ergaenzt

// Programm zur Berechnung von Kapitalertrag // Kommentar mit zwei // oder /* ... */

int main (void) { // Programmausfuerung startet bei main, Rueckgabe ist ein int-Wert
    int laufzeit = 10; // Datentyp ist int, nicht integer
    double grundkapital = 1000.00; // Datentyp ist double
    double kapital = grundkapital;
    double zins = 5.0; // Richtiger Datentyp ist double hier
    int jahr = 1;
    printf ("Tabelle fuer Grundkapital %7.2f EUR\n", grundkapital);
    printf ("Kapitalstand zum Jahresende:\n");
    while (jahr <= laufzeit) {
        printf ("\nJahr: %2d\t", jahr);
        // Formatierungsstring erwartet Integer-Variable
        kapital = kapital * (1. + zins/100.);
    }
}
```

```

        printf ("Kapital: %7.2f EUR", kapital); // Formatierungsstring mit "..."
        jahr = jahr + 1; // Zuweisung durch =
    }
    /* Ausgabe */ // Kommentarblock mit /* ... */
    printf ("\n\nAus %7.2f EUR Grundkapital\n", grundkapital);
    printf ("wurden in %d Jahren %7.2f EUR\n", laufzeit, kapital);
    // Parameter mit Komma getrennt
    return 0; // Rueckgabewert bei erfolgreicher Ausfuehrung sollte 0 sein
}

```

Aufgabe 2: Flussdiagramm

Abgabe: -

Der folgende Algorithmus ermittelt, ob eine Zahl eine Primzahl ist:

Geg.: natürliche Zahl n
 Teste nacheinander alle Zahlen von 2 bis $n-1$, ob bei der ganzzahligen Division von n durch eine dieser Zahlen ein Rest von 0 bleibt. In diesem Fall gebe aus "keine Primzahl", andernfalls "Primzahl".

Geben Sie eine Beschreibung des Algorithmus' in Form eines Flussdiagramms an!

Hinweis: Der Rest bei der ganzzahlige Division wird durch den Modulo-Operator % beschrieben (z.B. $16 \% 3$ ergibt 1).

Aufgabe 3: Quersumme

Abgabe: Exclaim als `zinsen.c`

Implementieren Sie den Algorithmus zur Berechnung der Quersumme aus Kapitel 01. Ergänzen Sie dazu folgenden Programmtext:

```

#include <stdio.h>

int main (void)
{
    // Einlesen der Eingabe in x
    int x;
    scanf ("%d", &x);

    // Hier zu ergaenzen: Berechnung der Quersumme und Ausgabe

    return 0;
}

```

Die Ausgabe des Programms soll dabei bei Eingabe von 5923 folgendermaßen gestaltet sein:

Quersumme: 19

```

#include <stdio.h>

int main(void)
{
    int x;
    scanf ("%d", &x);

    int qs = 0;
    while (x > 0) {
        qs = qs + x%10;
        x = x/10;
    }

    printf ("Quersumme: %d\n", qs);
    return 0;
}

```