

# OPR Praktikum

## Übung: Kraftsport

(Variante mit Templates)

Thomas Mahr

13. April 2018

### 1 Lernziele

- Sie sollen ein Programm aus mehreren Klassen auf Basis vorgegebener natürlichsprachlicher Anforderungen und einer vorgegebenen Prüffolge entwerfen und implementieren können.
- Sie sollen parametrierbare Klassen (Template) einsetzen können.

### 2 Voraussetzungen

Kapitel *Parametrierbare Klassen und Funktionen* aus dem Vorlesungsteil C++ *Vertiefungen*.

### 3 Aufgabe

Sie sollen die Durchführung eines Kraftsport-Trainings simulieren und den Energieumsatz eines Sportlers ermitteln, wobei die gegen die Gravitation geleistete potentielle Energie  $E_{pot} = m \cdot g \cdot h$  und ein Wirkungsgrad von 25% betrachtet werden soll. Ein Wirkungsgrad von 25 % bedeutet, dass 75 % der vom Sportler aufgewandten Energie als Wärme verloren geht.

Obwohl aus physikalischer Sicht kein Gewicht, sondern die Masse  $m$  angehoben wird, wird im Folgenden die in der Anwendungsdomäne übliche Bezeichnung Gewicht anstelle Masse verwendet.

Für die Abnahme der Übung müssen alle Abnahmekriterien einer der beiden Aufgabenvarianten erfüllt sein:

- Vereinfachte Aufgabe 3.1
- Ausführliche Aufgabe 3.2

### 3.1 Vereinfachte Aufgabenvariante

Funktionale Anforderungen:

- Ein Sportler führt Übungen durch.
- Einer Übung ist ein Gerät zugeordnet.
- An einem Gerät kann man ein Gewicht einstellen.
- Es gibt ein Gerät: Bankdrücken.
- Während einer Übung bleibt das Gewicht gleich.
- Eine Übung wird in mehreren Sätzen mit jeweils mehreren Wiederholungen ausgeführt, wobei die Anzahl der Wiederholungen pro Satz gleich bleibt.
- Eine Wiederholung ist eine vollständige Bewegung am Gerät: einmal Heben und Senken der Hantel beim Bankdrücken.
- Beim Bankdrücken stemmt der Sportler das Gewicht um seine Armlänge in die Höhe.
- Bei jeder Wiederholung setzt der Sportler Energie um. Nehmen Sie dabei einen Wirkungsgrad der Muskeln von 25 % an; d.h. 75 % der vom Sportler aufgewandten Energie geht als Wärme verloren.
- Die während einer einzelnen Übung umgesetzte Energie kann in zwei Formaten dargestellt werden:
  - in Joule
  - in der Einheit „Tafeln Schokolade“, wobei für eine 100 g Tafel 2200 kJ angenommen werden soll

Anforderungen an die Umsetzung:

- Für die Übung muss es die C++ Klasse `Uebung` geben.
- Die Klasse `Uebung` muss parametrierbar sein (also eine Template-Klassen sein). Über den Parameter wird die Art und Weise der Energieformatierung bestimmt.

Ergänzend zu diesen Anforderungen sind eine Prüffolge `PrueffolgeSport` und zwei Anwendungsbeispiele in der `main()`-Funktion vorgegeben, der Sie nähere Hinweise zur Architektur und Implementierung entnehmen können.

Stellen Sie außerdem in `beispiel2()` die Anzahl der Wiederholungen und das Gewicht so ein, dass Clark Kent im Training eine Tafel Schokolade umsetzt.

Für die Abnahme dieser Aufgabe gilt:

1. Zur Überprüfung der funktionalen Anforderungen wird eine Klasse `PrueffolgeSport` zur Verfügung gestellt, die die vorgegebenen Prüfungen abarbeitet.
2. Sie dürfen die Dateien `main.cpp` (außer in der Funktion `beispiel2()`), `Prueffolge.h`, `Prueffolge.cpp`, `PrueffolgeSport.h` und `PrueffolgeSport.cpp` nicht ändern.
3. Für die Abnahme dieser Aufgabe müssen alle Prüfungen fehlerfrei durchlaufen werden.
4. Clark Kent muss eine Tafel Schokolade umsetzen.

### 3.2 Ausführliche Aufgabenvariante

Funktionale Anforderungen:

- Ein Sportler führt ein Training durch.
- Das Training besteht aus mehreren Übungen.
- Jeder Übung ist ein Gerät zugeordnet.
- An einem Gerät kann man ein Gewicht einstellen.
- Es gibt zwei Geräte: Bankdrücken und Klimmzug.
- Eine Übung besteht aus mehreren Sätzen.
- Ein Satz besteht aus mehreren Wiederholungen mit einem Gewicht, das für den Satz gleich bleibt.
- Die einzelnen Sätze können sich in Wiederholungszahl und Gewicht unterscheiden.
- Eine Wiederholung ist eine vollständige Bewegung am Gerät, z.B. ein Klimmzug oder ein Heben und Senken der Hantel beim Bankdrücken.
- Beim Bankdrücken stemmt der Sportler das Gewicht um seine Armlänge in die Höhe.
- Beim Klimmzug zieht der Sportler seinen Körper (und ein mögliches Zusatzgewicht) um seine Armlänge in die Höhe.
- Bei jeder Wiederholung setzt der Sportler Energie um. Nehmen Sie dabei einen Wirkungsgrad der Muskeln von 25 % an; d.h. 75 % der vom Sportler aufgewandten Energie geht als Wärme verloren.
- Sowohl die während des gesamten Trainings als auch während einer einzelnen Übung umgesetzte Energie kann in zwei Formaten dargestellt werden:
  - in Joule
  - in der Einheit „Tafeln Schokolade“, wobei für eine 100 g Tafel 2200 kJ angenommen werden soll

Anforderungen an die Umsetzung:

- Für das Training und die Übung muss es die C++ Klassen `Training` bzw. `Uebung` geben.
- Die Klassen `Training` bzw. `Uebung` müssen parametrierbar sein (also Template-Klassen sein). Über den Parameter wird die Art und Weise der Energieformatierung bestimmt.
- Für den Sportler muss es eine Klasse `Sportler` geben.
- Die Klasse `Sportler` muss unabhängig von den konkreten Geräten sein.

Ergänzend zu diesen Anforderungen sind eine Prüffolge `PrueffolgeSport` und zwei Anwendungsbeispiele in der `main()`-Funktion vorgegeben, der Sie nähere Hinweise zur Architektur und Implementierung entnehmen können.

Stellen Sie außerdem in `beispiel2()` die Anzahl der Wiederholungen und das Gewicht so ein, dass Clark Kent im Training eine Tafel Schokolade umsetzt.

Für die Abnahme dieser Aufgabe gilt:

1. Zur Überprüfung der funktionalen Anforderungen wird eine Klasse `PrueffolgeSport` zur Verfügung gestellt, die die vorgegebenen Prüfungen abarbeitet.

2. Sie dürfen die Dateien *main.cpp* (außer in der Funktion `beispiel2()`), *Prueffolge.h*, *Prueffolge.cpp*, *PrueffolgeSport.h* und *PrueffolgeSport.cpp* nicht ändern.
3. Für die Abnahme dieser Aufgabe müssen alle Prüfungen fehlerfrei durchlaufen werden.
4. Clark Kent muss eine Tafel Schokolade umsetzen.