8 Hoare Logic and Model Checking (jp622)

Consider commands C composed from assignments $X := E$ (where X is a program variable, and E is an arithmetic expression), heap allocation $X := \texttt{alloc}(E_1, ..., E_n)$, heap assignment $[E_1] := E_2$, heap dereference $X := [E]$, disposal of heap locations $\texttt{dispose}(E)$, the no-op `skip`, sequencing $C_1; C_2$, conditionals $\texttt{if}\ B\ \texttt{then}\ C_1\ \texttt{else}\ C_2$ (where B is a boolean expression), and loops $\texttt{while}\ B\ \texttt{do}\ C$. `null` is 0.

Recall the separation logic partial list representation predicates:

$$\mathsf{plist}(t, [], u) = (t = u) \wedge \mathsf{emp}$$
$$\mathsf{plist}(t, h :: \alpha, u) = \exists y.\, ((t \mapsto h) * ((t+1) \mapsto y) * \mathsf{plist}(y, \alpha, u))$$

Circular lists can be represented by $\mathsf{clist}(t, \alpha) = \mathsf{plist}(t, \alpha, t) \wedge (\alpha = [] \Rightarrow t = \texttt{null})$.

(*a*) Assuming $\vdash \{P_1\}\ C_1\ \{Q_1\}$ and $\vdash \{P_2\}\ C_2\ \{Q_2\}$:

(*i*) explain precisely why $\vdash \{P_1 * P_2\}\ C_1; C_2\ \{Q_1 * Q_2\}$ [2 marks]

(*ii*) give a counterexample to $\vdash \{P_1 \wedge P_2\}\ C_1; C_2\ \{Q_1 \wedge Q_2\}$. [1 mark]

(*b*) Give a proof outline for the following circular list 'next' triple:

$$\{\mathsf{clist}(X, t :: \alpha)\}\ X := [X+1]\ \{\mathsf{clist}(X, \alpha +\!\!+ [t])\}$$ [3 marks]

(*c*) Give a loop invariant (no need for a proof outline) for the following circular list 'length' triple:]

$\{\mathsf{clist}(X, \alpha)\}$
$\texttt{if}\ X = \texttt{null}\ \texttt{then}\ Y := 0$
$\texttt{else}\ (Z := [X+1]; Y := 1; \texttt{while}\ Z \neq X\ \texttt{do}\ (Z := [Z+1]; Y := Y+1))$
$\{\mathsf{clist}(X, \alpha) * Y = \mathit{length}(\alpha)\}$

[3 marks]

(*d*) Give a loop invariant (no need for a proof outline) for the following triple for a 'previous' operation on non-empty circular lists:

$\{\mathsf{clist}(X, \alpha +\!\!+ [t])\}$
$Z := X; Y := [X+1]; (\texttt{while}\ Y \neq X\ \texttt{do}\ (Z := Y; Y := [Y+1])); X := Z$
$\{\mathsf{clist}(X, t :: \alpha)\}$

[4 marks]

(*e*) Give a loop invariant (no need for a proof outline) for the following triple for a 'dial to minimum' operation on non-empty circular lists:

$\{\mathsf{clist}(X, \alpha_1 +\!\!+ [t] +\!\!+ \alpha_2) \wedge \mathit{sorted}(t :: \mathit{merge}(\mathit{sort}(\alpha_1), \mathit{sort}(\alpha_2)))\}$
$Z := X; M := [X]; Y := [X+1];$
$(\texttt{while}\ Y \neq Z\ \texttt{do}$
$\quad (N := [Y]; (\texttt{if}\ N < M\ \texttt{then}\ X := Y\ \texttt{else skip}); Y := [Y+1]));$
$\{\mathsf{clist}(X, [t] +\!\!+ \alpha_2 +\!\!+ \mathit{reverse}(\alpha_1))\}$

[5 marks]

(*f*) Describe precisely *all* pairs of a stack and a heap that satisfy

$$\exists y, z.\, ((X \mapsto y * y \mapsto z * z \mapsto X) \wedge Y = 0)$$

[2 marks]