

## Wiederholklausur Fortgeschrittene Programmierung / Algorithmen und Datenstrukturen (Java 2)

- Datum: 13.02.2019
- Hilfsmittel: Taschenrechner, Beiblatt
- Hinweise: Benötigte Schnittstellen- und Klassenimporte müssen von Ihnen nicht explizit angegeben werden.
- Bearbeitungszeit: 100 Minuten

Aufgabe	Maximale Punktzahl	Erreichte Punktzahl
1	32	
2	28	
3	22	
4	18	
Gesamt	100	

## Aufgabe 1

- a) Erläutern Sie den wesentlichen Unterschied zwischen einer lokalen und einer anonymen Klasse.
- b) Erläutern Sie, was man in der Programmierung unter dem Begriff *Upcast* versteht.
- c) Skizzieren und erläutern Sie die Ereignisbehandlung in Java und benennen Sie das skizzierte Modell.
- d) Erläutern Sie, was die Besonderheit der Schnittstelle *Serializable* darstellt.
- e) Erläutern Sie kurz den Nachteil bei generischer Programmierung ohne Java Generics und erläutern Sie kurz, was man unter Java Generics versteht.
- f) Erläutern Sie kurz, was man unter dem FIFO-Prinzip versteht und benennen Sie eine Klasse, die auf diesem Prinzip basiert.
- g) Skizzieren Sie den Aufbau einer *ArrayList* und einer *LinkedList* und erläutern Sie, bei welchen Operationen die jeweilige Liste Vorteile besitzt.
- h) Erläutern Sie kurz, was man unter dem Teile-und-Herrsche-Prinzip versteht und erläutern Sie kurz die Funktionsweise des Insertsorts.

## Aufgabe 2

Gegeben sei das beigefügte Klassendiagramm.

- Implementieren Sie die Klasse *VideoStorageMedia*.
- Implementieren Sie die Klasse *BluRay*.
- Implementieren Sie die Klasse *VideoCollection*.

### Hinweise zur Klasse *BluRay*

Die display-Methode soll die Kapazität, sowie den Filmtitel auf der Konsole ausgeben. Realisieren Sie die Konsolenausgabe wie abgebildet.

Kapazität: 100 Filmtitel: Avengers - Endgame
-------------------------------------------------

### Hinweise zur Klasse *VideoCollection*

- Die addVideo-Methode soll der Videoliste (Attribut *videos*) den aufsteigenden Index, sowie das eingehende Video hinzufügen.
- Die loadVideoCollection-Methode soll mit Hilfe der addVideo-Methode alle Videos der eingehenden Datei der Videoliste hinzufügen. Die Videos wurden mit Hilfe der Standardserialisierung in der Datei gespeichert. Fangen Sie die Ausnahmen *IOException* und *ClassNotFoundException* ab und lösen Sie als Fehlerbehandlung die Ausnahme *VideoLoadException* aus.
- Die getIndexByMovieDescription-Methode soll den Index des Videos zur eingehenden Film-Beschreibung (Attribut *title* der Klasse *Movie*) zurückgeben.

## Aufgabe 3

Gegeben sei das beigefügte Klassendiagramm, sowie der beigefügte Entwurf der grafischen Benutzeroberfläche.

- a) Implementieren Sie die Methode *actionPerformed(ActionEvent)* der Klasse *AddMovieFrame*.
- Überprüfen Sie, ob die Drucktaste *saveButton* oder die Drucktaste *cancelButton* betätigt wurde.
  - Option A: Drucktaste *saveButton* wurde betätigt:
    - Erzeugen Sie zum eingegebenen Titel, zum eingegebenen Genre und zum eingegebenen Erscheinungsjahr einen Film (Klasse *Movie*).
    - Erzeugen Sie abhängig vom eingegebenen Medium entweder einen Eingabedialog für die Kapazität (Attribut *capacity* der Klasse *BluRay*) oder einen Bestätigungsdialog für die Überschreibbarkeit (Attribut *isRewriteable* der Klasse *VHS*).
    - Erzeugen Sie zum erzeugten Film und zur eingegebenen Kapazität bzw. zur eingegebene Überschreibbarkeit eine BluRay bzw. eine VHS und fügen Sie diese der Filmliste (Attribut *collection*) hinzu.
  - Option B: Drucktaste *cancelButton* wurde betätigt:
    - Rufen Sie die statische Methode *exit(int)* der Klasse *System* mit dem Wert 0 auf.
- a) Implementieren Sie die Methode *setFrame()* der Klasse *AddMovieFrame* so, dass der Inhaltsbereich des Fensters (Attribut *contentPane*) auf den Wert *panel*, der Titel auf den Wert *Film hinzufügen* und die Größe auf den Wert 500x250 gesetzt werden, dass das Fenster beim Erzeugen zentral positioniert wird und dass beim Schließen des Fensters das dazugehörige Programm ebenfalls beenden wird.

### Hinweise

- Die statische Methode *values()* einer Aufzählung gibt alle Aufzählungskonstanten der Aufzählung als Feld zurück.
- Die statische Methode *showInputDialog(String)* der Klasse *JOptionPane* erzeugt zum eingehenden Text einen Eingabedialog.
- Die statische Methode *showConfirmationDialog(Object, String)* der Klasse *JOptionPane* erzeugt zur eingehenden Komponente und zum eingehenden Text einen Bestätigungsdialog.

## Aufgabe 4

Gegeben sei die abgebildete Liste, sowie die beigefügte Implementierung des Quicksort-Verfahrens. Wenden Sie das Quicksort-Verfahren auf die Liste an. **Schreiben** Sie hierzu für jeden Durchlauf die jeweilige Reihenfolge der Zahlen auf, **kreisen** Sie den jeweiligen Teiler ein und **markieren** sie die entstandenen neuen Bereiche.

Index	0	1	2	3	4	5	6	7	8
0	7								
1	1								
2	4								
3	5								
4	8								
5	0								
6	2								
7	6								
8	3								

Durchgang	l	r	m	i	j	l/j	i/r
1							
2							
3							
4							
5							
6							
7							
8							