

SWE3 HÜ 5: Vererbung

Erik Pitzer <erik.pitzer@fh-hagenberg.at>

Medizin- und Bio-Informatik – WS 2024/25

Name

Aufwand (in h)

Punkte

Aufgabe	Lösungsidee	Implementierung	Tests	Gesamtpunkte
1	20% / 5P	50% / 15P	30% / 10P	30
2	20% / 15P	50% / 35P	30% / 20P	70

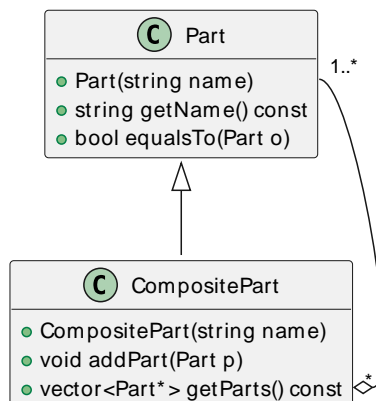
Erstellen Sie für die folgenden Aufgaben eine gemeinsame Solution und die jeweilige Aufgabe als Projekt in einem Unterverzeichnis.

Aufgabe 1: Flugreisen (./flugreise)

Sie werden vom einem Reisebüro beauftragt, die Mitarbeiter bei der Erledigung von Flugreisen softwaremäßig zu unterstützen: Für alle Personen werden im Reisebüro folgende Daten gehalten: Vor- und Familienname, Geschlecht, Alter, Adresse sowie Kreditkarten-Nummer. Für jene Flüge, die zu einer Flugreise gehören (z.B. Hinflug: Linz → Frankfurt → Denver → Las Vegas und Rückflug: Las Vegas → San Francisco → München → Linz), werden Flugnummer, Fluggesellschaft, Ort, Abflug- und Ankunftszeit sowie Flugdauer vermerkt. Modellieren Sie mindestens die Klassen **Flugreise**, **Flug** und **Person** und implementieren Sie diese als C++-Klassen: Es soll auch eine einfache Ausgabe von Flugreisen möglich sein (mittels `operator<<>`). Testen Sie ihre Implementierung ausführlich.

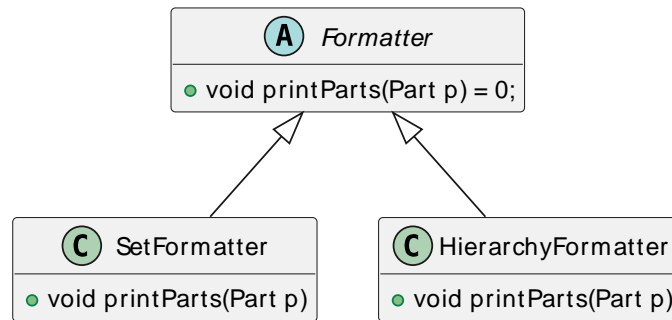
Aufgabe 2: Stücklistenverwaltung (./partlists)

1. Implementieren Sie die folgende Klassenhierarchie in einem namespace **PartLists**:



Die Klasse **Part** hat ein (privates) Attribut `name` mit der Zugriffsmethode `getName` sowie einen Konstruktor und eine Vergleichsmethode. Die Klasse **CompositePart** hat einen Konstruktor und die Operationen `addPart` zum Hinzufügen eines Teils. Die Funktion `getParts` liefert einen Vektor zurück, mit der man alle Teile, die im zusammengesetzten Teil enthalten sind, bearbeiten kann.

2. Implementieren Sie anschließend die folgenden Klassen:



Geben Sie die Stückliste einmal mit dem **SetFormatter** und einmal mit dem **HierarchyFormatter** aus, z.B.:

- **HierarchyFormatter:**

```
Sitzgarnitur
  Sessel
    Bein (klein)
    Bein (klein)
    Bein (klein)
    Bein (klein)
  Sitzfläche
  Sessel
    Bein (klein)
    Bein (klein)
    Bein (klein)
    Bein (klein)
  Sitzfläche
  Tisch
    Bein (groß)
    Bein (groß)
    Bein (groß)
    Bein (groß)
  Tischfläche
```

- **SetFormatter:**

```
Sitzgarnitur:
  8 Bein (klein)
  2 Sitzfläche
  4 Bein (groß)
  1 Tischfläche
```

3. Stücklisten sollen persistent gehalten werden können, d.h. man soll Stücklisten laden und speichern können. Entwerfen Sie hierfür ein Interface (also eine abstrakte Klasse ohne Implementierungen) **Storable** (mit den Methoden **store** und **load**) und implementieren Sie es an den notwendigen Stellen.

Anmerkungen: (1) Geben Sie für Ihre Problemlösungen auch Lösungsideen an. (2) Kommentieren Sie Ihre Algorithmen ausführlich. (3) Strukturieren Sie Ihre Programme sauber. (4) Geben Sie ausreichend Testfälle ab und prüfen Sie alle Eingabedaten auf ihre Gültigkeit.