Verifying Memory Managers

John Wickerson

1 Preliminaries

1.1 Spatial closure operators

Suppose R and S are of type $loc \to loc \to assertion$. Define:

$$\begin{aligned} R;S &\stackrel{\text{def}}{=} \lambda x\, z.\, \exists y.\, R\, x\, y \ * \ S\, y\, z \\ R \vee S &\stackrel{\text{def}}{=} \lambda x\, y.\, R\, x\, y \vee S\, x\, y \\ id &\stackrel{\text{def}}{=} \lambda x\, y.\, x = y \wedge emp \\ R^* &\stackrel{\text{def}}{=} \mu S.\, S = id \vee R; S \\ R^+ &\stackrel{\text{def}}{=} R; R^* \end{aligned}$$

Then the ordinary *list* predicate can be defined like so:

$$list(x) \stackrel{\text{def}}{=} (\lambda x\, y.\, x \mapsto y)^*\, x\, 0$$

Furthermore, we can parameterise the definitions by an element m of a partial commutative monoid (PCM) $(M, \cdot, u)$. Define:

$$\begin{aligned} R;S &\stackrel{\text{def}}{=} \lambda x\, z\, m.\, \exists y\, m_1\, m_2.\, m = m_1 \cdot m_2 \ * \ R\, x\, y\, m_1 \ * \ S\, y\, z\, m_2 \\ R \vee S &\stackrel{\text{def}}{=} \lambda x\, y\, m.\, R\, x\, y\, m \vee S\, x\, y\, m \\ id &\stackrel{\text{def}}{=} \lambda x\, y\, m.\, x = y \wedge m = u \wedge emp \\ R^* &\stackrel{\text{def}}{=} \mu S.\, S = id \vee R; S \\ R^+ &\stackrel{\text{def}}{=} R; R^* \end{aligned}$$

Firstly, using the PCM of sets of naturals, $(\mathcal{P}\,\mathbb{N}, \uplus, \emptyset)$, we can define Bornat-style lists, which are parameterised by the set X of locations through which they pass, like so:

$$blist(x, X) \stackrel{\text{def}}{=} (\lambda x\, y\, X.\, x \mapsto y \wedge X = \{x\})^*\, x\, 0\, X$$

Secondly, using the unique 1-element PCM, $(\{u\}, \lambda\_\ \_.\, u, u)$, the extra parameters become redundant, and can be removed in such a way as to restore the original version above.

Thirdly, we can define an arena that comprises a chain of unallocated, allocated, and system blocks – more on this later.

Lemma 1. $R^* = (R^*; R^*)$.

2 Proof system

The model:

$$
\begin{array}{rcll}
x & \in & \mathbb{LV} & \\
\mathtt{x} & \in & \mathbb{PV} & \\
v & \in & \mathsf{Val} & \\
C & \in & \mathsf{Com} & \\
C & \in & \mathsf{ClosedCom} & \stackrel{\text{def}}{=} \{C \in \mathsf{Com} \mid \mathrm{mods}(C) = \emptyset\} \\
\alpha & \in & \mathbb{A} & \\
f & \in & \mathbb{F} & \\
s & \in & \mathsf{Store} & \stackrel{\text{def}}{=} \mathbb{PV} \rightharpoonup \mathsf{Val} \\
h & \in & \mathsf{Heap} & \stackrel{\text{def}}{=} \mathbb{N} \rightharpoonup \mathsf{Val} \\
i & \in & \mathsf{LogEnv} & \stackrel{\text{def}}{=} \mathbb{LV} \rightharpoonup \mathsf{Val} \\
\sigma & \in & \mathsf{State} & \stackrel{\text{def}}{=} \mathsf{Store} \times \mathsf{Heap} \times \mathsf{LogEnv} \\
\pi & \in & \mathsf{PredEnv} & \stackrel{\text{def}}{=} (\alpha : \mathbb{A}) \rightharpoonup (\mathsf{Val}^{\mathrm{arity}(\alpha)} \to \mathcal{P}(\mathsf{State})) \\
\Delta, P, Q & \in & \mathsf{Assn} & \stackrel{\text{def}}{=} \mathsf{PredEnv} \to \mathcal{P}(\mathsf{State}) \\
F & \in & \mathsf{ProcImps} & \stackrel{\text{def}}{=} (f : \mathbb{F}) \rightharpoonup (\mathbb{PV}^{\mathrm{arity}(f)} \to \mathsf{ClosedCom}) \\
\Gamma & \in & \mathsf{ProcSpecs} & \stackrel{\text{def}}{=} (f : \mathbb{F}) \rightharpoonup (\mathbb{PV}^{\mathrm{arity}(f)} \to \mathsf{Assn} \times \mathsf{Assn})
\end{array}
$$

Definition 2 (Semantics of $\Delta \models_F \{P\}\, C\, \{Q\}$)**.**

$$
\begin{array}{l}
\forall \Delta, P, Q, \in \mathsf{Assn}.\, \forall C \in \mathsf{Com}.\, \forall F \in \mathsf{ProcImps}. \\
\Delta \models_F \{P\}\, C\, \{Q\} \stackrel{\text{def}}{=} \forall \sigma, \sigma' \in \mathsf{State}.\, \forall \pi \in \mathsf{PredEnv}. \\
\qquad \text{if } \sigma \in (\Delta \wedge P)(\pi) \\
\qquad \text{then } \neg \langle C, \sigma, F \rangle \lightning \\
\qquad\qquad \text{and if } \langle C, \sigma, F \rangle \Downarrow \sigma' \text{ then } \sigma' \in (Q \wedge \Delta)(\pi)
\end{array}
$$

Definition 3 (Semantics of $\Delta; \Gamma \models C$ sat (P, Q))**.**

$$
\begin{array}{l}
\forall \Gamma \in \mathsf{ProcSpecs}.\, \forall \Delta, P, Q, \in \mathsf{Assn}.\, \forall C \in \mathsf{Com}. \\
\Delta; \Gamma \models \{P\}\, C\, \{Q\} \stackrel{\text{def}}{=} \forall F \in \mathsf{ProcImps}. \\
\qquad \text{if } \forall f \in \mathrm{dom}(\Gamma).\, \forall \bar{\mathtt{x}} \in \mathbb{PV}^{\mathrm{arity}(f)}.\, \Delta \models_F \{\mathit{fst}(\Gamma\, f\, \bar{\mathtt{x}})\}\, F\, f\, \bar{\mathtt{x}}\, \{\mathit{snd}(\Gamma\, f\, \bar{\mathtt{x}})\} \\
\qquad \text{then } \Delta \models_F \{P\}\, C\, \{Q\}
\end{array}
$$

2.1 Separation Logic proof rules

All the normal ones, plus the hypothetical frame rule, below. I've adapted the rule to include a C_{init} command that declares all the variables declared at the top level of the module.

$$\dfrac{\Delta; \Gamma \vdash \{P\}\, C_{\text{init}}\, \{P' * R\} \qquad \begin{array}{c}\overline{\Delta; \Gamma \vdash \{P_i * R\}\, C_i\, \{Q_i * R\}}^{\,i} \\ \Delta; \Gamma, \overline{\{P_i\}\, f_i\, \{Q_i\}}^{\,i} \vdash \{P'\}\, C\, \{Q\}\end{array}}{\Delta; \Gamma \vdash \{P * R\}\, \texttt{module}\ (C_{\text{init}}; \overline{f_i = C_i}^{\,i})\ \texttt{in}\ C\, \{Q * R\}}\ \textsc{HypFrame}$$

2.2 GSep

We extend our model as follows:

$$\begin{array}{rcll}
w & \in & \mathsf{World} & \stackrel{\text{def}}{=} \{(\sigma_l, \sigma_s) \mid \sigma_l, \sigma_s \in \mathsf{State} \wedge \sigma_l \perp \sigma_s\} \\
\rho & \in & \mathsf{GPredEnv} & \stackrel{\text{def}}{=} (\alpha : \mathbb{A}) \rightharpoonup (\mathsf{Val}^{\text{arity}(\alpha)} \to \mathcal{P}(\mathsf{World})) \\
G & \in & \mathsf{Guar} & \stackrel{\text{def}}{=} \mathcal{P}(\mathsf{World} \times \mathsf{World}) \\
\delta, p, q & \in & \mathsf{GAssn} & \stackrel{\text{def}}{=} \mathsf{GPredEnv} \to \mathcal{P}(\mathsf{World}) \\
\gamma & \in & \mathsf{GProcSpecs} & \stackrel{\text{def}}{=} (f : \mathbb{F}) \rightharpoonup (\mathbb{PV}^{\text{arity}(f)} \to \mathsf{GAssn} \times \mathsf{GAssn})
\end{array}$$

Definition 4 (Semantics of $\delta; G \models_F \{p\}\, C\, \{q\}$)**.**

$$\begin{array}{l}
\forall \delta, p, q, \in \mathsf{GAssn}.\, \forall C \in \mathsf{Com}.\, \forall F \in \mathsf{ProcImps}.\, \forall G \in \mathsf{Guar}. \\
\Delta; G \models_F \{p\}\, C\, \{q\} \stackrel{\text{def}}{=} \forall \rho \in \mathsf{GPredEnv}.\, \forall w, w' \in \mathsf{World}. \\
\qquad \text{if } w \in (\delta \wedge p)(\rho) \\
\qquad \text{then } \neg \langle C, \lfloor w \rfloor, F \rangle \lightning \\
\qquad\quad \text{and if } \langle C, \lfloor w \rfloor, F \rangle \Downarrow \lfloor w' \rfloor \\
\qquad\qquad \text{then } \lfloor w' \rfloor \in (q \wedge \delta)(\rho) \\
\qquad\qquad\quad \text{and } (w, w') \in G
\end{array}$$

