

# Programmierkurs Java

Dr. Dietrich Boles

## Aufgaben zu UE31-Polymorphie (Stand 05.02.2010)

### Aufgabe 1:

Implementieren Sie mit Hilfe der Klasse `java.util.ArrayList` die folgende Klasse `Stack`:

```
/**
 * A Last-In-First-Out(LIFO) stack of objects.
 */
class Stack extends java.util.ArrayList {
    /**
     * Pushes an item onto the stack.
     *
     * @param item
     *         the item to be pushed on.
     */
    public Object push(Object item);

    /**
     * Pops an item off the stack.
     */
    public Object pop();

    /**
     * Peeks at the top of the stack.
     */
    public Object peek();

    /**
     * Returns true if the stack is empty.
     */
    public boolean empty();

    /**
     * Sees if an object is on the stack.
     *
     * @param o
     *         the desired object
     * @return the distance from the top, or -1 if it is not found.
     */
    public int search(Object o);
}
```

Schreiben Sie dann ein Programm, bei dem zunächst ein `Stack`-Objekt erzeugt wird. Anschließend wird der Benutzer aufgefordert, int-Werte einzugeben. So lange die Werte größer als 0 sind, werden sie jeweils als Integer-Objekte auf den Stack gepusht. Nach Abschluss der Eingaben sollen alle Werte vom Stack gepoppt und auf den Bildschirm ausgegeben werden.

### Aufgabe 2:

Schauen Sie sich das Protokoll der Klasse `java.util.ArrayList` an. Schreiben Sie dann ein Programm, in dem ein `ArrayList`-Objekt erzeugt und mehrere Objekte unterschiedlicher Klassen darin gespeichert werden.

### Aufgabe 3:

Implementieren Sie eine Klasse `ArrayList` in Anlehnung an die entsprechende Klasse in UE25. Ein Objekt der Klasse `ArrayList` soll Objekte der folgende Klasse (sowie Objekte von Unterklassen dieser Klasse) speichern können:

```
public class Funktion {  
    public void printFunktionswert(int x) {  
        System.out.println(x);  
    }  
}
```

Sehen Sie in der Klasse `ArrayList` eine Methode `ausgeben` vor, die für alle gespeicherten Objekte deren Methode `printFunktionswert` aufruft.

Implementieren Sie Unterklassen von `Funktion` und schreiben Sie ein Testprogramm für die Klasse `ArrayList`.

### Aufgabe 4:

Implementieren Sie eine eigene Klasse `ArrayList` in Anlehnung an die entsprechende Klasse in UE25. Ein Objekt der Klasse `ArrayList` soll Objekte der Klasse `java.awt.Component` (sowie Objekte von Unterklassen dieser Klasse, also Buttons, Labels, ... (siehe UE22)) speichern können.

Sehen Sie folgende Methoden in der Klasse `ArrayList` vor:

- Eine Methode zum Hinzufügen eines neuen `Component`-Objektes (`add`)
- Eine Methode zum Entfernen eines `Component`-Objektes an einem bestimmten Index (`remove`)
- Eine Methode zum Liefern eines `Component`-Objektes an einem bestimmten Index (`get`)
- Eine Methode zum Liefern der Anzahl an aktuell in der `ArrayList` enthaltenen `Component`-Objekte (`size`)
- Eine Methode, in der ein neues Fenster erzeugt und auf dem Bildschirm angezeigt wird, in dem alle `Component`-Objekte, die sich aktuell in der `ArrayList` befinden, in einer einzelnen Spalte angeordnet dargestellt werden (`createWindow`).

Schreiben Sie ein Testprogramm für Ihre Klasse `ArrayList`.