Aufgabe 1.1

erweitern: Gibt es in diesem Kontext nicht für Methoden. Gemeint könnte sein, dass eine Methode aus einer abgeleiteten Klasse per super() aufgerufen werden kann.

überschreiben: In einer Kindklasse kann die Funktionalität einer ererbten Methoden verändert werden, ohne, dass die Parameter oder der Name verändert werden. Dies wird üblicherweise mit @Override gekennzeichnet.

überladen: Eine Methode kann ihre Funktionalität in Abhängigkeit der übergebenen Parameter verändern. Dem gleichen Prinzip nach ist der Operator „+“ überladen, da er für Integer und Strings eine andere Funktionalität aufweist. So können beispielsweise verschiedene Konstruktoren angelegt werden (siehe Kopierkonstruktor etc.).

Aufgabe 1.2

Klassifikation: Dunno

Abstraktion: Dunno

Spezialisierung: Dunno

Vererbung: Eine Klasse kann beispielsweise von einer anderen Klasse deren Attribute und Methoden übernehmen. Ebenso ist es möglich von Interfaces Methoden/Attribute zu übernehmen. Eine Klasse kann immer nur von einer anderen Klasse erben, jedoch von mehreren Interfaces (diese werden dem Stichwort nach auch nur „implementiert“).

Abstrakte Klasse: Eine Abstrakte Klasse wird per „abstract“ gekennzeichnet und enthält abstrakte Methoden und/oder Attribute. Diese sind i.d.R. nicht vollständig implementiert, sondern lediglich als Header (Name, Typ und Parameter) dargestellt. Methoden können aber auch vollständig abgebildet sein und werden dann in dieser Form vererbt.

Interface: Ähnlich einer abstrakten Klasse enthält ein Interface lediglich Methoden-Header und bildet so eine Grundlage an mindestens zu implementierenden Methoden. Eine Klasse kann verschiedene Interfaces implementieren.