Definition 5 (Semantics of $\delta; \gamma; G \models \{p\}\, C\, \{q\}$)**.**

$$\begin{array}{l}
\forall \gamma \in \mathsf{GProcSpecs}.\, \forall \delta, p, q, \in \mathsf{GAssn}.\, \forall C \in \mathsf{Com}.\, \forall G \in \mathsf{Guar}. \\
\delta; \gamma; G \models \{p\}\, C\, \{q\} \stackrel{\text{def}}{=} \forall F \in \mathsf{ProcImps}. \\
\qquad \text{if } \forall f \in \text{dom}(\gamma).\, \forall \bar{\mathtt{x}} \in \mathbb{PV}^{\text{arity}(f)}.\, \delta; G \models_F \{\mathit{fst}(\gamma\, f\, \bar{\mathtt{x}})\}\, F\, f\, \bar{\mathtt{x}}\, \{\mathit{snd}(\gamma\, f\, \bar{\mathtt{x}})\} \\
\qquad \text{then } \delta; G \models_F \{p\}\, C\, \{q\}
\end{array}$$

2.3 Proof rules of GSep

Weakening the environment:

$$\dfrac{\delta'; \gamma'; G' \vdash \{p\}\, C\, \{q\} \qquad \delta \Rightarrow \delta' \qquad \gamma \Rightarrow \gamma' \qquad G' \subseteq G}{\delta; \gamma; G \vdash \{p\}\, C\, \{q\}}\ \textsc{Env-Weaken}$$

Rule of consequence:

$$\frac{\delta;\gamma;G\vdash\{p\}\,C\,\{q\}\qquad(\delta\wedge p)\Rightarrow p'\qquad(\delta\wedge q')\Rightarrow q}{\delta;\gamma;G\vdash\{p\}\,C\,\{q\}}\ \textsc{Conseq}$$

Frame rule

$$\frac{\delta;\gamma;G\vdash\{p\}\,C\,\{q\}\qquad r\text{ stable under }G}{\delta;\gamma;G\vdash\{p*r\}\,C\,\{q*r\}}\ \textsc{Frame}$$

Region update

$$\frac{\delta;\gamma\vdash\{P*P'\}\,C\,\{\exists x.\,Q*Q'\}\qquad(P\rightsquigarrow\exists x.\,Q)\subseteq G\qquad P,Q\text{ precise}}{\delta;\gamma;G\vdash\{\boxed{R*P}*P'\}\,C\,\{\exists x.\,\boxed{R*Q}*Q'\}}\ \textsc{RegUpdate}$$

Converting between sequential and GSep specifications

$$\frac{\Delta;\Gamma\vdash\{P\}\,C\,\{Q\}}{G;\Delta;\Gamma\vdash\{P\}\,C\,\{Q\}}\ \textsc{Basic}\qquad\qquad\frac{G;\Delta;\Gamma\vdash\{P\}\,C\,\{Q\}}{\Delta;\Gamma\vdash\{P\}\,C\,\{Q\}}\ \textsc{Erase}$$

3 A variable-sized allocator

External spec

$$\vdash\left\{\mathit{emp}\right\}\texttt{malloc(n)}\left\{(\mathit{token}\,\texttt{ret}\,\lceil\texttt{n/WORD}\rceil\ *\ \bigstar_{i=0}^{\lceil\texttt{n/WORD}\rceil-1}.\,\texttt{ret}+i\mapsto\_)\vee\texttt{ret}=0\right\}$$

$$\vdash\left\{\exists n.\,\mathit{token}\,\texttt{x}\,n\ *\ \bigstar_{i=0}^{n-1}.\,\texttt{x}+i\mapsto\_\right\}\texttt{free(x)}\left\{\mathit{emp}\right\}$$

3.1 Second implementation (Unix V7)

Note that the various 'pure' operators, such as '$=$' and '$>$' and 'def($-$)', are all given an empty footprint. That is, read $x=5$ as $x=5\wedge \mathit{emp}$.

The external spec can be derived from the following internal spec using the hypothetical frame rule (which removes the invariant *anArena*), the rule for weakening predicate environments (which removes Δ), and the Erase rule (which removes G).

Internal spec

$$\delta;\gamma;G\vdash\left\{\mathit{anArena}\right\}\texttt{malloc(n)}\left\{\begin{array}{l}\mathit{anArena}\ *\\ ((\mathit{token}\,\texttt{ret}\,\lceil\texttt{n/WORD}\rceil\ *\ \bigstar_{i=0}^{\lceil\texttt{n/WORD}\rceil-1}.\,\texttt{ret}+i\mapsto\_)\vee\texttt{ret}=0)\end{array}\right\}$$

$$\delta;\gamma;G\vdash\left\{\mathit{anArena}\ *\ \exists n.\,\mathit{token}\,\texttt{x}\,n\ *\ \bigstar_{i=0}^{n-1}.\,\texttt{x}+i\mapsto\_\right\}\texttt{free(x)}\left\{\mathit{anArena}\right\}$$

where δ defines:

$$
\begin{aligned}
\mathit{ublock}\, x\, y\, B &\stackrel{\text{def}}{=} B = \{x+1 \mapsto_{\mathsf{u}} y-x-1\} \;*\; x<y \;*\; x \mapsto y \;*\; \mathop{\bigast}_{i=x+1}^{y-1}.\, i \mapsto \_ \\
\mathit{ablock}\, x\, y\, B &\stackrel{\text{def}}{=} B = \{x+1 \mapsto_{\mathsf{a}} y-x-1\} \;*\; x<y \;*\; x_{|1} \stackrel{.5}{\mapsto} y \\
\mathit{sblock}\, x\, y\, B &\stackrel{\text{def}}{=} B = \{x+1 \mapsto_{\mathsf{s}} y-x-1\} \;*\; x<y \;*\; x_{|1} \mapsto y \\
\mathit{block} &\stackrel{\text{def}}{=} \mathit{ublock} \vee \mathit{ablock} \vee \mathit{sblock} \\
\mathit{uninit}\ A &\stackrel{\text{def}}{=} \mathtt{s} \mapsto 0\, 0 \;*\; A=\emptyset \;*\; \mathit{brka}(\mathtt{s}+2) \\
\mathit{arena}\ A &\stackrel{\text{def}}{=} \exists B_1, B_2 : \mathcal{B}.\, \mathit{block}^*\, \mathtt{s}\, \mathtt{v}\, B_1 \;*\; \mathit{block}^*\, \mathtt{v}\, \mathtt{t}\, B_2 \\
&\qquad * \; A = (B_1 \uplus B_2)^{\mathsf{a}} \;*\; \mathtt{t}_{|1} \mapsto \mathtt{s} \;*\; \mathit{brka}(\mathtt{t}+1) \\
\mathit{anArena} &\stackrel{\text{def}}{=} \boxed{\exists A.\, \mathit{uninit}\ A \;\vee\; \mathit{arena}\ A} \\
\mathit{token}\, x\, n &\stackrel{\text{def}}{=} \boxed{\exists A.\, \mathit{arena}(A \uplus \{x \mapsto n\})} \;*\; (x-1)_{|1} \stackrel{.5}{\mapsto} x+n
\end{aligned}
$$

Note that we use the following separation algebra for the spatial closure operators:

$$\mathcal{B} \stackrel{\text{def}}{=} (\mathbb{N} \rightharpoonup \{\mathsf{u}, \mathsf{a}, \mathsf{s}\} \times \mathbb{N}_0, \uplus, \emptyset)$$

Note also that B^{a} returns a function of type $\mathbb{N} \rightharpoonup \mathbb{N}_0$, such that $(x \mapsto n) \in B^{\mathsf{a}}$ if and only if $(x \mapsto_{\mathsf{a}} n) \in B$.

The guarantee G is defined as $\bigcup_x \{\mathit{Malloc}, \mathit{Free}\, x\}$, where:

$$
\begin{aligned}
\mathit{Malloc} &\stackrel{\text{def}}{=} \exists A, x, n.\, (\mathtt{s} \mapsto 0\, 0 \;*\; A=\emptyset) \;\vee\; \mathit{arena}\, A \;\rightsquigarrow\; \mathit{arena}(A \uplus \{x \mapsto n\}) \\
\mathit{Free}\, x &\stackrel{\text{def}}{=} \exists A, n.\, (x-1)_{|1} \stackrel{.5}{\mapsto} (x+n) \;\mid\; \mathit{arena}(A \uplus \{x \mapsto n\}) \;\rightsquigarrow\; \mathit{arena}\, A
\end{aligned}
$$

The procedure environment γ provides a specification for `sbrk`. The 'official' spec for `sbrk`is as follows:

$$\vdash \left\{\mathit{brk}(b)\right\} \mathtt{sbrk(n)} \left\{\begin{array}{l}(\mathit{brk}(b) \;*\; \mathtt{ret} = -1 \;*\; \mathtt{n} \neq 0) \vee \\ (\mathit{brk}(b + \lceil \mathtt{n}/\mathtt{WORD} \rceil) \;*\; \mathtt{ret} = b \;*\; \mathop{\bigast}_{i=0}^{\lceil \mathtt{n}/\mathtt{WORD} \rceil - 1}.\, \mathtt{ret} + i \mapsto \_)\end{array}\right\}$$

but if we define $\mathit{brka}(x)$ as shorthand for $\exists b \geq x.\, \mathit{brk}(b)$, then we obtain the following derived spec:

$$\vdash \left\{\mathit{brka}(x)\right\} \mathtt{sbrk(n)} \left\{\begin{array}{l}(\mathit{brka}(x) \;*\; \mathtt{ret} = -1 \;*\; \mathtt{n} \neq 0) \vee \\ (\mathit{brka}(\mathtt{ret} + \lceil \mathtt{n}/\mathtt{WORD} \rceil) \;*\; x \leq \mathtt{ret} \;*\; \mathop{\bigast}_{i=0}^{\lceil \mathtt{n}/\mathtt{WORD} \rceil - 1}.\, \mathtt{ret} + i \mapsto \_)\end{array}\right\}$$

which is easier to use, and is hence the one contained in γ.

