

## Fragen zur Vorlesung „Informatik B“ Teil 1

### **Objektorientierte Programmierung Teil 1**

(Klassen und Objekte, Operationen, Überladen von Operationen)

- Erklären Sie die Begriffe Klasse, Objekt, Attribut und Operation
- Wie ist der allgemeine Aufbau einer Klasse in Java?
- Wie wird ein Objekt einer Klasse erzeugt?
- Wie erfolgt der Zugriff auf Attribute?
- Wie erfolgt der Aufruf von Operationen?
- Wofür steht der this-Zeiger?
- Wie ist die Sichtbarkeit zwischen lokalen Variablen und Attributen geregelt?
- Was versteht man unter dem Überladen von Operationen?
- Welche Regeln gelten für das Überladen von Operationen in Java?
- Was ist die Signatur einer Operation?

### **Objektorientierte Programmierung Teil 2**

(Konstruktoren, Destruktoren)

- Was ist ein Konstruktor?
- Was versteht man unter der Verkettung von Konstruktoren?
- Was ist ein Destruktor?
- Welche Besonderheiten gelten für die Anwendung von Destruktoren in Java?

(Klassen mit static-Elementen)

- Was ist ein Klassenattribut?
- Welche Eigenschaften besitzt ein Klassenattribut?
- Wie lässt sich eine Konstante definieren?
- Was ist eine Klassenoperation?
- Welche Regeln gelten für Klassenoperationen?
- Wozu werden Klassenoperationen eingesetzt?
- Was ist ein statischer Konstruktor?

### **Objektorientierte Programmierung Teil 3**

(Vererbung)

- Was versteht man unter Vererbung?
- Was ist der Unterschied zwischen einfacher Vererbung und Mehrfachvererbung?
- Wie wird in Java eine Klasse abgeleitet?
- Wie kann in Java verhindert werden, dass aus einer Klasse weitere Klassen abgeleitet werden?
- Was versteht man unter der Überlagerung von Operationen?
- Was versteht man unter dynamischem Binden?
- Welche Möglichkeiten existieren in Java, um zu verhindern, dass eine Operation überlagert wird?
- Wie kann in Java eine verdeckte Operation der Oberklasse aufgerufen werden?
- Welche Regeln gelten für die Verkettung von Konstruktoren in Java?
- Wie sieht der Default-Konstruktor aus?

## **Objektorientierte Programmierung Teil 4**

(Modifikatoren, Kapselung, Abstrakte Klassen und Operationen)

- Welche Sichtbarkeitsebenen für Elemente einer Klasse existieren in Java?
- Was versteht man unter Kapselung?
- Wie sind get- und set-Operationen aufgebaut?
- Was ist eine abstrakte Operation?
- Was ist eine abstrakte Klasse?
- Welche Eigenschaften hat eine abstrakte Klasse?
- Wozu dienen abstrakte Klassen?
- Was bedeutet Polymorphie?

(Interfaces)

- Was ist ein Interface in Java?
- Was ist zu tun, damit eine Klasse ein Interface implementiert?
- Welche wichtigen gemeinsamen Eigenschaften besitzen Klassen und Interfaces?
- Welche der folgenden Elemente Attribute, Konstanten, Operationen, abstrakte Operationen dürfen in Klassen, abstrakten Klassen und Interfaces enthalten sein?
- Erläutern Sie den Begriff Generizität.

## **Assoziationen**

- Was gibt die Kardinalität einer Beziehung an?
- Wie verwaltet man Assoziationen in Java?

## **Pakete**

- In welchen organisatorischen Struktur sind die Klassen in Java unterteilt?
- Welche beiden Möglichkeiten gibt es, Klassen aus verschiedenen Pakete zu benutzen?
- Was bedeutet „import on demand“?
- Mit welcher Anweisung kann man eigene Pakete erstellen und wie lautet deren Syntax?
- Was versteht man unter dem Default-Paket?

## **Fehlerbehandlung in Java**

(Ausnahmen, Fehlerklassen)

- Erläutern Sie das Grundprinzip des Ausnahme-Mechanismus in Java
- Wie kann eine Ausnahme im Programm behandelt werden?
- Welche wichtigen Operationen besitzen Fehlerobjekte?
- Beschreiben Sie die grobe Einteilung von Fehlern in Java
- Was besagt die Catch-or-throw-Regel?
- Wie funktioniert die Weitergabe von Ausnahmen?
- Wie können Ausnahmen selbst ausgelöst werden?