Definition 1. *We assume that in every state of the Kripke structure, for every heap h, all objects are ordered by some total order $<_h$, such that there is some object o_h, such that (1) for all $o' <_h o_h$, the object o' is allocated (i.e., its* `<allocated>` *field is set to true), and (2) for all $o_h \leq_h o''$, the object o'' is not allocated. We introduce a unary function symbol* `allocate` *with the signature* `Heap` $\rightarrow$ `Object`*, whose interpretation must adhere to $\mathcal{I}(\mathtt{allocate})(h) = o_h$. The (slightly prettified) rule is as follows:*

$$\frac{\begin{array}{r}\mathit{\Gamma}, \{U\}(\mathtt{v} \not\doteq \mathtt{null} \land \mathtt{v} \doteq \mathtt{allocate}(\mathtt{heap}) \land \mathtt{C}{::}\,\mathtt{exactInstance}(\mathtt{v}) \doteq \mathtt{TRUE}) \\ \Rightarrow \{U\}\{\mathtt{heap} := \mathtt{create}(\mathtt{heap}, \mathtt{v})\}[\mathtt{s}]\phi, \mathit{\Delta}\end{array}}{\mathit{\Gamma} \Rightarrow \{U\}[\mathtt{v\ =\ C.allocate();\ s}]\phi, \mathit{\Delta}}$$