

Theory exercises

Program verification with types and logic (NWI-IMC060)

Week 9

1. For each proposition of separation logic below, state precisely the set of heaps it describes. You should explain your answers. (Recall that \top is **True**, \perp is **False**, and $\neg P$ is $P \rightarrow \perp$.)

- (a) $\exists v. l \mapsto v$
- (b) $\forall v. l \mapsto v$
- (c) $l \mapsto v * (\top \wedge \mathbf{emp})$
- (d) $(l \mapsto v * \top) \wedge \mathbf{emp}$
- (e) $(\exists k. k \mapsto 10 * \top) \wedge (\exists v_1. \exists v_2. l_1 \mapsto v_1 * l_2 \mapsto v_2)$
- (f) $l \mapsto n * \neg(l \mapsto n)$
- (g) $l \mapsto n \wedge \neg(l \mapsto n)$
- (h) $\forall n. (l \mapsto n) \rightarrow \mathbf{emp}$

2. In separation logic, we can define the following variant of the points-to connective:

$$l \hookrightarrow v \triangleq \lambda h. h(l) = v$$

Whereas the ordinary points-to connective $l \mapsto v$ expresses that the heap contains exactly the location l , the connective $l \hookrightarrow v$ expresses that the heap contains *at least* the location l .

- (a) Depending on the kind of language that one wishes to reason about, there are arguments in favor of using either $l \mapsto v$ or $l \hookrightarrow v$. Describe for what kinds of languages $l \mapsto v$ is more suitable, and for what kind of languages $l \hookrightarrow v$ is more suitable.
- (b) Instead of adding $l \hookrightarrow v$ as a primitive to separation logic, one could define it as syntactic sugar:

$$l \hookrightarrow v \triangleq l \mapsto v * \top$$

In a similar way, $l \mapsto v$ can also be defined as syntactic sugar in terms of \hookrightarrow and the other separation logic connectives (\top , \perp , \wedge , \vee , $*$, **emp**, \forall , \exists). Give such a definition and explain why your definition is correct.