

## Homework 3

### 1. Shuffling cards

In lecture 4-2 we presented a shuffle algorithm. We applied the algorithm to a deck of cards, implemented as an array of `String` values. The algorithm performed "in place" shuffling, meaning that the original array is mutated. As a result, following a shuffle, there is no way to go back to, or reconstruct, the original array.

Implement a version of the shuffle algorithm that creates a new array which is the shuffled version of the original array. In order to do so, modify the code of the supplied `Deck` class. In particular, the `shuffle` function should create, and return, a new array representing a random shuffle of the input deck. The original array must be kept unchanged.

Running the modified `Deck` program should print the original deck, then the shuffled deck, and finally the original deck again. For example:

```
♠2 ♦2 ♥2 ♣2 ♠3 ♦3 ♥3 ♣3 ♠4 ...
♠3 ♠2 ♦3 ♠4 ♥2 ♣2 ♣3 ♦2 ♥3 ...
♠2 ♦2 ♥2 ♣2 ♠3 ♦3 ♥3 ♣3 ♠4 ...
```

This is a relatively simple exercise, since most of the necessary code is given. However, it provides a good practice for working on the remaining exercises in this homework.

### 2. Array operations

Given an array of `int` values and an integer  $m$ , we wish to go through all the subarrays of size  $m$ , and find the subarray that has the largest sum of array values. This operation is featured, among other operations, in the supplied `ArrayOps` class.

Complete the supplied `ArrayOps` class by implementing the three functions whose code is missing. If you'll complete your work successfully, the last test function should print "success: true" instead of "success: false".

### 3. String operations

Complete the given `StringOps` class. You have to write five functions. We recommend implementing and testing them in the order in which they appear in the class.

### Submission

Before submitting your solution, inspect your code and make sure that it is written according to our **Java Coding Style Guidelines**. Also, make sure that each program starts with the program header described in the **Homework Submission Guidelines**. Both documents can be found in the Moodle site, it is your responsibility to read them. Any deviations from these guidelines will result in points penalty. Submit the following five files only:

- `Deck.java`
- `ArrayOps.java`
- `StringOps.java`

**Deadline:** Submit Homework 3 no later than December 2, 23:55. You are welcome to submit earlier.