H. Gast, L. Noschinski

SHEET 8

Separation Logic

Date: 18.12.2012
Hand-In: 25.12.2012, 23:50

Goals You know how to use the "spatial formulae" of separation logic for specifications.

Exercise 1 [3] Swap Values on the Heap

Analogue to the examples of the lecture write the specification and proof for the following program, which swaps the contents of the memory cells pointed to by `p` and `q` (two `int` pointers).

```
t = *p;
*p = *q;
*q = t;
```

Syntax hints:

- The syntax for the points-to predicate is `p ↦ t. X`, where `p` is the pointer, `X` the value `p` points to, and `t` describes the type of `X`. For `X` of type *int*, *t* is `cty_int`.
- For proofs about separation logic, there are a number special proof methods:
  - `prep_vc`: Prepares the verification condition by extracting all heap modification into let-expressions (`heapop`). Is usually the first step after the VCG.
  - `step`: Evaluates one heap as prepared by `prep_vc`.
  - `run`: Iterates `step`.
  - `heap`: Matches the separation logic expressions in premises and conclusion and simplifies them.
- To use the `prep_vc` method, the heap variable in the pre- and postconditions must be called `heap`.

Exercise 2 [3] `Cons` in Separation Logic

On sheet 8 you saw how to implement the operation `cons` on functional lists with a single assignment:

```
q->next = p;
```

Formulate the pre- and post-conditions analogue to list reversal from the exercise.

Pre-condition On the heap there are a list at `p` and a single list node at `q`.
Post-condition On the heap there is a list at `q` and the `data` fields of this list are the `data` field of `q`, followed by the original data fields of `p`.

Exercise 3 [4] (No) Surprises

Conjunction and universal quantifier can be lifted from booleans to separation logic expressions. Prove that the following laws hold:

$$((P \wedge\star Q) \star R)\ heap \implies ((P \star R) \wedge\star (Q \star R))\ heap$$
$$((\forall\star x.\ P\ x) \star Q)\ heap \implies (\forall\star x.\ P\ x \star Q)\ heap$$

Moreover, prove that the converse does not hold for both of these laws, i.e. prove

H. Gast, L. Noschinski

SHEET 8

Separation Logic

Date: 18.12.2012
Hand-In: 25.12.2012, 23:50

$$\neg\big(\forall P\ Q\ R\ heap.\ ((P \star R) \wedge\star\ (Q \star R))\ heap \longrightarrow ((P \wedge\star\ Q) \star R)\ heap\big)$$

$$\neg\big(\forall P\ Q\ heap.\ (\forall\star\ x.\ P\ x \star Q)\ heap \longrightarrow ((\forall\star\ x.\ P\ x) \star Q)\ heap\big)$$