Formal Methods in Software Development
Exercise 3 (May 31)

Wolfgang Schreiner
Wolfgang.Schreiner@risc.uni-linz.ac.at

May 1, 2007

The result is to me submitted to me by **May 31** (hard deadline) as an email that includes as attachments

- the JML-annotated Java files of the exercise,
- the output of `escjava2` on these files,
- the KeY proofs of the behavior specifications (saved as `.proof` files).

1 Maximum Search (KeY Verification, 5 Points)

Write a Java class `Search` with a Java method

```
  static int searchMax(int[] a)
```

that returns the maximum element of a non-empty array a (see Exercise 1). Specify the method with an appropriate JML header, and annotate the loop in its body with a suitable invariant (`loop_invariant`) and termination term (`decreases`). Check the class with `jml` and `escjava2`.

Now also add a frame condition to the loop (`assignable`, not understood by `jml/escjava2`) and verify the method's total correctness (proof obligation for normal behavior only) with KeY.

2 Inserting an Element (Verification, 5 Points)

Proceed as above with the method `insert` of class `Arrays` from Exercise 2.

As shown in class, please explicitly add (in both exercises) a condition on `a.length` to the method precondition respectively loop invariant.

If some proofs should not be completely successful, minimize the number of open goals as far as possible and give your ideas whey they cannot be proved.