

Zwei beispielhafte Klausuraufgaben*

Erstellen Sie für die folgende Aufgabe eine Musterlösung

Aufgabe 4 (12 Punkte):

Geben Sie Syntaxdiagramme an, um die folgende Sprache zur Beschreibung von Klausuren zu definieren:

- Eine Klausur besitzt zu Beginn zwei (an anderer Stelle definierte) Titelblätter.
- Eine Klausur besteht danach aus beliebig vielen Aufgaben, mindestens jedoch aus einer Aufgabe.
- Eine Aufgabe ist durch das Schlüsselwort **Aufgabe** gekennzeichnet, gefolgt von der Anzahl der maximal zu erreichenden Punktzahl, gefolgt vom Schlüsselwort **Punkte**. Hierauf folgen beliebig viele Teilaufgaben, mindestens jedoch eine Teilaufgabe.
- Jede Teilaufgabe wird durch einen Aufzählungsstrich „–“ eingeleitet, auf den ein (an anderer Stelle definierter) Aufgabentext folgt.
- Die maximal zu erreichende Punktzahl ist eine ein- bzw. zweistellige Zahl zwischen 1 und 99 (einschliesslich), jeweils ohne führende Null.

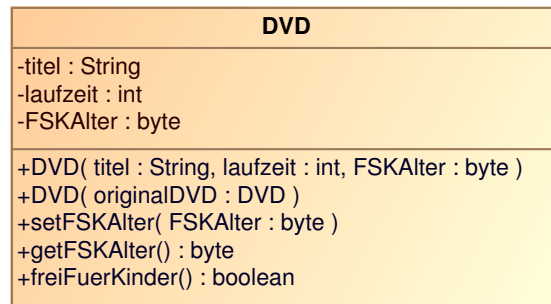
Sie dürfen annehmen, dass alle „an anderer Stelle definierten“ Nonterminalsymbole eindeutig erkennbar sind. Sie dürfen aus der Vorlesung bekannte Nonterminalsymbole verwenden, kennzeichnen Sie diese Nonterminalsymbole dann als „aus der Vorlesung bekannt“.

*Aus der Informatik I Klausur WS 2012/2013

Erstellen Sie für die folgende Aufgabe eine fehlerhafte Abgabe

Aufgabe 8 (8+8=16 Punkte):

Betrachten Sie zunächst das folgende UML-Diagramm, das die Modellierung eines als DVD verfügbaren Films darstellt:



1. Implementieren Sie die durch obiges Diagramm vorgegebene Klasse in **Java**. Es müssen nur die im Diagramm angegebenen Methoden umgesetzt werden.

Die Funktion `freiFuerKinder` soll genau dann `true` zurückliefern, wenn das bei der freiwilligen Selbstkontrolle angegebene Alter (`FSKAlter`) maximal den Wert 6 hat.

Um den Rahmen der Aufgabe nicht zu sprengen, dürfen Sie auf Kommentare verzichten.

2. Zu der gegebenen Klasse **DVD** soll eine Unterklasse **ActionDVD** modelliert und implementiert werden, ohne dass hierfür die Klasse **DVD** verändert werden soll.

Jedes Objekt der Klasse **ActionDVD** modelliert einen als DVD verfügbaren Action-Film, dessen `FSKAlter` immer den Wert 18 hat und der somit niemals für Kinder freigegeben sein kann.

- (a) Ergänzen Sie das UML-Diagramm um die Darstellung dieser Unterklasse. Achten Sie auf eine sinnvolle Signatur für beide Konstruktoren.
- (b) Implementieren Sie diese Unterklasse in **Java**. Auch hier dürfen Sie auf Kommentare verzichten.