

# Praktikum 3: Kontrollstrukturen und Container

## 1 Lernziele

- Definieren und Verwenden von Strukturen
- Anwenden von Kontrollstrukturen, insbesondere Schleifen
- Verwenden von ineinander verschachtelten Kontrollstrukturen
- Implementieren von textbasierten Menüs
- Verwenden von Vektoren

## 2 Aufgabenstellungen

### 2.1 Einfaches Histogramm

Schreiben Sie eine **Funktion**, die wahlweise ein `int`-Array (und dessen Länge als `int`-Wert) **oder** einen Vektor aus `int`-Werten als Parameter übergeben bekommt:

```
1 void histogramm(int a[], int laenge)
2 //oder
3 void histogramm(vector<int> a)
```

Die Funktion soll jeden der `int`-Werte als Anzahl Sternchen in einer Zeile pro Wert ausgeben. Die Werte 1, 2, 10, 3, 7 ergeben so beispielsweise die folgende Ausgabe:

```
1 *
2 **
3 *****
4 ***
5 *****
```

### 2.2 Primzahlen

Schreiben Sie ein Programm, welches vom Benutzer zwei Zahlen erfasst. Geben Sie **alle** Primzahlen aus, die zwischen diesen beiden Zahlen liegen.

**Hinweis:** Ihr Programm wird „ordentlicher“, wenn Sie eine Funktion schreiben, die eine übergebene Zahl darauf prüft, ob es sich um eine Primzahl handelt oder nicht.

### 2.3 Pythagoreische Tripel

In rechtwinkligen Dreiecken gilt stets der Satz des Pythagoras:  $c^2 = a^2 + b^2$ , sprich das Quadrat der längsten Seite entspricht der Summe der Quadrate der beiden übrigen Seiten. Schreiben Sie ein Programm, das alle diese Tripel findet und ausgibt. Dabei gilt:

- Berücksichtigen Sie nur **ganzzahlige** Seitenlängen
- Betrachten Sie keine Seitenlängen, die größer sind als 500
- Geben Sie die Gesamtanzahl aller gefundenen Tripel aus

**Hinweis:** Sie werden, je nach Implementierung, ineinander verschachtelte Schleifen benötigen.

**Zusatzaufgabe:** Stellen Sie sicher, dass Sie keine doppelten Tripel finden, also beispielsweise 5, 3, 4 und 5, 4, 3.

Beispielausgabe der ersten drei Tripel:

```
1 5^2 = 3^2 + 4^2
2 10^2 = 6^2 + 8^2
3 13^2 = 5^2 + 12^2
```

## 2.4 Durchgehende Aufgabe: Version 0.2

Definieren Sie ein `struct` oder eine `class` „Lied“, welches folgende Informationen über ein Musikstück speichern kann:

- Titel
- Interpret
- Erscheinungsjahr
- Länge
- Genre

Verwenden Sie für die Variable `Genre` einen `enum`-Typen, der beispielsweise folgende Werte annehmen kann:

- Pop
- Rock
- Klassik
- Hard Rock
- ... (was Ihnen sonst noch einfällt)

### 2.4.1 Speicherung

Benutzen Sie einen `vector` um verschiedene Lied-Objekte zu speichern. Dieser kann einfach in der `main()` Funktion angelegt werden, oder aber (ausnahmsweise) als globales Objekt.

### 2.4.2 Funktionen

Implementieren Sie die Funktionen zu den Menüpunkten 1, 2 und 6!