


<b>ARBEITSAUFTRAG</b>	
<b>Fach: SEW2</b>	
<b>Thema: JAVA – String, Zufallsgenerator, Sortieren, Suchen</b>	

**Aufgabe 1: Erweitern Sie das Projekt Sort\_Random, wie folgt:**

- Kommentieren Sie inline den bereits vorhandenen Sourcecode
- Erweitern Sie Sort\_RandomTest dahingehend, dass ein Array randomStrings[] mit 100 zufälligen Strings erzeugt wird.
- Erweitern Sie Sort\_RandomTest dahingehend, dass ein Array randomInts[] mit 100 zufälligen Integerwerten erzeugt wird.
- Eine Kopie von randomStrings[] soll an Methoden der Klassen SortStrings bzw. eine Kopie von randomInts[] soll an Methoden der Klassen SortInts übergeben werden und dann jeweils mittels SelectionSort, InsertionSort, BubbleSort und Quicksort sortiert werden.
- Ferner binden Sie java.util.Arrays ein. Schauen Sie sich die darin verfügbaren Methoden an und schreiben Sie eine Anwendung für die dort zur Verfügung gestellten Methoden sort().
- Erweitern Sie Sort\_RandomTest dahingehend, dass alle Sortiermöglichkeiten durch den Benutzer auswählbar sind. Dabei soll jeweils nach Aufruf die ursprünglichen Werte und die sortierten Werte ausgegeben werden.

**Aufgabe 2: Erstellen Sie ein Projekt Sort\_Search\_Random als Erweiterung des fertigen Projekts Sort\_Random aus Aufgabe 1 und ergänzen Sie:**

- Fügen Sie eine Klasse SearchStrings und eine Klasse SearchInts hinzu
- In SearchStrings soll es eine Methode searchSeq() für die sequentielle Suche nach einem Zeichen geben, eine Methode searchBin() für die binäre Suche nach einem Zeichen geben.
- In SearchInts soll es eine Methode searchSeq() für die sequentielle Suche nach einem Integerwert geben, eine Methode searchBin() für die binäre Suche nach einem Integerwert geben.
- Ferner soll es zumindest eine Methode geben, die eine Methode aufruft, die von java.util.Arrays zur Verfügung gestellt wird.
- Das Benutzermenü ist hier entsprechend zu erweitern.