The verification of the module depends on the following two lemmas:

Lemma 6. $\mathit{block}^*\, x_1\, y_1\, B_1 \;*\; \mathit{block}^*\, x_2\, y_2\, B_2 \implies B_1 \perp B_2$

Lemma 7. $\mathit{block}^*\, x\, y\, B \;*\; w \mapsto z \implies w+1 \notin \mathrm{dom}(B)$

Verification of malloc routine

```
#define WORD sizeof(union store)
#define BLOCK 1024 /* a multiple of WORD*/
#define testbusy(p) ((int)(p)&1)
#define setbusy(p) (struct store *)((int)(p)|1)
#define clearbusy(p) (struct store *)((int)(p)&~1)

struct store {struct store *ptr;};
static struct store s[2]; /* initial arena */
static struct store *v; /* search ptr */
static struct store *t; /* arena top */

char *malloc(unsigned int nbytes)
```

$\left\{ anArena \right\}$

$\left\{ \boxed{\exists A.\, uninit\, A \;\vee\; arena\, A} \right\}$

```
// begin Existential
```

$\left\{ \boxed{uninit\, A \;\vee\; arena\, A} \right\}$

```
// begin region update (action is either Malloc or none)
```

$\left\{ uninit\, A \;\vee\; arena\, A \right\}$

```
// Precondition for returning:
```

$$\left\{ \begin{pmatrix} arena(A \uplus \{\mathtt{ret} \mapsto \lceil \mathtt{nbytes/WORD} \rceil\}) \\ * \;\; \bigast_{i=0}^{\lceil \mathtt{nbytes/WORD} \rceil - 1}.\, \mathtt{ret} + i \mapsto \_ \\ * \;\; (\mathtt{ret}-1)_{|1} \overset{.5}{\mapsto} \mathtt{ret} + \lceil \mathtt{nbytes/WORD} \rceil \end{pmatrix} \vee (arena\ A \;*\; \mathtt{ret} = 0) \right\}$$

```
{
```

$\left\{ uninit\, A \;\vee\; arena\, A \right\}$

```
  register struct store *p, *q;
  register nw;
  static temp;
  if(s[0].ptr == 0) { /*first time*/
```

$\left\{ uninit\, A \right\}$

$\left\{ \mathtt{s} \mapsto 0\,0 \;*\; brka(\mathtt{s}+2) \;*\; A = \emptyset \right\}$

```
    s[0].ptr = setbusy(&s[1]);
```

$\left\{ \mathtt{s}_{|1} \mapsto \mathtt{s}+1 \;*\; \mathtt{s}+1 \mapsto 0 \;*\; brka(\mathtt{s}+2) \;*\; A = \emptyset \right\}$

```
    s[1].ptr = setbusy(&s[0]);
```

$\left\{ \mathtt{s}_{|1} \mapsto \mathtt{s}+1 \;*\; (\mathtt{s}+1)_{|1} \mapsto \mathtt{s} \;*\; brka(\mathtt{s}+2) \;*\; A = \emptyset \right\}$

```
t = &s[1];
```

$$\left\{\mathtt{s}_{|1} \mapsto \mathtt{t} \;*\; \mathtt{t}_{|1} \mapsto \mathtt{s} \;*\; \mathtt{s} < \mathtt{t} \;*\; \mathit{brka}(\mathtt{t}+1) \;*\; A = \emptyset\right\}$$

```
v = &s[0];
```

$$\left\{\mathtt{s}_{|1} \mapsto \mathtt{t} \;*\; \mathtt{t}_{|1} \mapsto \mathtt{s} \;*\; \mathtt{s} < \mathtt{t} \;*\; \mathtt{v} = \mathtt{s} \;*\; \mathit{brka}(\mathtt{t}+1) \;*\; A = \emptyset\right\}$$

$$\left\{\mathit{sblock}\,\mathtt{s}\,\mathtt{t}\,\{\mathtt{s}+1 \mapsto_{\mathsf{s}} 0\} \;*\; \mathtt{t}_{|1} \mapsto \mathtt{s} \;*\; \mathtt{v} = \mathtt{s} \;*\; \mathit{brka}(\mathtt{t}+1) \;*\; A = \emptyset\right\}$$

$$\left\{\begin{array}{l}\exists B_1, B_2.\, \mathit{block}^*\,\mathtt{s}\,\mathtt{v}\,B_1 \;*\; \mathit{block}^*\,\mathtt{v}\,\mathtt{t}\,B_2 \;*\; \mathtt{t}_{|1} \mapsto \mathtt{s} \\ \;*\; A = (B_1 \uplus B_2)^{\mathsf{a}} \;*\; \mathit{brka}(\mathtt{t}+1) \;*\; A = \emptyset\end{array}\right\}$$

$$\left\{\mathit{arena}\,A \;*\; A = \emptyset\right\}$$

$$\left\{\mathit{arena}\,A\right\}$$

```
}
```

$$\left\{\begin{array}{l}\exists B_1, B_2.\, \mathit{block}^*\,\mathtt{s}\,\mathtt{v}\,B_1 \;*\; \mathit{block}^*\,\mathtt{v}\,\mathtt{t}\,B_2 \;*\; \mathtt{t}_{|1} \mapsto \mathtt{s} \\ \;*\; A = (B_1 \uplus B_2)^{\mathsf{a}} \;*\; \mathit{brka}(\mathtt{t}+1)\end{array}\right\}$$

```
nw=(nbytes+WORD+WORD-1)/WORD;
```

$$\left\{\begin{array}{l}\exists B_1, B_2.\, \mathit{block}^*\,\mathtt{s}\,\mathtt{v}\,B_1 \;*\; \mathit{block}^*\,\mathtt{v}\,\mathtt{t}\,B_2 \;*\; \mathtt{t}_{|1} \mapsto \mathtt{s} \\ \;*\; A = (B_1 \uplus B_2)^{\mathsf{a}} \;*\; \mathit{brka}(\mathtt{t}+1) \;*\; \mathtt{nw} = 1 + \left\lceil \frac{\mathtt{nbytes}}{\mathtt{WORD}} \right\rceil\end{array}\right\}$$

```
for(p=v; ; ) {
  // Loop inv 1:
```

$$\left\{\begin{array}{l}\exists B_1, B_2.\, \mathit{block}^*\,\mathtt{s}\,\mathtt{p}\,B_1 \;*\; \mathit{block}^*\,\mathtt{p}\,\mathtt{t}\,B_2 \;*\; \mathtt{t}_{|1} \mapsto \mathtt{s} \\ \;*\; A = (B_1 \uplus B_2)^{\mathsf{a}} \;*\; \mathit{brka}(\mathtt{t}+1) \;*\; \mathtt{nw} = 1 + \left\lceil \frac{\mathtt{nbytes}}{\mathtt{WORD}} \right\rceil\end{array}\right\}$$

```
  for(temp=0; ; ) {
    // Loop inv 2:
```

$$\left\{\begin{array}{l}\exists B_1, B_2.\, \mathit{block}^*\,\mathtt{s}\,\mathtt{p}\,B_1 \;*\; \mathit{block}^*\,\mathtt{p}\,\mathtt{t}\,B_2 \;*\; \mathtt{t}_{|1} \mapsto \mathtt{s} \\ \;*\; A = (B_1 \uplus B_2)^{\mathsf{a}} \;*\; \mathit{brka}(\mathtt{t}+1) \;*\; \mathtt{nw} = 1 + \left\lceil \frac{\mathtt{nbytes}}{\mathtt{WORD}} \right\rceil\end{array}\right\}$$

```
    if(!testbusy(p->ptr)) {
```

$$\left\{\begin{array}{l}\exists B_1, B_2, q.\, \mathit{block}^*\,\mathtt{s}\,\mathtt{p}\,B_1 \;*\; \mathit{ublock}\,\mathtt{p}\,q\,\{\mathtt{p}+1 \mapsto_{\mathsf{u}} q-\mathtt{p}-1\} \;*\; \mathit{block}^*\,q\,\mathtt{t}\,B_2 \\ \;*\; \mathtt{t}_{|1} \mapsto \mathtt{s} \;*\; A = (B_1 \uplus B_2)^{\mathsf{a}} \;*\; \mathit{brka}(\mathtt{t}+1) \;*\; \mathtt{nw} = 1 + \left\lceil \frac{\mathtt{nbytes}}{\mathtt{WORD}} \right\rceil\end{array}\right\}$$

```
      while(!testbusy((q=p->ptr)->ptr)) {
```

$$\left\{\begin{array}{l}\exists B_1, B_2, r.\, \mathit{block}^*\,\mathtt{s}\,\mathtt{p}\,B_1 \;*\; \mathit{ublock}\,\mathtt{p}\,\mathtt{q}\,\{\mathtt{p}+1 \mapsto_{\mathsf{u}} \mathtt{q}-\mathtt{p}-1\} \\ \;*\; \mathit{ublock}\,\mathtt{q}\,r\,\{\mathtt{q}+1 \mapsto_{\mathsf{u}} r-\mathtt{q}-1\} \;*\; \mathit{block}^*\,r\,\mathtt{t}\,B_2 \;*\; \mathtt{t}_{|1} \mapsto \mathtt{s} \\ \;*\; A = (B_1 \uplus B_2)^{\mathsf{a}} \;*\; \mathit{brka}(\mathtt{t}+1) \;*\; \mathtt{nw} = 1 + \left\lceil \frac{\mathtt{nbytes}}{\mathtt{WORD}} \right\rceil\end{array}\right\}$$

```
        p->ptr = q->ptr; // coalesce consecutive free blocks
```

