



J. Hoenicke  
A. Nutz

21.6.2017

Please hand in your solution until  
28.6.2017, via email to  
nutz@informatik.uni-freiburg.de.

## Tutorials for “Formal methods for Java” Exercise sheet 8

### Exercise 1: Universe Type System

On the lecture’s webpage you find an implementation of a heap (`Heap1.java`). The heap is realized by an array, and every element in the heap stores its position in the heap.

A method to **enqueue** new elements into the heap, and a method to **remove** an element from the heap are provided. **enqueue** is called with object of type `HeapElem` as parameter that is created by the caller. The `User` class illustrates a possible use case. The *Universe Type System* is used to guard the invariant that every element in the heap knows its current position. Unfortunately, this does not work. Explain why the *Universe Type System* cannot be used here.

### Exercise 2: Pack/Unpack

On the lecture’s webpage you find a second file containing the same heap data structure from exercise 1. That file is already annotated with some slightly different invariants (`Heap2.java`, it also comes with a main method you can try out the data structure with).

Use the *pack/unpack* mechanism to guard the invariant. Change the implementation and verify it with `ESC/Java 2`. Your solutions are allowed to contain warnings for possible exceptions of the type (`PossiblyBadArrayAssignment`). Also note that the class contains some “unchecked or unsafe” operations that Java and `OpenJML(esc)` complains about.