

Lösung Übung 2

infT

1.

Sind die folgenden Aussagen wahr oder falsch?

```

a) int a = 5, b = 18;           // Loesung
   if (a > 0 && b != 18)      // false

b) int zahl = 10, wert = 100;
   if (zahl != 0 || wert - zahl == 90) // true (short-circuit bool. eval.)

c) double x = 1.0, y = 5.7;
   if (x >= .9 && y <= 5.8) // true

d) int n1, n2 = 17;
   if (n1 > 0 && n2 > 0 || n1 > n2 && n2 != 17) // n1 nicht initialisiert!

```

2.

Überlegen Sie, was das Programm tut, und ergänzen Sie die beiden `printf`-Anweisungen in sinnvoller Weise. Benutzen Sie zur Lösung eine ASCII-Tabelle.

```

#include <stdio.h>

int main() {
    char bstb; // haelt char von stdin

    printf("\nBitte keine Umlaute eingeben!\n");
    printf("\nBuchstabe: ");
    bstb = getchar(); // Warte auf Eingabe eines Zeichens stdin
    if (bstb < 'A' || bstb > 'z' || bstb > 'Z' && bstb < 'a') {
        printf("\nDas ist kein Buchstabe!");
    }
    else if (bstb >= 'A' && bstb <= 'Z') {
        printf("\nSie haben einen Grossbuchstaben eingegeben."); // Loesung
    }
    else {
        printf("\nSie haben einen Kleinbuchstaben eingegeben."); // Loesung
    }
    return 0;
}

```

3.

Schreiben Sie ein Programm, das einen Temperaturwert abfragt und danach anzeigt, ob Wasser verdampft, flüssig bleibt oder gefriert.

Lösung:

```

#include <stdio.h>

int main() {
    double celsius;

    printf("\nGeben Sie einen Temperaturwert in Grad Celsius ein: ");
    scanf("%lf", &celsius);
    if (celsius < -273.15) {
        printf("\nWert unter dem absoluten Nullpunkt!");
    }
    else if (celsius < 0.0) {
        printf("\nDas Wasser gefriert.");
    }
    else if (celsius < 100.0) {
        printf("\nDas Wasser ist fluessig.");
    }
}

```

```

else {
    printf("\nDas Wasser verdampft.");
}

return 0;
}

```

4.

Geben Sie für die folgenden `for`-Schleifen an, wie oft sie durchlaufen werden und welche Werte die Zählvariable dabei annimmt.

- `for (zaehler = 0; zaehler != 10; zaehler = zaehler + 1)`
zaehler wird 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, also 10 x durchlaufen; Endwert zaehler: 10
- `for (n = 0; n > 10; n = n - 1)`
n wird 0, n > 10 ist falsch, also nie durchlaufen; Endwert n: 0
- `for (x = 1; x <= 15; x = x + 3)`
x wird 1, 4, 7, 10, 13, 16, also 5 x durchlaufen; Endwert x: 16
- `for (anfang = 0; anfang < 3; anfang = anfang + .5)`
float anfang wird 0, 0.5, 1.0, 1.5, 2.0, 2.5, 3.0, also 6 x durchlaufen; Endwert anfang: 3.0
- `for (i = 10; i < 0; i += 2)`
i wird 10, also nie durchlaufen, Endwert i: 10

5.

Das folgende Programm rechnet Temperaturen von Grad Celsius in Werte mit anderen Einheiten um. Warum arbeitet es nicht korrekt?

```

#include <stdio.h>
#include <ctype.h>

int main() {
    float temperatur;
    char zeichen;

    printf("\n\tT E M P E R A T U R R E C H N E R\n");
    printf("\n1 Fahrenheit\n2 Reaumur\n3 Kelvin\n4 Rankine\n0 Ende");
    printf("\n\nAuswahl: ");
    zeichen = getchar();
    switch (zeichen) {
        case '0': // Korrektur Apostroph
            break;
        case '1': case '2': case '3': case '4': // Korrektur Apostroph
            printf("\nBitte Grad Celsius eingeben: ");
            scanf("%f", &temperatur);
            if (temperatur >= -273.15) {
                switch (zeichen) {
                    case '1': // Korrektur Apostroph
                        printf(
                            "\n%.2f Grad C sind %.2f Grad F",
                            temperatur, 9.0 / 5.0 * temperatur + 32.0);
                        break; // Korrektur
                    case '2': // Korrektur Apostroph
                        printf(
                            "\n%.2f Grad C sind %.2f Grad R",
                            temperatur, 4.0 / 5.0 * temperatur);
                        break; // Korrektur
                }
            }
    }
}

