Q1) a) i)

$$\dfrac{\dfrac{\langle x+1, S_0\rangle \Downarrow 1 \quad S_0[x\mapsto 1] = S'}{\langle x := x+1, S_0\rangle \Downarrow S'} \quad \dfrac{\dfrac{\langle x+1, S'\rangle \Downarrow 2 \quad S'[x\mapsto 2] = S_2}{\langle x := x+1, S'\rangle \Downarrow S_2} \quad \dfrac{}{\langle C, S_2\rangle \Downarrow S_2}}{\langle C, S'\rangle \Downarrow S_2}}{\langle C, S_0\rangle \Downarrow S_2.}$$

ii) Performs ~~a~~ a loop where the expression inside the loop is evaluated any number of times before terminating.
It ~~performs~~ runs the body inside it any no. of times because of the non-determinism in NONDET, we can evaluate $\langle \text{loop}\{c\}, s\rangle$ in 2 ways

iii) Set of possible values for x are ~~the~~ $\mathbb{N} \cup \{0\}$, natural numbers including 0.

This is because the loop can perform as many iterations of the body as needed before terminating thanks to the 2 rules given to evaluate it.

b)

$$\dfrac{}{\langle \text{or}(C_1, C_2), s\rangle \rightarrow \langle C_1, s\rangle}$$

$$\dfrac{}{\langle \text{or}(C_1, C_2), s\rangle \rightarrow \langle C_2, s\rangle}$$

$$\overline{\langle loop(c), s\rangle \to \langle skip, s\rangle}$$

$$\overline{\langle loop(C), s\rangle \to \langle C; loop(C), s\rangle}$$

c) Let the property $P(i)$ be defined as

$$\forall C, s, s' \; \langle C, s\rangle \Downarrow_i s' \Rightarrow \langle f(C), s\rangle \Downarrow s'$$

~~Base cases~~:

Base case: $i=0$.

Take an arbitrary ~~C, s, s'~~ ~~commands of while~~ $s, s' \in$ states.

Assume $\langle C, s\rangle \Downarrow_0 s'$:

3 rules which have depth 0:

Case 1: $C =$ skip, in this case $s = s'$ from WHILE rule for skip.

$f(skip) = skip$.

So $\overline{\langle skip, s\rangle \Downarrow s}$ (NONDET)

$$\overline{\langle skip, s\rangle \Downarrow_0 s}$$

So $\langle skip, s\rangle \Downarrow_0 s \Rightarrow \langle f(skip), s\rangle \Downarrow s$

Case 2: $C = x := E$

$$\frac{\langle E, s\rangle \Downarrow n \quad s[x \mapsto n] = s'}{\langle x := E, s\rangle \Downarrow_0 s'} \quad \text{(WHILE)}$$

$f(x := E) = x := E$

$$\frac{\langle E, s\rangle \Downarrow n \quad s[x \mapsto n] = s''}{\langle x := E, s\rangle \Downarrow s''} \quad \text{(NONDET)}$$

So $s'' = s' = s[x \mapsto n]$

So $\langle x := E, s\rangle \Downarrow_0 s' \Rightarrow \langle f(x := E), s\rangle \Downarrow s'$

case 3: $C = \text{while } B \text{ do } C$, where $\langle B, s\rangle \Downarrow \text{false}$.

$$\frac{\langle B, s\rangle \Downarrow \text{false}}{\langle \text{while } B \text{ do } C, s\rangle \Downarrow_0 s} \quad (\text{WHILE})$$

$f(\text{while } B \text{ do } C) = \text{loop}(\text{assume } B; f(C)); \text{ assume } \neg B.$

from lemma & $\langle B, s\rangle \Downarrow \text{false}$

$$\frac{\langle \text{loop}(\text{assume } B, f(C)), s\rangle \Downarrow s \qquad \dfrac{\langle \neg B, s\rangle \Downarrow \text{true}}{\langle \text{assume } \neg B, s\rangle \Downarrow \cancel{\text{false}}\ \cancel{\text{true}}\ s}}{\langle \text{loop}(\text{assume } B; f(C)); \text{assume } \neg B, s\rangle \Downarrow \cancel{\text{assume}}\ s} \quad (\text{NONDET})$$

So $\langle \text{while } B \text{ do } C, s\rangle \Downarrow_0 s \Rightarrow$
$\langle \text{loop}(\text{assume } B; f(C)); \text{assume } \neg B, s\rangle \Downarrow s$

So we have $P(0)$

Inductive Hypothesis:
~~$P(0)$~~ $P(1), P(2) \dots P(K)$ holds

Inductive Case: $i = K+1$

We have 4 rules with a depth of ≥ 1.

