Übung 05

Korrektur: Sebastian Pritz

Name: Lukas Nigl-Eder

Punkte: 81

Allgemeine Anmerkungen:

1 Tag verspätet -4

Beispiel 1: Flugreisen 26 von 30

Lösungsidee: 3/3

Anmerkung	**Abzug**
Passt! 😊	

Quellcode: 15/17

Anmerkung	**Abzug**
Gesamter Code in einer einzigen .h/.cpp, anstatt pro Klasse. Bezüglich der Implementierung der Tests: Neue Instanzen einer Klasse sollten mit dem „new“ Stichwort angelegt werden (d.h. „Person* p = new Person(...)“ anstatt „Person p(...)“) → sonst hört das Objekt nach der Funktion auf zu existieren und auch der Speicher kann nicht händisch freigegeben werden. Kann zu Problemen führen! Sonst aber alles okay.	**-2** **Hinweis**

Testfälle: 8/10

Anmerkung	**Abzug**
Keine Flugreise ohne Personen und/oder Flüge Sonst aber ok.	**-2**

Beispiel 2: Stücklistenverwaltung 59 von 70

Lösungsidee: 5/7

Anmerkung	Abzug
Storable nicht beschrieben	-2

Quellcode: 33/40

Anmerkung	Abzug
Warum alles in einem einzigen File? Nicht nur wurde nicht auf .h und .cpp aufgeteilt, sondern auch beide Klassen in ein File. Das wird bei Projekten mega unübersichtlich! Dass man .h und .cpp aufteilt, sowie verschiedene Module (zuvor statt Klassen) erstellt, ist bereits seit dem 1. Semester so, und das hat auch seinen Sinn.	**-2**
Storable nicht als Interface verwirklicht, sonderm einfach die Funktionen in Part/CompositePart übernommen. Außerdem, wie bereits erwähnt, load nicht funktionstüchtig bzw. auch store nicht ganz korrekt, weil die Hierarchie nicht enthalten ist. Im einfachsten Fall, könnte man mit Einrückungen arbeiten.	**-3**
Alle Testfälle (auch wenn es zusammengefasst nur eine Funktion ist) in die Main, und nicht wie sonst in ein dediziertes Testfile...	**-2**
Abseits davon aber korrekt implementiert und schön kommentiert!	

Testfälle: 21/23

Anmerkung	Abzug
Equals nicht getestet, sonst aber alles abgedeckt. 😊	**-2**