```

```

    case '3':
        printf(
            "\n%.2f Grad C sind %.2f K",
            temperatur, temperatur + 273.15);
        break;
    case '4':
        printf(
            "\n%.2f Grad C sind %.2f Grad Rank",
            temperatur, 9.0 / 5.0 * (temperatur + 273.15));
        break;
    }
}
else {
    printf("\nDiese Temperatur gibt es nicht!");
    return -1;
}
break;
default:
    printf("\nUngueltige Option");
    return -1;
}
return 0;
}

```

6.

- Schreiben Sie ein Programm, das alle ungeraden Zahlen von 1 bis n addiert. Verwenden Sie dazu eine `for`-Schleife.
- Ändern Sie das Programm aus a) so, dass statt der `for`-Schleife eine `while`-Schleife benutzt wird.

Lösung:

```

#include <stdio.h>
#define MAX_VAL 10 // erlaubte Werte: 1 .. << UINT_MAX

int main() {

    unsigned int i;
    unsigned n = MAX_VAL;
    unsigned sumFor = 0, sumWhile = 0;

    printf(
        "\nSumme aller ungeraden Zahlen von 1 bis und mit %d",
        n % 2 == 1 ? n : n - 1);
    for (i = 1; i <= n; i += 2) { // Loesung a)
        sumFor += i;
    }

    i = 1; // Loesung b)
    while (i < n) {
        sumWhile += i;
        i += 2;
    }

    // Test
    printf("\nSumme mit for-Schleife: %u", sumFor);
    printf("\nSumme mit while-Schleife: %u", sumWhile);

    return 0;
}

```

7.

Wie oft wird der Text "Hallo Welt!" ausgegeben?

```

...
for (x = 1; x < 6; x = x + 1) {
    for (y = 2; y <= 4; y = y + 1) {
        printf("\nHallo Welt!");
    }
}
...

```

Lösung: 5 x äussere, 3 x inner Schleife: 15 x insgesamt

8.

Das folgende Programm gibt eine Anzahl von Sternen (*) in Form eines Dreiecks auf dem Bildschirm aus. Ändern Sie das Programm so, dass dieses Dreieck auf dem Kopf steht.

#include <stdio.h>

```

int main() {
    int zeile, spalte;

    for (zeile = 1; zeile <= 20; zeile = zeile + 1) {
        for (spalte = 1; spalte <= zeile; spalte = spalte + 1) {
            printf("*");
        }
        printf("\n");
    }
    return 0;
}

```

```

// *
// **
// ***
// **** etc.

```

Lösung:

#include <stdio.h>

```

int main() {
    unsigned int zeile, spalte;

    for (zeile = 1; zeile <= 20; ++zeile) {
        for (spalte = 1; spalte <= 21 - zeile; ++spalte) {
            printf("*");
        }
        printf("\n");
    }
    return 0;
}

```

```

// ****
// ***
// **
// *

```

9.

An einer Champagner-Matinée stossen n Gäste paarweise miteinander an. Wieviele Male erklingen die Gläser?

```
#include <stdio.h>

int main() {
    unsigned int nofGuests, n;

    printf("Champagner-Klingklang-Rechner");
    printf("\nGeben Sie die Anzahl anstossender Gaeste ein: ");
    scanf("%u", &nofGuests);
    printf("Die Glaeser erklingen %u mal.", (n = nofGuests) * --n / 2);
    return 0;
}
```

10.

Notieren Sie die folgenden Aussagen in C-Schreibweise für eine `while`-Schleife.

- | | |
|--|-------------------------------------|
| | Lösungen: |
| a) Solange summe ungleich 1000. | <code>while (summe != 1000)</code> |
| b) Solange x gleich y. | <code>while (x == y)</code> |
| c) Solange zahl grösser oder gleich n. | <code>while (zahl >= n)</code> |
| d) Solange x minus y gleich 145. | <code>while (x - y == 145)</code> |
| e) Solange a mal b gleich c mal d. | <code>while (a * b == c * d)</code> |