$$\left\{\begin{array}{l}\exists B_1, B_2, r.\, \mathit{block}^*\,\mathtt{s}\,\mathtt{p}\,B_1 \;*\; \mathit{ublock}\,\mathtt{p}\,r\,\{\mathtt{p}+1 \mapsto_{\mathsf{u}} r-\mathtt{p}-1\} \;*\; \mathit{block}^*\,r\,\mathtt{t}\,B_2 \\ \;*\; \mathtt{t}_{|1} \mapsto \mathtt{s} \;*\; A = (B_1 \uplus B_2)^{\mathsf{a}} \;*\; \mathit{brka}(\mathtt{t}+1) \;*\; \mathtt{nw} = 1 + \left\lceil \frac{\mathtt{nbytes}}{\mathtt{WORD}} \right\rceil\end{array}\right\}$$

```
      }
```

$$\left\{\begin{array}{l}\exists B_1, B_2.\, \mathit{block}^*\,\mathtt{s}\,\mathtt{p}\,B_1 \;*\; \mathit{ublock}\,\mathtt{p}\,\mathtt{q}\,\{\mathtt{p}+1 \mapsto_{\mathsf{u}} \mathtt{q}-\mathtt{p}-1\} \;*\; \mathit{block}^*\,\mathtt{q}\,\mathtt{t}\,B_2\\ *\; \mathtt{t}_{|1} \mapsto \mathtt{s} \;*\; A = (B_1 \uplus B_2)^{\mathsf{a}} \;*\; \mathit{brka}(\mathtt{t}+1) \;*\; \mathtt{nw} = 1 + \left\lceil \frac{\mathtt{nbytes}}{\mathtt{WORD}} \right\rceil\end{array}\right\}$$

```
  if(q>=p+nw && p+nw>=p) {
```

$$\left\{\begin{array}{l}\exists B_1, B_2.\, \mathit{block}^*\,\mathtt{s}\,\mathtt{p}\,B_1 \;*\; \mathit{ublock}\,\mathtt{p}\,\mathtt{q}\,\{\mathtt{p}+1 \mapsto_{\mathsf{u}} \mathtt{q}-\mathtt{p}-1\}\\ *\; \mathit{block}^*\,\mathtt{q}\,\mathtt{t}\,B_2 \;*\; \mathtt{t}_{|1} \mapsto \mathtt{s} \;*\; A = (B_1 \uplus B_2)^{\mathsf{a}} \;*\; \mathit{brka}(\mathtt{t}+1)\\ *\; \mathtt{nw} = 1 + \left\lceil \frac{\mathtt{nbytes}}{\mathtt{WORD}} \right\rceil \;*\; \mathtt{q} \geq \mathtt{p} + \mathtt{nw}\end{array}\right\}$$

```
    goto found;
```

$$\left\{\mathsf{false}\right\}$$

```
  }
}
// p's block is unavailable / too small,
// or p points to the top of the arena
```

$$\left\{\begin{array}{l}\exists B_1, B_2.\, \mathit{block}^*\,\mathtt{s}\,\mathtt{p}\,B_1 \;*\; \mathit{block}^*\,\mathtt{p}\,\mathtt{t}\,B_2 \;*\; A = (B_1 \uplus B_2)^{\mathsf{a}}\\ *\; \mathtt{t}_{|1} \mapsto \mathtt{s} \;*\; \mathit{brka}(\mathtt{t}+1) \;*\; \mathtt{nw} = 1 + \left\lceil \frac{\mathtt{nbytes}}{\mathtt{WORD}} \right\rceil\end{array}\right\}$$

```
q = p;
```

$$\left\{\begin{array}{l}\exists B_1, B_2.\, \mathit{block}^*\,\mathtt{s}\,\mathtt{q}\,B_1 \;*\; \mathit{block}^*\,\mathtt{q}\,\mathtt{t}\,B_2 \;*\; A = (B_1 \uplus B_2)^{\mathsf{a}}\\ *\; \mathtt{t}_{|1} \mapsto \mathtt{s} \;*\; \mathit{brka}(\mathtt{t}+1) \;*\; \mathtt{nw} = 1 + \left\lceil \frac{\mathtt{nbytes}}{\mathtt{WORD}} \right\rceil \;*\; \mathtt{q} = \mathtt{p}\end{array}\right\}$$

$$\left\{\begin{array}{l}\exists B_1, B_2.\, \mathit{block}^*\,\mathtt{s}\,\mathtt{q}\,B_1 \;*\; \mathit{block}^*\,\mathtt{q}\,\mathtt{t}\,B_2 \;*\; A = (B_1 \uplus B_2)^{\mathsf{a}}\\ *\; \mathtt{t}_{|1} \mapsto \mathtt{s} \;*\; \mathit{brka}(\mathtt{t}+1) \;*\; \mathtt{nw} = 1 + \left\lceil \frac{\mathtt{nbytes}}{\mathtt{WORD}} \right\rceil \;*\; \mathtt{q} = \mathtt{p}\end{array}\right\}$$

```
p = clearbusy(p->ptr);
```

$$\left\{\begin{array}{l}((\exists B_1, B_2, \tau.\, \mathit{block}^*\,\mathtt{s}\,\mathtt{q}\,B_1 \;*\; \mathit{block}\,\mathtt{q}\,\mathtt{p}\,\{\mathtt{q}+1 \mapsto_{\tau} \mathtt{p}-\mathtt{q}-1\}\\ *\; \mathit{block}^*\,\mathtt{p}\,\mathtt{t}\,B_2 \;*\; A = (B_1 \uplus \{\mathtt{q}+1 \mapsto_{\tau} \mathtt{p}-\mathtt{q}-1\} \uplus B_2)^{\mathsf{a}})\\ \vee (\exists B.\, \mathit{block}^*\,\mathtt{s}\,\mathtt{q}\,B \;*\; A = B^{\mathsf{a}} \;*\; \mathtt{q} = \mathtt{t} \;*\; \mathtt{p} = \mathtt{s}))\\ *\; \mathtt{t}_{|1} \mapsto \mathtt{s} \;*\; \mathit{brka}(\mathtt{t}+1) \;*\; \mathtt{nw} = 1 + \left\lceil \frac{\mathtt{nbytes}}{\mathtt{WORD}} \right\rceil\end{array}\right\}$$

```
if(p>q) {
```

$$\left\{\begin{array}{l}\exists B_1, B_2, \tau.\, \mathit{block}^*\,\mathtt{s}\,\mathtt{q}\,B_1 \;*\; \mathit{block}\,\mathtt{q}\,\mathtt{p}\,\{\mathtt{q}+1 \mapsto_{\tau} \mathtt{p}-\mathtt{q}-1\}\\ *\; \mathit{block}^*\,\mathtt{p}\,\mathtt{t}\,B_2 \;*\; A = (B_1 \uplus \{\mathtt{q}+1 \mapsto_{\tau} \mathtt{p}-\mathtt{q}-1\} \uplus B_2)^{\mathsf{a}}\\ *\; \mathtt{t}_{|1} \mapsto \mathtt{s} \;*\; \mathit{brka}(\mathtt{t}+1) \;*\; \mathtt{nw} = 1 + \left\lceil \frac{\mathtt{nbytes}}{\mathtt{WORD}} \right\rceil\end{array}\right\}$$

```
} else if(q!=t || p!=s) {
```

$$\left\{\begin{array}{l}\exists B.\, \mathit{block}^*\,\mathtt{s}\,\mathtt{q}\,B \;*\; \mathtt{t}_{|1} \mapsto \mathtt{s} \;*\; A = B^{\mathsf{a}} \;*\; \mathit{brka}(\mathtt{t}+1)\\ *\; \mathtt{nw} = 1 + \left\lceil \frac{\mathtt{nbytes}}{\mathtt{WORD}} \right\rceil \;*\; \mathtt{q} = \mathtt{t} \;*\; \mathtt{p} = \mathtt{s} \;*\; (\mathtt{q} \neq \mathtt{t} \vee \mathtt{p} \neq \mathtt{s})\end{array}\right\}$$

$$\left\{\mathsf{false}\right\}$$

```
  return 0; // unreachable
```

$$\left\{\mathsf{false}\right\}$$

```
} else if(++temp>1) {
```

$$\left\{\begin{array}{l}\exists B.\, \mathit{block}^* \, \mathtt{s}\,\mathtt{q}\, B \;*\; \mathtt{t}_{|1} \mapsto \mathtt{s} \;*\; A = B^{\mathsf{a}} \;*\; \mathit{brka}(\mathtt{t}+1)\\ *\; \mathtt{nw} = 1 + \left\lceil \frac{\mathtt{nbytes}}{\mathtt{WORD}} \right\rceil \;*\; \mathtt{q} = \mathtt{t} \;*\; \mathtt{p} = \mathtt{s}\end{array}\right\}$$

```
    break; // jump to [Extend arena]
```

$$\left\{\mathsf{false}\right\}$$

```
  }
  // Reestablish loop inv 2:
```

$$\left\{\begin{array}{l}\exists B_1, B_2.\, \mathit{block}^* \, \mathtt{s}\,\mathtt{p}\, B_1 \;*\; \mathit{block}^* \, \mathtt{p}\,\mathtt{t}\, B_2 \;*\; A = (B_1 \uplus B_2)^{\mathsf{a}}\\ *\; \mathtt{t}_{|1} \mapsto \mathtt{s} \;*\; \mathit{brka}(\mathtt{t}+1) \;*\; \mathtt{nw} = 1 + \left\lceil \frac{\mathtt{nbytes}}{\mathtt{WORD}} \right\rceil\end{array}\right\}$$

```
}
// We never exit the loop 'normally' (because the non-existent
// test condition never fails). We only reach this point by
// breaking.
// [Extend arena]:
```

$$\left\{\begin{array}{l}\exists B.\, \mathit{block}^* \, \mathtt{s}\,\mathtt{t}\, B \;*\; \mathtt{t}_{|1} \mapsto \mathtt{s} \;*\; A = B^{\mathsf{a}}\\ *\; \mathit{brka}(\mathtt{t}+1) \;*\; \mathtt{nw} = 1 + \left\lceil \frac{\mathtt{nbytes}}{\mathtt{WORD}} \right\rceil \;*\; \mathtt{p} = \mathtt{s}\end{array}\right\}$$

```
temp = ((nw+BLOCK/WORD)/(BLOCK/WORD))*(BLOCK/WORD);
```

$$\left\{\begin{array}{l}\exists B.\, \mathit{block}^* \, \mathtt{s}\,\mathtt{t}\, B \;*\; \mathtt{t}_{|1} \mapsto \mathtt{s} \;*\; A = B^{\mathsf{a}} \;*\; \mathit{brka}(\mathtt{t}+1)\\ *\; \mathtt{nw} = 1 + \left\lceil \frac{\mathtt{nbytes}}{\mathtt{WORD}} \right\rceil \;*\; \mathtt{p} = \mathtt{s} \;*\; \mathtt{temp} > \mathtt{nw}\end{array}\right\}$$

```
q = (struct store *)sbrk(0);
```

