Dr. Matthias Heizmann
Tanja Schindler
Dominik Klumpp

Hand in until 16:15 on June 5, 2019
Discussion: June 5, 2019

Tutorial for Program Verification
Exercise Sheet 11

Exercise 1: Sortedness 1 Point

On Exercise Sheet 4 we used an 1-ary function symbol ar to represent an array. Your task was to find the formula

$$\forall i, j \,.\, i \leq j \rightarrow ar(i) \leq ar(j)$$

which states that the array is sorted in ascending order. If we assume that domain and values of the array are integers and we use the theory of arrays and SMT-LIB syntax the formula is written as follows.

```
(forall ((i Int) (j Int)) (=> (<= i j) (<= (select ar i) (select ar j))))
```

(a) Give analogously an SMT-LIB formula `fsort` that states that the array ar is sorted between two integer indices lo and hi (inclusive).

(b) Give additionally an SMT-LIB formula that you can use to test your result. E.g., state a formula `ftest` such that the result for the `check-sat` command in the following SMT script is `unsat` but becomes `sat` if the line with `fsort` is deleted.

```
1 (set-logic ALIA)
2 (declare-fun ar () (Array Int Int))
3 (declare-fun lo () Int)
4 (declare-fun hi () Int)
5 (assert fsort)
6 (assert ftest)
7 (check-sat)
```

You can use the web interface of the Z3 SMT solver [1] to check your SMT script.

Exercise 2: Sorting Algorithm 1 Point

Implement a procedure in Boogie[2] that sorts an array. The signature of the procedure should be `sort(lo : int, hi : int, a : [int]int) returns (ar : [int]int)`. The values of the resulting array `ar` between the indices `lo` and `hi` (inclusive) should be sorted in ascending order. Each value that occurs n times in `a` between indices `lo` and `hi` should occur n times in `ar` between indices `lo` and `hi`.
You can use the Boogie interpreter Boogaloo[3] to test your program.

Please submit your SMT script and your Boogie program electronically (via Email)!

[1] `https://rise4fun.com/Z3/`
[2] `https://www.microsoft.com/en-us/research/wp-content/uploads/2016/12/krml178.pdf`
[3] `https://comcom.csail.mit.edu/comcom/#Boogaloo`