Take arbitrary s, s' which are states

Case 1: $C = \text{if } B \text{ then } C_1 \text{ else } C_2$

Assume $\langle B,s\rangle \Downarrow true$

$$\frac{\langle B,s\rangle \Downarrow true \qquad \langle C_1,s\rangle \Downarrow_K s'}{\langle \text{if } B \text{ then } C_1 \text{ else } C_2, s\rangle \Downarrow_{K+1} s'} \quad (\text{WHILE})$$

$$f(C) = or((assume\ B; f(C_1)), (assume\ \neg B; f(C_2)))$$

$$\cfrac{\cfrac{\cfrac{\langle B,s\rangle \Downarrow true}{\langle assume\ B, s\rangle \Downarrow s} \qquad \langle f(C_1), s\rangle \Downarrow s'}{\langle assume\ B; f(C_1), s\rangle \Downarrow s'}}{\langle or((assume\ B; f(C_1)), (assume\ \neg B; f(C_2))), s\rangle \Downarrow s'}$$

$\rightarrow$ From IH of $P(K)$ & derivation shown

Case 2:

Assume $\langle B,s\rangle \Downarrow false$

So we have $\langle \neg B, s\rangle \Downarrow true$

$$\frac{\langle B,s\rangle \Downarrow false \qquad \langle C_2,s\rangle \Downarrow_K s'}{\langle \text{if } B \text{ then } C_1 \text{ else } C_2, s\rangle \Downarrow_{K+1} s'}$$

$$f(C) = or((assume\ B; f(C_1)), (assume\ \neg B; f(C_2)))$$

$$\cfrac{\cfrac{\cfrac{\langle \neg B,s\rangle \Downarrow true}{\langle assume\ \neg B, s\rangle \Downarrow s} \qquad \langle f(C_2), s\rangle \Downarrow s'}{\langle assume\ \neg B; f(C_2), s\rangle \Downarrow s'}}{\langle or((assume\ B; f(C_1)), (assume\ \neg B; f(C_2))), s\rangle \Downarrow s'}$$

$\rightarrow$ from IH of $P(K)$ and derivation tree shown case 3: $C = $ while B do C , assume $\langle B, s\rangle \Downarrow$ true

$$\frac{\langle B, s\rangle \text{ true} \quad \langle C, s\rangle \Downarrow_i s'' \quad \langle \text{while } B \text{ do } C, s''\rangle \Downarrow_j s' \qquad k = \max(i, j)}{\langle \text{while } B \text{ do } C, s\rangle \Downarrow_{k+1} s'} \text{(WHILE)}$$

$f(C) = $ loop (assume B; $f(C)$); assume $\neg B$.

$$\cfrac{\cfrac{\cfrac{\langle B, s\rangle \Downarrow \text{true}}{\langle \text{assume } B, s\rangle \Downarrow s} \quad \langle f(C), s\rangle \Downarrow s'}{\langle \text{assume } B; f(C), s\rangle \Downarrow s''} \quad \langle \text{assume } \neg B, s'\rangle \Downarrow s'}{\langle \text{loop (assume } B; f(C)); \text{ assume } \neg B, s\rangle \Downarrow s'}$$

case 4. $C = C_1; C_2$

$$\frac{\langle C_1, s\rangle \Downarrow_i s'' \quad \langle C_2, s''\rangle \Downarrow_j s' \qquad k = \max(i, j)}{\langle C_1; C_2, s\rangle \Downarrow_{k+1} s'}$$

$f(C_1; C_2) = f(C_1); f(C_2)$

$$\frac{\langle C_1, s\rangle \Downarrow s'' \quad \langle C_2, s''\rangle \Downarrow s'}{\langle f(C_1); f(C_2), s\rangle \Downarrow s'}$$

So we have $P(k+1)$

As we have shown for $P(0)$ & assumed $P(1) \ldots P(k)$ and shown $P(k+1)$ by induction we have $\forall i\, P(i)$

Q2) a) $C(I) = 01739144.$
$= 0b1101010001001100001000$

$\langle\langle x, y\rangle\rangle = \langle\langle 3, 0b1101010001001100001000\rangle\rangle$
$= \langle\langle 3, 108696\rangle\rangle$

$\ulcorner L \urcorner = \ulcorner 3, 3, 0, 2, 3, 1, 1, 0 \urcorner$

$\cancel{3 = 0b11}$ $\qquad 0 = \ulcorner \cancel{H}ALT \urcorner$

$\cancel{3 = \langle\langle 0, 1\rangle\rangle}$
$\cancel{= \ulcorner R_0^+ \to L_1 \urcorner}$

$\cancel{2 = 0b10}$
$\cancel{= \langle\langle 1, 0\rangle\rangle}$
$\cancel{= \langle\langle 1, \langle 0, 0\rangle\rangle\rangle}$
$\cancel{= \ulcorner R_1^- \to L_0, L_0 \urcorner}$

$\cancel{1 = \langle\langle 0, 0\rangle\rangle}$
$\cancel{= \ulcorner R_0^+ \to L_0 \urcorner}$