// note that $\mathit{brka}(\mathtt{q}) \implies \mathit{brka}(\mathtt{t}+1)$ because $\mathtt{q} \geq \mathtt{t}+1$

$$\left\{\begin{array}{l}\exists B.\, \mathit{block}^* \, \mathtt{s}\,\mathtt{t}\, B \;*\; \mathtt{t}_{|1} \mapsto \mathtt{s} \;*\; A = B^{\mathsf{a}} \;*\; \mathit{brka}(\mathtt{t}+1)\\ *\; \mathtt{nw} = 1 + \left\lceil \frac{\mathtt{nbytes}}{\mathtt{WORD}} \right\rceil \;*\; \mathtt{p} = \mathtt{s} \;*\; \mathtt{temp} > \mathtt{nw} \;*\; \mathtt{q} \geq \mathtt{t}+1\end{array}\right\}$$

```
if(q + temp < q) {
```

$\left\{\mathsf{false}\right\}$ // integer overflows aren't modelled

```
  return 0;
```

$$\left\{\mathsf{false}\right\}$$

```
}
```

$$\left\{\begin{array}{l}\exists B.\, \mathit{block}^* \, \mathtt{s}\,\mathtt{t}\, B \;*\; \mathtt{t}_{|1} \mapsto \mathtt{s} \;*\; A = B^{\mathsf{a}} \;*\; \mathit{brka}(\mathtt{t}+1)\\ *\; \mathtt{nw} = 1 + \left\lceil \frac{\mathtt{nbytes}}{\mathtt{WORD}} \right\rceil \;*\; \mathtt{p} = \mathtt{s} \;*\; \mathtt{temp} > \mathtt{nw} \;*\; \mathtt{q} \geq \mathtt{t}+1\end{array}\right\}$$

```
q = (struct store *)sbrk(temp * WORD);
```

$$\left\{\begin{array}{l}\exists B.\, \mathit{block}^* \, \mathtt{s}\,\mathtt{t}\, B \;*\; \mathtt{t}_{|1} \mapsto \mathtt{s} \;*\; A = B^{\mathsf{a}} \;*\; \mathtt{nw} = 1 + \left\lceil \frac{\mathtt{nbytes}}{\mathtt{WORD}} \right\rceil\\ *\; \mathtt{p} = \mathtt{s} \;*\; \mathtt{temp} > \mathtt{nw} \;*\; ((\mathit{brka}(\mathtt{t}+1) \;*\; \mathtt{q} = -1)\\ \vee\, (\mathit{brka}(\mathtt{q}+\mathtt{temp}) \;*\; \mathtt{t}+1 \leq \mathtt{q} \;*\; \mathop{\huge *}_{i=0}^{\mathtt{temp}-1} . \, \mathtt{q}+i \mapsto \_))\end{array}\right\}$$

```
if((INT)q == -1) {
```

$$\left\{\exists B.\, \mathit{block}^* \, \mathtt{s}\,\mathtt{t}\, B \;*\; \mathtt{t}_{|1} \mapsto \mathtt{s} \;*\; A = B^{\mathsf{a}} \;*\; \mathit{brka}(\mathtt{t}+1)\right\}$$

```
  v = s; // line added to fix bug
```

$$\left\{\begin{array}{l}\exists B_1, B_2.\ \mathit{block}^*\,\mathtt{s}\,\mathtt{v}\,B_1 \ *\ \mathit{block}^*\,\mathtt{v}\,\mathtt{t}\,B_2 \ *\ \mathtt{t}_{|1} \mapsto \mathtt{s} \\ *\ A = (B_1 \uplus B_2)^{\mathsf{a}} \ *\ \mathit{brka}(\mathtt{t}+1)\end{array}\right\}$$

$$\left\{\mathit{arena}\,A\right\}$$

$$\left\{(\mathit{arena}\,A \ *\ \mathtt{ret} = 0)[0/\mathtt{ret}]\right\}$$

```
return 0;
```

$$\left\{\mathsf{false}\right\}$$

```
}
```

$$\left\{\begin{array}{l}\exists B.\ \mathit{block}^*\,\mathtt{s}\,\mathtt{t}\,B \ *\ \mathtt{t}_{|1} \mapsto \mathtt{s} \ *\ A = B^{\mathsf{a}} \ *\ \mathtt{nw} = 1 + \left\lceil\frac{\mathtt{nbytes}}{\mathtt{WORD}}\right\rceil \ *\ \mathtt{p} = \mathtt{s} \\ *\ \mathtt{temp} > \mathtt{nw} \ *\ \mathit{brka}(\mathtt{q}+\mathtt{temp}) \ *\ \mathtt{t}+1 \le \mathtt{q} \ *\ \bigstar_{i=0}^{\mathtt{temp}-1}.\, \mathtt{q}+i \mapsto \_\end{array}\right\}$$

```
t->ptr = q;
```

$$\left\{\begin{array}{l}\exists B.\ \mathit{block}^*\,\mathtt{s}\,\mathtt{t}\,B \ *\ \mathtt{t} \mapsto \mathtt{q} \ *\ A = B^{\mathsf{a}} \ *\ \mathtt{nw} = 1 + \left\lceil\frac{\mathtt{nbytes}}{\mathtt{WORD}}\right\rceil \ *\ \mathtt{p} = \mathtt{s} \\ *\ \mathtt{temp} > \mathtt{nw} \ *\ \mathit{brka}(\mathtt{q}+\mathtt{temp}) \ *\ \mathtt{t}+1 \le \mathtt{q} \ *\ \bigstar_{i=0}^{\mathtt{temp}-1}.\, \mathtt{q}+i \mapsto \_\end{array}\right\}$$

```
if(q!=t+1) {
```

$$\left\{\begin{array}{l}\exists B.\ \mathit{block}^*\,\mathtt{s}\,\mathtt{t}\,B \ *\ \mathtt{t} \mapsto \mathtt{q} \ *\ A = B^{\mathsf{a}} \ *\ \mathtt{nw} = 1 + \left\lceil\frac{\mathtt{nbytes}}{\mathtt{WORD}}\right\rceil \ *\ \mathtt{p} = \mathtt{s} \\ *\ \mathtt{temp} > \mathtt{nw} \ *\ \mathit{brka}(\mathtt{q}+\mathtt{temp}) \ *\ \mathtt{t}+1 < \mathtt{q} \ *\ \bigstar_{i=0}^{\mathtt{temp}-1}.\, \mathtt{q}+i \mapsto \_\end{array}\right\}$$

```
  t->ptr = setbusy(t->ptr);
```

$$\left\{\begin{array}{l}\exists B.\ \mathit{block}^*\,\mathtt{s}\,\mathtt{t}\,B \ *\ \mathtt{t}_{|1} \mapsto \mathtt{q} \ *\ A = B^{\mathsf{a}} \ *\ \mathtt{nw} = 1 + \left\lceil\frac{\mathtt{nbytes}}{\mathtt{WORD}}\right\rceil \ *\ \mathtt{p} = \mathtt{s} \\ *\ \mathtt{temp} > \mathtt{nw} \ *\ \mathit{brka}(\mathtt{q}+\mathtt{temp}) \ *\ \mathtt{t}+1 < \mathtt{q} \ *\ \bigstar_{i=0}^{\mathtt{temp}-1}.\, \mathtt{q}+i \mapsto \_\end{array}\right\}$$

$$\left\{\begin{array}{l}\exists B.\ \mathit{block}^*\,\mathtt{s}\,\mathtt{t}\,B \ *\ \mathit{sblock}\,\mathtt{t}\,\mathtt{q}\,\{\mathtt{t}+1 \mapsto_{\mathsf{s}} \mathtt{q}-\mathtt{t}-1\} \ *\ A = B^{\mathsf{a}} \ *\ \mathtt{nw} = 1 + \left\lceil\frac{\mathtt{nbytes}}{\mathtt{WORD}}\right\rceil \\ *\ \mathtt{p} = \mathtt{s} \ *\ \mathtt{temp} > \mathtt{nw} \ *\ \mathit{brka}(\mathtt{q}+\mathtt{temp}) \ *\ \bigstar_{i=0}^{\mathtt{temp}-1}.\, \mathtt{q}+i \mapsto \_\end{array}\right\}$$

```
}
// t is either a ublock of size 0 or an sblock
```

