

Aufgabe 1: Allgemeines Verständnis (25 Punkte)

a) Nennen Sie 5 Grundbausteine einer Programmiersprache: (5 Punkte)

- Syntax (gr.: Form, Lehre vom Satzbau)
- Semantik (Bedeutung)
- Operatoren (z. B.: arithmetische Operatoren: {+, -, *, /,...}, boolesche Operatoren, Bit-Operatoren etc.)
- Kontrollstrukturen: jeder Algor. Ist grundsätzlich mit den 3 Bausteinen der strukturierten Programmierung abbildbar; das sind die Sequenz, die Iteration und die Auswahl. Weitere Kontroll(flow)strukturen sind die Recursion, der Block, die Funktion, die Procedure, die aber nicht in allen Programmiersprachen unterstützt werden.
- Datentypen: a.) fundamentale (z.B.: Integer, Float, Char etc.) und b.) Abstrakte (user defined) (z. B.: struct in C, oder Record in Pascal).

b) Die fundamentalen Datentypen in Java haben (2 Punkte)

- eine fixe Größe: ja/nein Antwort = „Ja“
- einen vordefinierten Defaultwert: ja /nein Antwort= „Ja“

c) Wie in der Vorlesung gehört, ist Java stark an C/C++ angelehnt, weist aber einige bemerkenswerte Unterschiede auf. Nennen Sie 5 Unterschiede:

(5 Punkte)

- Java kennt keine Pointer
- Java keine Mehrfachvererbung (multiple inheritance): eine Unterklasse kann nur aus genau 1 Oberklasse Merkmale erben.
- Java wird zu einem (maschinenunabhängigen) Bytecode kompiliert, der dann durch eine (maschinenabhängige) Virtuelle Maschine (beim Durchführen) abgearbeitet wird.
- C/C++ -Programme (im speziellen Sourcen) werden zu einem maschinenabhängigen Code (durch den Compiler) umgewandelt und können nur auf gleichartigen Maschinen laufen, auf den sie übersetzt worden sind.
- Java ist grundsätzlich Objekt-Orientiert
- in C sind fundamentale Datentypen in ihrer Größe maschinenabhängig/compilerabhängig, nicht so in Java.

d) Was sind die wesentlichen Merkmale (Besonderheiten) von Java?
Nenne Sie 4 Merkmale.

(4 Punkte)

- Plattformunabhängige bzw. Maschinenunabhängige Programme
- zur Gänze OO
- keine Pointer
- keine Mehrfachvererbung
- ad Sicherheitsaspekt: Zugriffe auf Systemressourcen erfolgen über eine standardisierte Schnittstelle der Virtuellen Maschine; dies erhöht die Sicherheit.
- Applikationen können aus vielen kleineren Modulen bestehen, die erst bei Bedarf nachgeladen werden (z. B: Textverarbeitungsapplikation über Internet: erst wenn die Formatierungsfunktion vom User angestoßen wird, wird das dazugehörige Modul nachgeladen. → smart code
- unterschiedliche Arten von durchführbaren Java-Code:
Applets: ermöglichen interaktives Arbeiten am Client ohne Kanalbelastung zum Server; sie bedienen sich meist der Clientressourcen, sind nicht selbstständig lauffähig und sind meist in HTML-Dokumenten „eingebettet“. Typischerweise Einsatz zur Datenerfassung bzw. Aufbereitung direkt am Client (Dialoge direkt am Client → keine Kanal/Leitungsbelastung während einer Transaktion).

e) Beschreiben Sie die folgenden Konzepte und Begriffe der objektorientierten Programmierung (OOP): (9 Punkte)

- 1) **Objekte**
Ein Objekt ist die reale Ausprägung der Klassenbeschreibung. Ein Objekt ist ein Gegenstand, von der Realität übernommen und durch Abstraktion zu einem wichtigen Kern reduziert. Es hat eine Eigenschaft, wohl definiertes Verhalten und ist eindeutig bestimmbar (Name).
- 2) **Klassen**
beschreibt Eigenschaften und Fähigkeiten von ähnlichen Objekten
- 3) **Instanzen**
sind konkrete Ausprägungen von Klassen
- 4) **Methoden und Messages**
Methoden sind die Fähigkeiten von Objekten. Messages sind zum Beispiel Aufrufe von Methoden (durch Schicken einer Message an die Methode).
- 5) **Vererbung**
Man übernimmt Methoden und Eigenschaften einer Klasse und fügt neue Methoden und Eigenschaften hinzu. (Oberklasse→Unterklasse).
- 6) **Überladen (overloading)**
Im Unterschied zur Vererbung, wo nur **neue** Eigenschaften hinzugefügt werden, muß man durch overloading bestehende Methoden überschreiben.
- 7) **Polymorphismus (Vielgestaltigkeit)**
Objekte reagieren aufgrund ihrer spezifischen Merkmalsausprägungen verschieden auf Messages:
Beispiel:
Objekt Nokia und Objekt Ericsson: bei Message „Beleuchte Display“ reagiert Nokia mit „Grün“ und Ericsson mit „weiss“.
Oder klassischer Objekt Katze und Objekt Hund: bei Message „gib Laut“ reagiert Katze mit „Miau“ und Hund mit „Wau“.
- 8) **Kapselung**
Prinzip: Es gibt public Daten und Private Daten. Der Zugriff auf Private Daten erfolgt durch eigene Methoden (Datenkapseln) bzw. deren stand. Schnittstellen.
Beispiel: Datenkapsel „KUNDEN“: über eine stand. Schnittstelle gibt man der Datenkapsel die Message „ich will den Namen des Kunden 4711“ und erhält Max Mustermann zurück. Woher (und wie) die Datenkapsel den Namen des Kunden 4711 hat weiß man nicht.→ Information Hiding
- 9) **Geheimnisprinzip (Information Hiding)**
eng verbundener Begriff bei Kapselung: Man weiß nicht wirklich was vor Sich geht.
Die bewusste Herbeiführung eines solchen Zustandes (siehe auch Kapselung oben), nämlich das Verbergen unwichtiger Details, ist als Information Hiding bekannt geworden und spielt beim OO-Entwurf von Softwaresystemen eine zentrale Rolle.

Aufgabe 2: Fundamentale Datentypen (20 Punkte)

Definieren Sie die folgenden Angaben in Java. Geben Sie die Definitionen sowie die jeweiligen Werte aus

- a) Eine Gleitkommavariablen, die mit negativ unendlich initialisiert wird.

```
double a;  
a = DOUBLE.NEGATIVE_INFINITY;
```
- b) Eine ganzzahlige Variable, die mit 1 initialisiert wird. Diese Variable soll durch bitweise Linksverschiebung zu ihrem Minimalwert geändert werden.

```
int b=1;  
b << 31;
```
- c) Eine ganzzahlige Variable, die mit -1 initialisiert wird und durch bitweise Rechtsverschiebung ohne Vorzeichenverschiebung (Bitmanipulation: >>>) zu ihrem Maximalwert geändert wird.

```
int c = -1;  
c >>> 1;
```
- d) Eine 2 x 3 Matrix mit doppelt genauen Gleitkommazahlen, die mit den Werten 1 bis 6 initialisiert wird.

```
double mat [ ] [ ] = {{1,2,3},{4,5,6}};
```