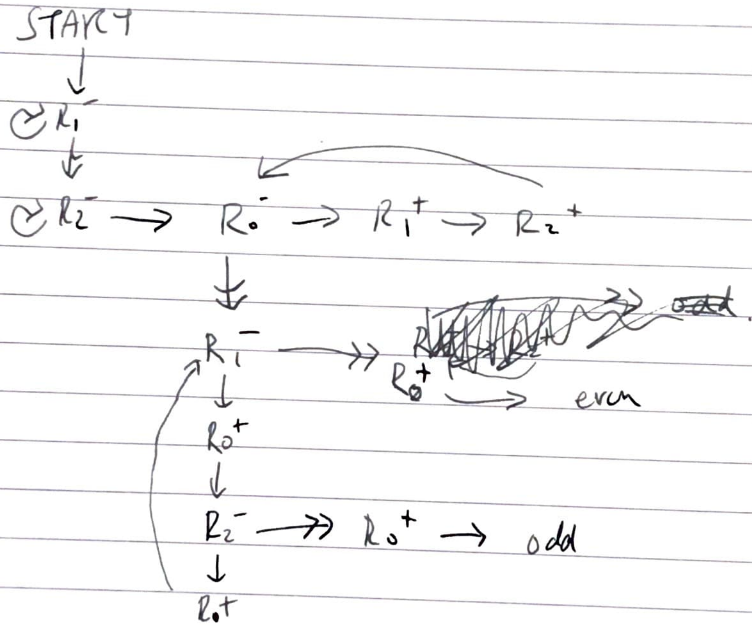
So Assuming "instructions" mean just one instruction

$\langle\langle 3, 108696\rangle\rangle = \langle\langle 3, \langle 0, 54348\rangle\rangle\rangle$
$= \ulcorner R_1^- \to L_0, L_{54348} \urcorner$

So

$\therefore I = R_1^- \to L_0, L_{54348}$

b) i)



ii) $L_0 : R_1^- \to L_1, L_6$

$L_1 : R_0^- \to L_2, L_6$

$L_2 : R_0^+ \to L_3$

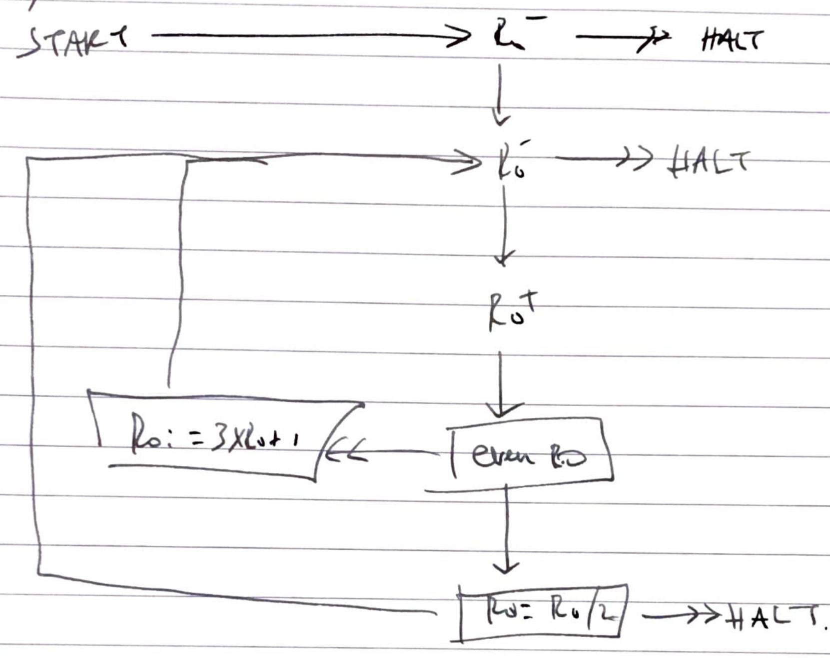
$L_3 : R_1^- \to L_4, L_5$

$L_4 : R_2^- \to L_1, L_6$

$L_5 : R_0^- \to L_1, L_6$

L_6 : HALT.

iii)



$R_0 = 1$	2	3	4	5
0	1	2	3	4
HALT	0	1	2	3
	1	2	3	4
	4	1	10	⋮
	3	0	9	Infinite loop.
	4	1	10	
	2	4	5	
	1	⋮	11	
	2	Infinite loop	15	
	1		16	
	0		8	
	1		7	
	4		8	
	3		4	
	4		⋮	
	⋮		Infinite loop.	
	Infinite loop.			

Seems to be run infinitely except for value $k = 1$.

So it's computing the function:

$$f(x) = \begin{cases} 0 & \text{if } x = 1 \\ \text{undefined} & \text{else} \end{cases}$$

iv)

c) $Y' = VV$ $\quad V = \lambda ux. x(uux)$

They are closed λ-terms
All identities are bound by the binding occurrences.

$$Y' = (\lambda ux.\ x(uux))\ (\lambda ux.\ x(uux)$$

~~reduces to~~

Y' ux reduces to u