$$\left\{\begin{array}{l}\exists B, \tau.\ \mathit{block}^*\,\mathtt{s}\,\mathtt{t}\,B \ *\ \mathit{block}\,\mathtt{t}\,\mathtt{q}\,\{\mathtt{t}+1 \mapsto_{\tau} \mathtt{q}-\mathtt{t}-1\} \ *\ A = B^{\mathsf{a}} \ *\ \mathtt{nw} = 1 + \left\lceil\frac{\mathtt{nbytes}}{\mathtt{WORD}}\right\rceil \\ *\ \mathtt{p} = \mathtt{s} \ *\ \mathtt{temp} > \mathtt{nw} \ *\ \mathit{brka}(\mathtt{q}+\mathtt{temp}) \ *\ \bigstar_{i=0}^{\mathtt{temp}-1}.\, \mathtt{q}+i \mapsto \_\end{array}\right\}$$

```
// B swallows the block at t. A=B^a still holds because
// the block at t isn't allocated.
```

$$\left\{\begin{array}{l}\exists B.\ \mathit{block}^*\,\mathtt{s}\,\mathtt{q}\,B \ *\ A = B^{\mathsf{a}} \ *\ \mathtt{nw} = 1 + \left\lceil\frac{\mathtt{nbytes}}{\mathtt{WORD}}\right\rceil \ *\ \mathtt{p} = \mathtt{s} \ *\ \mathtt{temp} > \mathtt{nw} \\ *\ \mathit{brka}(\mathtt{q}+\mathtt{temp}) \ *\ \mathtt{q} \mapsto \_ \ *\ \bigstar_{i=\mathtt{q}+1}^{\mathtt{q}+\mathtt{temp}-2}.\, i \mapsto \_ \ *\ (\mathtt{q}+\mathtt{temp}-1) \mapsto \_\end{array}\right\}$$

```
t = q->ptr = q+temp-1;
```

$$\left\{\begin{array}{l}\exists B.\ \mathit{block}^*\,\mathtt{s}\,\mathtt{q}\,B \ *\ A = B^{\mathsf{a}} \ *\ \mathtt{nw} = 1 + \left\lceil\frac{\mathtt{nbytes}}{\mathtt{WORD}}\right\rceil \ *\ \mathtt{p} = \mathtt{s} \\ *\ \mathit{brka}(\mathtt{t}+1) \ *\ \mathtt{q} < \mathtt{t} \ *\ \mathtt{q} \mapsto \mathtt{t} \ *\ \bigstar_{i=\mathtt{q}+1}^{\mathtt{t}-1}.\, i \mapsto \_ \ *\ \mathtt{t} \mapsto \_\end{array}\right\}$$

$$\left\{\begin{array}{l}\exists B.\ \mathit{block}^*\,\mathtt{s}\,\mathtt{q}\,B \ *\ A = B^{\mathsf{a}} \ *\ \mathtt{nw} = 1 + \left\lceil\frac{\mathtt{nbytes}}{\mathtt{WORD}}\right\rceil \ *\ \mathtt{p} = \mathtt{s} \\ *\ \mathit{brka}(\mathtt{t}+1) \ *\ \mathit{ublock}\,\mathtt{q}\,\mathtt{t}\,\{\mathtt{q}+1 \mapsto_{\mathsf{u}} \mathtt{t}-\mathtt{q}-1\} \ *\ \mathtt{t} \mapsto \_\end{array}\right\}$$

```
// B swallows the block at q. A=B^a still holds because
// the block at q isn't allocated.
```

$$\left\{\begin{array}{l}\exists B.\, \mathit{block}^{*}\, \mathtt{s}\, \mathtt{t}\, B \;*\; A = B^{\mathsf{a}} \;*\; \mathtt{nw} = 1 + \left\lceil \frac{\mathtt{nbytes}}{\mathtt{WORD}} \right\rceil \\ *\; \mathtt{p} = \mathtt{s} \;*\; \mathit{brka}(\mathtt{t}+1) \;*\; \mathtt{t} \mapsto \_\end{array}\right\}$$

```
t->ptr = setbusy(s);
// reestablish loop inv 1:
```

$$\left\{\begin{array}{l}\exists B_1, B_2.\, \mathit{block}^{*}\, \mathtt{s}\, \mathtt{p}\, B_1 \;*\; \mathit{block}^{*}\, \mathtt{p}\, \mathtt{t}\, B_2 \;*\; \mathtt{t}_{|1} \mapsto \mathtt{s} \\ *\; A = (B_1 \uplus B_2)^{\mathsf{a}} \;*\; \mathit{brka}(\mathtt{t}+1) \;*\; \mathtt{nw} = 1 + \left\lceil \frac{\mathtt{nbytes}}{\mathtt{WORD}} \right\rceil\end{array}\right\}$$

```
}
```

$$\left\{\mathsf{false}\right\}$$

```
found:
```

$$\left\{\begin{array}{l}\exists B_1, B_2.\, \mathit{block}^{*}\, \mathtt{s}\, \mathtt{p}\, B_1 \;*\; \mathit{ublock}\, \mathtt{p}\, \mathtt{q}\, \{\mathtt{p}+1 \mapsto_{\mathsf{u}} \mathtt{q}-\mathtt{p}-1\} \\ *\; \mathit{block}^{*}\, \mathtt{q}\, \mathtt{t}\, B_2 \;*\; \mathtt{t}_{|1} \mapsto \mathtt{s} \;*\; A = (B_1 \uplus B_2)^{\mathsf{a}} \;*\; \mathit{brka}(\mathtt{t}+1) \\ *\; \mathtt{nw} = 1 + \left\lceil \frac{\mathtt{nbytes}}{\mathtt{WORD}} \right\rceil \;*\; \mathtt{q} \geq \mathtt{p} + \mathtt{nw}\end{array}\right\}$$

```
v = p+nw;
```

$$\left\{\begin{array}{l}\exists B_1, B_2.\, \mathit{block}^{*}\, \mathtt{s}\, \mathtt{p}\, B_1 \;*\; \mathtt{p} < \mathtt{q} \;*\; \mathtt{p} \mapsto \mathtt{q} \;*\; \mathop{\ast}_{i=\mathtt{p}+1}^{\mathtt{q}-1}.\, i \mapsto \_ \\ *\; \mathit{block}^{*}\, \mathtt{q}\, \mathtt{t}\, B_2 \;*\; \mathtt{t}_{|1} \mapsto \mathtt{s} \;*\; A = (B_1 \uplus B_2)^{\mathsf{a}} \;*\; \mathit{brka}(\mathtt{t}+1) \\ *\; \mathtt{nw} = 1 + \left\lceil \frac{\mathtt{nbytes}}{\mathtt{WORD}} \right\rceil \;*\; \mathtt{q} \geq \mathtt{v} \;*\; \mathtt{v} = \mathtt{p} + \mathtt{nw}\end{array}\right\}$$

```
if (q>v) {
```

$$\left\{\begin{array}{l}\exists B_1, B_2.\, \mathit{block}^{*}\, \mathtt{s}\, \mathtt{p}\, B_1 \;*\; \mathtt{p} < \mathtt{q} \;*\; \mathtt{p} \mapsto \mathtt{q} \;*\; \mathop{\ast}_{i=\mathtt{p}+1}^{\mathtt{q}-1}.\, i \mapsto \_ \\ *\; \mathit{block}^{*}\, \mathtt{q}\, \mathtt{t}\, B_2 \;*\; \mathtt{t}_{|1} \mapsto \mathtt{s} \;*\; A = (B_1 \uplus B_2)^{\mathsf{a}} \;*\; \mathit{brka}(\mathtt{t}+1) \\ *\; \mathtt{nw} = 1 + \left\lceil \frac{\mathtt{nbytes}}{\mathtt{WORD}} \right\rceil \;*\; \mathtt{q} > \mathtt{v} \;*\; \mathtt{v} = \mathtt{p} + \mathtt{nw}\end{array}\right\}$$

```
  v->ptr = p->ptr;
```

$$\left\{\begin{array}{l}\exists B_1, B_2.\, \mathit{block}^{*}\, \mathtt{s}\, \mathtt{p}\, B_1 \;*\; \mathtt{p} \mapsto \mathtt{q} \;*\; \mathop{\ast}_{i=\mathtt{p}+1}^{\mathtt{v}-1}.\, i \mapsto \_ \\ *\; \mathit{ublock}\, \mathtt{v}\, \mathtt{q}\, \{(\mathtt{v}+1) \mapsto_{\mathsf{u}} (\mathtt{q}-\mathtt{v}-1)\} \\ *\; \mathit{block}^{*}\, \mathtt{q}\, \mathtt{t}\, B_2 \;*\; \mathtt{t}_{|1} \mapsto \mathtt{s} \;*\; A = (B_1 \uplus B_2)^{\mathsf{a}} \;*\; \mathit{brka}(\mathtt{t}+1) \\ *\; \mathtt{nw} = 1 + \left\lceil \frac{\mathtt{nbytes}}{\mathtt{WORD}} \right\rceil \;*\; \mathtt{v} = \mathtt{p} + \mathtt{nw}\end{array}\right\}$$

$$\left\{\begin{array}{l}\exists B_1, B_2.\, \mathit{block}^{*}\, \mathtt{s}\, \mathtt{p}\, B_1 \;*\; \mathtt{p} \mapsto \mathtt{q} \;*\; \mathop{\ast}_{i=\mathtt{p}+1}^{\mathtt{v}-1}.\, i \mapsto \_ \;*\; \mathit{block}^{*}\, \mathtt{v}\, \mathtt{t}\, B_2 \\ *\; \mathtt{t}_{|1} \mapsto \mathtt{s} \;*\; A = (B_1 \uplus B_2)^{\mathsf{a}} \;*\; \mathit{brka}(\mathtt{t}+1) \;*\; \mathtt{nw} = 1 + \left\lceil \frac{\mathtt{nbytes}}{\mathtt{WORD}} \right\rceil \;*\; \mathtt{v} = \mathtt{p} + \mathtt{nw}\end{array}\right\}$$

```
}
```

