Advanced Programming

Sorting

In this project we will

- design a (simple) class to model sorted arrays;
- learn about arrays and loops;
- implement some classical sorting algorithms;

Arrays

An array in Java is similar to an n-tupel in mathematics: it is a combination of a fixed number of objects of equal type, where the individual objects can be accessed via an index.

An array variable in Java is declared, for example, by one of the following definitions:

```
int[] integerArray;
double[] doubleArray;
```

Array objects may be created by statements like

```
integerArray = new int[100];
int size = someInput() ;
doubleArray = new double[size];
```

Array elements can be accessed in the following way:

```
integerArray[50] = 123;
double tmp = Math.sqrt(doubleArray[0] + doubleArray[doubleArray.length-1]);
```

Each array has a field `length` that contains the number of elements that are stored in the array. The indices of an array are given by the set $\{0, 1, 2, \ldots, \texttt{length} - 1\}$. Thus, in the previous example the variable `tmp` will contain the square root of the sum of the first and the last element of array `doubleArray`.

Sorted arrays of integers

Keeping an array sorted gives lots of advantages, when a specific object needs to be found within an array. Therefore, set up a class `SortedArray` that models an array that is always sorted. It should contain an integer array as private field, and a method `sort` which sorts the array by implementing *Insertion Sort*[1].

Start with a class definition that can only store integer (type `int`). Provide a constructor that generates a sorted array of random integers, and implement methods to read and modify elements. Keep the array sorted after each modification.

More Sorting Algorithms

Rewrite the class `SortedArray` so that the method `sort` is abstract. Write a class `SortedArrayInsertion` which inherits from `SortedArray`. Furthermore, provide three other subclasses which implement:

- *Bubble sort*[1];
- *Merge sort*;
- *Quicksort*.

Provide a UML class diagram which shows all involved classes and their relations.

Optional: Sorted arrays of comparable objects

The use of a sorted array that can only store integers is rather limited. What we would like to have is a class for sorted arrays that can store any type of objects that can be sorted. However, to sort objects, we need at least a method to compare objects. Java provides an interface for such objects: `Comparable`. Re-implement your class `SortedArray` such that it can store objects of type `Comparable`.

[1] Have a look at the lecture slides of *Fundamental Algorithms* or a standard textbook on algorithms