Computation Assessed Coursework II: Solutions

1. (a) (0,0,7), (3,0,6), (1,0,5), (4,0,4), (0,1,4), (3,1,3), (1,1,2), (4,1,1), (0,2,1), (3,2,0), (2,2,0).
(Note: the first component is the program counter.)

[5 marks]

(b) $f(x)$ is the integer part of x divided by 3. $f(x) = \lfloor \frac{x}{3} \rfloor$.
Use of `div` to mean integer division is acceptable.

[2 marks]

(c)

$$\begin{aligned}
\ulcorner R_1^- \to L_3, L_5 \urcorner &= \langle\!\langle 2 \times 1 + 1, \langle 3, 5 \rangle \rangle\!\rangle = \langle\!\langle 3, 2^3(2 \times 5 + 1) - 1 \rangle\!\rangle \\
&= \langle\!\langle 3, 87 \rangle\!\rangle = 2^3(2 \times 87 + 1) = 1400 \\
\ulcorner R_1^- \to L_4, L_2 \urcorner &= \langle\!\langle 2 \times 1 + 1, \langle 4, 2 \rangle \rangle\!\rangle = \langle\!\langle 3, 2^4(2 \times 2 + 1) - 1 \rangle\!\rangle \\
&= \langle\!\langle 3, 79 \rangle\!\rangle = 2^3(2 \times 79 + 1) = 1272 \\
\ulcorner HALT \urcorner &= 0 \\
\ulcorner R_1^- \to L_1, L_2 \urcorner &= \langle\!\langle 2 \times 1 + 1, \langle 1, 2 \rangle \rangle\!\rangle = \langle\!\langle 3, 2^1(2 \times 2 + 1) - 1 \rangle\!\rangle \\
&= \langle\!\langle 3, 9 \rangle\!\rangle = 2^3(2 \times 9 + 1) = 152 \\
\ulcorner R_0^+ \to L_0 \urcorner &= \langle\!\langle 2 \times 0, 0 \rangle\!\rangle = 2^0(2 \times 0 + 1) = 1 \\
\ulcorner HALT \urcorner &= 0
\end{aligned}$$

[3 marks]

[Total for question: 10 marks]

2. We first compute the list corresponding to $2^{94} \times 16395$.

$$2^{94} \times 16395 = 2^{94}(2 \times 8197 + 1) = \langle\!\langle 94, 8197 \rangle\!\rangle$$

Note that 8197 is indivisible by 2, so we have:

$$8197 = 2^0(2 \times 4098 + 1) = \langle\!\langle 0, 4098 \rangle\!\rangle$$

Now, we have

$$4098 = 2 \times 2049 = 2^1(2 \times 1024 + 1) = \langle\!\langle 1, 1024 \rangle\!\rangle$$

Finally,

$$1024 = 2^{10}(2 \times 0 + 1) = \langle\!\langle 10, 0 \rangle\!\rangle$$

So we have

$$2^{94} \times 16395 = \ulcorner [94, 0, 1, 10] \urcorner$$

Decoding the list by use of the binary representation is also acceptable.

Now we decode each instruction in this list.

$$
\begin{aligned}
94 &= 2^1(2 \times 23 + 1) = \langle\!\langle 1, 23 \rangle\!\rangle = \langle\!\langle 2 \times 0 + 1, 2^3 \times (2 \times 1 + 1) - 1 \rangle\!\rangle \\
&= \langle\!\langle 2 \times 0 + 1, \langle 3, 1 \rangle \rangle\!\rangle = \ulcorner R_0^- \to L_3, L_1 \urcorner \\
0 &= \ulcorner HALT \urcorner \\
1 &= 2^0(2 \times 0 + 1) = \langle\!\langle 0, 0 \rangle\!\rangle = \langle\!\langle 2 \times 0, 0 \rangle\!\rangle = \ulcorner R_0^+ \to L_0 \urcorner \\
10 &= 2^1 \times (2 \times 2 + 1) = \langle\!\langle 1, 2 \rangle\!\rangle = \langle\!\langle 2 \times 0 + 1, 2^0 \times (2 \times 1 + 1) - 1 \rangle\!\rangle \\
&= \langle\!\langle 2 \times 0 + 1, \langle 0, 1 \rangle \rangle\!\rangle = \ulcorner R_0^- \to L_0, L_1 \urcorner
\end{aligned}
$$

The code is:

$$
\begin{aligned}
L_0 &: R_0^- \to L_3, L_1 \\
L_1 &: HALT \\
L_2 &: R_0^+ \to L_0 \\
L_3 &: R_0^- \to L_0, L_1
\end{aligned}
$$

The graph is:

$$
\begin{tikzcd}
 & R_0^+ \arrow[d] & \\
START \arrow[r] & R_0^- \arrow[r, two heads] \arrow[d, bend right] & HALT \\
 & R_0^- \arrow[u, bend right] \arrow[ur] &
\end{tikzcd}
$$

(Note that, while instruction L_3 is not reachable in the execution of the register machine, it must still be included in the graph. Also, if there are two nodes labelled $HALT$ in the graph, it implies that there are two $HALT$ instructions; this machine only has one.)

[Total for question: 11 marks]

3. (a) Such a machine is:

$$
\begin{tikzcd}
 & {} \arrow[rrr, phantom, "\overbrace{\hphantom{R_1^+ \longrightarrow \cdots \longrightarrow R_1^+}}^{n \text{ times}}"] & & & & & \\
START \arrow[r] & R_1^+ \arrow[r] & \cdots \arrow[r] & R_1^+ \arrow[r] & R_1^- \arrow[r, two heads] \arrow[dl] & {} & \\
 & & & R_0^+ \arrow[r] & R_0^+ \arrow[u] & &
\end{tikzcd}
$$

An alternative is:

$$
\begin{tikzcd}
 & & {} \arrow[rrr, phantom, "\overbrace{\hphantom{R_0^+ \longrightarrow \cdots \longrightarrow R_0^+}}^{n \text{ times}}"] & & & & \\
START \arrow[r] & R_1^+ \arrow[r] & R_0^+ \arrow[r] & \cdots \arrow[r] & R_0^+ \arrow[r] & R_1^- \arrow[r, two heads] \arrow[llll, bend left] & HALT
\end{tikzcd}
$$

Variations of either of these are possible: the first increments R_0 by 2 n times, while the second increments R_0 by n twice. It was not necessary to prove that your machine was correct, as long as the marker could see that it was.

[2 marks]

(b) We can construct a machine with $n_\phi + 7$ instructions that loads $2n_\phi + 8$ into register R_1, by simply renaming the registers in part (a). We then feed that into M_ϕ, which has n_ϕ instructions. We redirect the halting instructions of M_ϕ through a final instruction that increments R_0. The resulting machine therefore has $2n_\phi + 8$ instructions and computes $\phi(2n_\phi + 8) + 1$.

[9 marks]

(c) If S were a computable function, there would be some register machine M_S that computes it, having, say, n_S instructions. By part (b), we could construct a machine with $2n_S + 8$ instructions that computes $S(2n_S + 8) + 1$. This contradicts that fact that $S(2n_S + 8)$ is the maximum output of any register machine with $2n_S + 8$ instructions. Consequently, S cannot be a computable function.

[6 marks]

[Total for question: 17 marks]