$$\left\{\begin{array}{l}\exists B_1, B_2.\, \mathit{block}^{*}\, \mathtt{s}\, \mathtt{p}\, B_1 \;*\; \mathtt{p} \mapsto \mathtt{q} \;*\; \mathop{\ast}_{i=\mathtt{p}+1}^{\mathtt{v}-1}.\, i \mapsto \_ \;*\; \mathit{block}^{*}\, \mathtt{v}\, \mathtt{t}\, B_2 \\ *\; \mathtt{t}_{|1} \mapsto \mathtt{s} \;*\; A = (B_1 \uplus B_2)^{\mathsf{a}} \;*\; \mathit{brka}(\mathtt{t}+1) \;*\; \mathtt{nw} = 1 + \left\lceil \frac{\mathtt{nbytes}}{\mathtt{WORD}} \right\rceil \;*\; \mathtt{v} = \mathtt{p} + \mathtt{nw}\end{array}\right\}$$

```
p->ptr = setbusy(v);
```

$$\left\{\begin{array}{l}\exists B_1, B_2.\, \mathit{block}^{*}\, \mathtt{s}\, \mathtt{p}\, B_1 \;*\; \mathtt{p}_{|1} \mapsto \mathtt{v} \;*\; \mathop{\ast}_{i=\mathtt{p}+1}^{\mathtt{v}-1}.\, i \mapsto \_ \;*\; \mathit{block}^{*}\, \mathtt{v}\, \mathtt{t}\, B_2 \\ *\; \mathtt{t}_{|1} \mapsto \mathtt{s} \;*\; A = (B_1 \uplus B_2)^{\mathsf{a}} \;*\; \mathit{brka}(\mathtt{t}+1) \;*\; \mathtt{nw} = 1 + \left\lceil \frac{\mathtt{nbytes}}{\mathtt{WORD}} \right\rceil \;*\; \mathtt{v} = \mathtt{p} + \mathtt{nw}\end{array}\right\}$$

$$\left\{\begin{array}{l}\exists B_1, B_2.\, \mathit{block}^* \,\mathtt{s}\,\mathtt{p}\, B_1 \;*\; \mathit{ablock}\,\mathtt{p}\,\mathtt{v}\,\{\mathtt{p}+1 \mapsto_{\mathtt{a}} \mathtt{nw}-1\} \;*\; \mathit{block}^* \,\mathtt{v}\,\mathtt{t}\, B_2 \\ *\; \mathtt{t}_{|1} \mapsto \mathtt{s} \;*\; A = (B_1 \uplus B_2)^{\mathtt{a}} \;*\; \mathit{brka}(\mathtt{t}+1) \;*\; \mathtt{nw} = 1 + \left\lceil \frac{\mathtt{nbytes}}{\mathtt{WORD}} \right\rceil \\ *\; \mathtt{p}_{|1} \overset{.5}{\mapsto} \mathtt{v} \;*\; \bigstar_{i=\mathtt{p}+1}^{\mathtt{v}-1}.\, i \mapsto \_ \;*\; \mathtt{v} = \mathtt{p} + \mathtt{nw}\end{array}\right\}$$

```
// use lemma to deduce that B1 and p+1 are disjoint
```

$$\left\{\begin{array}{l}\exists B_1, B_2.\, \mathit{block}^* \,\mathtt{s}\,\mathtt{v}\, B_1 \;*\; \mathit{block}^* \,\mathtt{v}\,\mathtt{t}\, B_2 \;*\; \mathtt{t}_{|1} \mapsto \mathtt{s} \\ *\; A \uplus \{\mathtt{p}+1 \mapsto \left\lceil \frac{\mathtt{nbytes}}{\mathtt{WORD}} \right\rceil\} = (B_1 \uplus B_2)^{\mathtt{a}} \\ *\; \mathit{brka}(\mathtt{t}+1) \;*\; \mathtt{p}_{|1} \overset{.5}{\mapsto} \mathtt{p} + \left\lceil \frac{\mathtt{nbytes}}{\mathtt{WORD}} \right\rceil + 1 \\ *\; \bigstar_{i=0}^{\lceil \mathtt{nbytes}/\mathtt{WORD} \rceil - 1}.\, \mathtt{p} + 1 + i \mapsto \_\end{array}\right\}$$

$$\left\{\begin{array}{l}(\mathit{arena}(A \uplus \{\mathtt{ret} \mapsto \left\lceil \frac{\mathtt{nbytes}}{\mathtt{WORD}} \right\rceil\}) \;*\; \bigstar_{i=0}^{\lceil \mathtt{nbytes}/\mathtt{WORD} \rceil - 1}.\, \mathtt{ret} + i \mapsto \_ \\ \;*\; (\mathtt{ret}-1)_{|1} \overset{.5}{\mapsto} \mathtt{ret} + \left\lceil \frac{\mathtt{nbytes}}{\mathtt{WORD}} \right\rceil)[\mathtt{p}+1/\mathtt{ret}]\end{array}\right\}$$

```
return((char *)(p+1));
```

$\{\mathsf{false}\}$

```
}
```

$$\left\{\left(\begin{array}{l}\mathit{arena}(A \uplus \{\mathtt{ret} \mapsto \lceil \mathtt{nbytes}/\mathtt{WORD} \rceil\}) \\ *\; \bigstar_{i=0}^{\lceil \mathtt{nbytes}/\mathtt{WORD} \rceil - 1}.\, \mathtt{ret} + i \mapsto \_ \\ *\; (\mathtt{ret}-1)_{|1} \overset{.5}{\mapsto} \mathtt{ret} + \lceil \mathtt{nbytes}/\mathtt{WORD} \rceil\end{array}\right) \vee (\mathit{arena}\ A \;*\; \mathtt{ret} = 0)\right\}$$

```
// end region update
```

$$\left\{\left(\begin{array}{l}\boxed{\mathit{arena}(A \uplus \{\mathtt{ret} \mapsto \lceil \mathtt{nbytes}/\mathtt{WORD} \rceil\})} \\ *\; \bigstar_{i=0}^{\lceil \mathtt{nbytes}/\mathtt{WORD} \rceil - 1}.\, \mathtt{ret} + i \mapsto \_ \\ *\; (\mathtt{ret}-1)_{|1} \overset{.5}{\mapsto} \mathtt{ret} + \lceil \mathtt{nbytes}/\mathtt{WORD} \rceil\end{array}\right) \vee (\boxed{\mathit{arena}\ A} \;*\; \mathtt{ret} = 0)\right\}$$

```
// end existential
```

$$\left\{\left(\begin{array}{l}\boxed{\exists A.\, \mathit{arena}(A \uplus \{\mathtt{ret} \mapsto \lceil \mathtt{nbytes}/\mathtt{WORD} \rceil\})} \\ *\; \bigstar_{i=0}^{\lceil \mathtt{nbytes}/\mathtt{WORD} \rceil - 1}.\, \mathtt{ret} + i \mapsto \_ \\ *\; (\mathtt{ret}-1)_{|1} \overset{.5}{\mapsto} \mathtt{ret} + \lceil \mathtt{nbytes}/\mathtt{WORD} \rceil\end{array}\right) \vee (\boxed{\exists A.\, \mathit{arena}\ A} \;*\; \mathtt{ret} = 0)\right\}$$

// note that $\exists A.\, \mathit{arena}(A \uplus \{\mathtt{ret} \mapsto \lceil \mathtt{nbytes}/\mathtt{WORD} \rceil\})$ implies $\exists A.\, \mathit{arena}(A)$

$$\left\{\left(\begin{array}{l}\boxed{\exists A.\, \mathit{arena}\ A} \\ *\; \boxed{\exists A.\, \mathit{arena}(A \uplus \{\mathtt{ret} \mapsto \lceil \mathtt{nbytes}/\mathtt{WORD} \rceil\})} \\ *\; \bigstar_{i=0}^{\lceil \mathtt{nbytes}/\mathtt{WORD} \rceil - 1}.\, \mathtt{ret} + i \mapsto \_ \\ *\; (\mathtt{ret}-1)_{|1} \overset{.5}{\mapsto} \mathtt{ret} + \lceil \mathtt{nbytes}/\mathtt{WORD} \rceil\end{array}\right) \vee (\boxed{\exists A.\, \mathit{arena}\ A} \;*\; \mathtt{ret} = 0)\right\}$$

$$\left\{\left(\begin{array}{l}\boxed{\exists A.\, \mathit{uninit}\, A \;\vee\; \mathit{arena}\ A} \\ *\; \boxed{\exists A.\, \mathit{arena}(A \uplus \{\mathtt{ret} \mapsto \lceil \mathtt{nbytes}/\mathtt{WORD} \rceil\})} \\ *\; \bigstar_{i=0}^{\lceil \mathtt{nbytes}/\mathtt{WORD} \rceil - 1}.\, \mathtt{ret} + i \mapsto \_ \\ *\; (\mathtt{ret}-1)_{|1} \overset{.5}{\mapsto} \mathtt{ret} + \lceil \mathtt{nbytes}/\mathtt{WORD} \rceil\end{array}\right) \vee (\boxed{\exists A.\, \mathit{uninit}\, A \;\vee\; \mathit{arena}\ A} \;*\; \mathtt{ret} = 0)\right\}$$

$$\left\{ \begin{pmatrix} anArena \\ * \ token(\texttt{ret}, \lceil \texttt{nbytes/WORD} \rceil) \\ * \ \bigstar_{i=0}^{\lceil \texttt{nbytes/WORD} \rceil - 1}. \texttt{ret} + i \mapsto \_ \end{pmatrix} \vee (anArena \ * \ \texttt{ret} = 0) \right\}$$

$$\left\{ anArena \ * \ ((token\, \texttt{ret}\ \lceil \texttt{nbytes/WORD} \rceil \ * \ \bigstar_{i=0}^{\lceil \texttt{nbytes/WORD} \rceil - 1}. \texttt{ret} + i \mapsto \_) \ \vee \ \texttt{ret} = 0) \right\}$$

Verification of free routine

```
free(register char *ap)
```

$$\left\{ anArena \ * \ \exists n.\, token\ \texttt{ap}\, n \ * \ \bigstar_{i=0}^{n-1}. (\texttt{ap} + i) \mapsto \_ \right\}$$

$$\left\{ \begin{array}{l} \exists n. \boxed{\exists A.\, uninit\, A \ \vee \ arena\, A} \ * \ \boxed{\exists A.\, arena(A \uplus \{\texttt{ap} \mapsto n\})} \ * \ (\texttt{ap} - 1)_{|1} \overset{.5}{\mapsto} (\texttt{ap} + n) \\ * \ \bigstar_{i=0}^{n-1}. (\texttt{ap} + i) \mapsto \_ \end{array} \right\}$$

$$\left\{ \exists n. \boxed{\exists A.\, arena(A \uplus \{\texttt{ap} \mapsto n\})} \ * \ (\texttt{ap} - 1)_{|1} \overset{.5}{\mapsto} (\texttt{ap} + n) \ * \ \bigstar_{i=0}^{n-1}. \texttt{ap} + i \mapsto \_ \right\}$$

```
//begin existential
```

$$\left\{ \boxed{arena(A \uplus \{\texttt{ap} \mapsto n\})} \ * \ (\texttt{ap} - 1)_{|1} \overset{.5}{\mapsto} (\texttt{ap} + n) \ * \ \bigstar_{i=0}^{n-1}. \texttt{ap} + i \mapsto \_ \right\}$$

```
//begin "Free x" action
{
```

$$\left\{ arena(A \uplus \{\texttt{ap} \mapsto n\}) \ * \ (\texttt{ap} - 1)_{|1} \overset{.5}{\mapsto} (\texttt{ap} + n) \ * \ \bigstar_{i=0}^{n-1}. \texttt{ap} + i \mapsto \_ \right\}$$

$$\left\{ \begin{array}{l} \exists B_1, B_2.\, block^*\ \texttt{s}\,\texttt{v}\, B_1 \ * \ block^*\ \texttt{v}\,\texttt{t}\, B_2 \ * \ A \uplus \{\texttt{ap} \mapsto n\} = (B_1 \uplus B_2)^{\mathsf{a}} \ * \ \texttt{t}_{|1} \mapsto \texttt{s} \\ * \ brka(\texttt{t} + 1) \ * \ (\texttt{ap} - 1)_{|1} \overset{.5}{\mapsto} (\texttt{ap} + n) \ * \ \bigstar_{i=0}^{n-1}. \texttt{ap} + i \mapsto \_ \end{array} \right\}$$

```
// use lemma to deduce that B1 and B2 are disjoint
```

$$\left\{ \begin{array}{l} \exists B.\, block^*\ \texttt{s}\,\texttt{t}\, B \ * \ A \uplus \{\texttt{ap} \mapsto n\} = B^{\mathsf{a}} \ * \ \texttt{t}_{|1} \mapsto \texttt{s} \\ * \ brka(\texttt{t} + 1) \ * \ (\texttt{ap} - 1)_{|1} \overset{.5}{\mapsto} (\texttt{ap} + n) \ * \ \bigstar_{i=0}^{n-1}. \texttt{ap} + i \mapsto \_ \end{array} \right\}$$

// note that $\{x \mapsto_{\mathsf{a}} n\} \in B$ implies $\exists B_1, B_2.\, B = B_1 \uplus \{x \mapsto_{\mathsf{a}} n\} \uplus B_2$

$$\left\{ \begin{array}{l} \exists B_1, B_2.\, block^*\ \texttt{s}\, (\texttt{ap} - 1)\, B_1 \ * \ ablock\, (\texttt{ap} - 1)\, (\texttt{ap} + n)\, \{\texttt{ap} \mapsto_{\mathsf{a}} n\} \\ * \ block^*\, (\texttt{ap} + n)\, \texttt{t}\, B_2 \ * \ A \uplus \{\texttt{ap} \mapsto n\} = (B_1 \uplus \{\texttt{ap} \mapsto_{\mathsf{a}} n\} \uplus B_2)^{\mathsf{a}} \ * \ \texttt{t}_{|1} \mapsto \texttt{s} \\ * \ brka(\texttt{t} + 1) \ * \ (\texttt{ap} - 1)_{|1} \overset{.5}{\mapsto} (\texttt{ap} + n) \ * \ \bigstar_{i=0}^{n-1}. \texttt{ap} + i \mapsto \_ \end{array} \right\}$$

// by cancellativity of $\uplus$:

$$\left\{ \begin{array}{l} \exists B_1, B_2.\, block^*\ \texttt{s}\, (\texttt{ap} - 1)\, B_1 \ * \ ablock\, (\texttt{ap} - 1)\, (\texttt{ap} + n)\, \{\texttt{ap} \mapsto_{\mathsf{a}} n\} \\ * \ block^*\, (\texttt{ap} + n)\, \texttt{t}\, B_2 \ * \ A = (B_1 \uplus B_2)^{\mathsf{a}} \ * \ \texttt{t}_{|1} \mapsto \texttt{s} \\ * \ brka(\texttt{t} + 1) \ * \ (\texttt{ap} - 1)_{|1} \overset{.5}{\mapsto} (\texttt{ap} + n) \ * \ \bigstar_{i=0}^{n-1}. \texttt{ap} + i \mapsto \_ \end{array} \right\}$$

```
register struct store *p = (struct store *)ap;
v = --p;
```

$$\left\{\begin{array}{l}\exists B_1, B_2.\, \mathit{block}^* \;\mathtt{s}\,\mathtt{p}\, B_1 \;*\; \mathit{ablock}\,\mathtt{p}\,(\mathtt{p}+1+n)\,\{\mathtt{p}+1 \mapsto_{\mathsf{a}} n\} \\ *\; \mathit{block}^*\,(\mathtt{p}+1+n)\,\mathtt{t}\,B_2 \;*\; A = (B_1 \uplus B_2)^{\mathsf{a}} \;*\; \mathtt{t}_{|1} \mapsto \mathtt{s} \\ *\; \mathit{brka}(\mathtt{t}+1) \;*\; \mathtt{p}_{|1} \overset{.5}{\mapsto} (\mathtt{p}+1+n) \;*\; \bigast_{i=0}^{n-1}.\,\mathtt{p}+1+i \mapsto \_ \;*\; \mathtt{p} = \mathtt{v}\end{array}\right\}$$

$$\left\{\begin{array}{l}\exists B_1, B_2.\, \mathit{block}^* \;\mathtt{s}\,\mathtt{p}\, B_1 \;*\; \mathtt{p}_{|1} \mapsto \mathtt{p}+1+n \;*\; \bigast_{i=0}^{n-1}.\,\mathtt{p}+1+i \mapsto \_ \\ *\; \mathit{block}^*\,(\mathtt{p}+1+n)\,\mathtt{t}\,B_2 \;*\; A = (B_1 \uplus B_2)^{\mathsf{a}} \;*\; \mathtt{t}_{|1} \mapsto \mathtt{s} \;*\; \mathit{brka}(\mathtt{t}+1) \;*\; \mathtt{p} = \mathtt{v}\end{array}\right\}$$

```
p->ptr = clearbusy(p->ptr);
```

$$\left\{\begin{array}{l}\exists B_1, B_2.\, \mathit{block}^* \;\mathtt{s}\,\mathtt{p}\, B_1 \;*\; \mathtt{p} \mapsto \mathtt{p}+1+n \;*\; \bigast_{i=0}^{n-1}.\,\mathtt{p}+1+i \mapsto \_ \\ *\; \mathit{block}^*\,(\mathtt{p}+1+n)\,\mathtt{t}\,B_2 \;*\; A = (B_1 \uplus B_2)^{\mathsf{a}} \;*\; \mathtt{t}_{|1} \mapsto \mathtt{s} \;*\; \mathit{brka}(\mathtt{t}+1) \;*\; \mathtt{p} = \mathtt{v}\end{array}\right\}$$

$$\left\{\begin{array}{l}\exists B_1, B_2.\, \mathit{block}^* \;\mathtt{s}\,\mathtt{p}\, B_1 \;*\; \mathit{ublock}\,\mathtt{p}\,(\mathtt{p}+1+n)\,\{\mathtt{p}+1 \mapsto_{\mathsf{u}} n\} \\ *\; \mathit{block}^*\,(\mathtt{p}+1+n)\,\mathtt{t}\,B_2 \;*\; A = (B_1 \uplus B_2)^{\mathsf{a}} \;*\; \mathtt{t}_{|1} \mapsto \mathtt{s} \;*\; \mathit{brka}(\mathtt{t}+1) \;*\; \mathtt{p} = \mathtt{v}\end{array}\right\}$$

```
// use lemma to deduce that p and B2 are disjoint
```

$$\left\{\exists B_1, B_2.\, \mathit{block}^* \;\mathtt{s}\,\mathtt{v}\, B_1 \;*\; \mathit{block}^*\;\mathtt{v}\,\mathtt{t}\,B_2 \;*\; A = (B_1 \uplus B_2)^{\mathsf{a}} \;*\; \mathtt{t}_{|1} \mapsto \mathtt{s} \;*\; \mathit{brka}(\mathtt{t}+1)\right\}$$

$$\left\{\mathit{arena}\,A\right\}$$

```
}
//end "Free x" action
```

$$\left\{\boxed{\mathit{arena}\,A}\right\}$$

```
//end existential
```

$$\left\{\boxed{\exists A.\, \mathit{arena}\,A}\right\}$$

$$\left\{\boxed{\exists A.\, \mathit{uninit}\,A \;\vee\; \mathit{arena}\,A}\right\}$$

$$\left\{\mathit{anArena}\right\}$$